ACE

Applied and Computational Engineering

Proceedings of the 5th International Conference
on Computing and Data Science

Macau SAR, China

July 14 – July 21, 2023

EWA Publishing

Volume 15

Editors
Alan Wang
University of Auckland

Marwan Omar
Illinois Institute of Technology

Roman Bauer
University of Surrey

ISSN: 2755-2721
ISSN: 2755-273X (eBook)

ISBN: 978-1-83558-021-9
ISBN: 978-1-83558-022-6 (eBook)

Publication of record for individual papers is online:
https://ace.ewapublishing.org/

The publisher, the editors and the authors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the editors or the authors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This imprint is published by **EWA Publishing**
Address: John Eccles House, Robert Robinson Avenue, Oxford, England, OX4 4GP
Email: info@ewapublishing.org

Committee Members

CONF-CDS 2023

General Chairs

Alan Wang, *University of Auckland*
Anil Fernando, *University of Strathclyde*

Organizing Chair

Roman Bauer, *University of Surrey*

Organizing Committee

Festus Adedoyin, *Bournemouth University*
Haseeb Ahmad, *National Textile University, Faisalabad*
Richa Gupta, *Jamia Hamdard*
Tauqir Nasir, *Eastern Mediterranean University*
Alex Siow, *National University of Singapore*
Ce Li, *China University of Mining and Technology*
Brajendra Panda, *University of Arkansas*
Long Bao, *Apple Inc.*
Guoqing Xiang, *Peking University*
Karen Works, *Florida State University*
Rahul Kumar Dubey, *Robert Bosch GmbH*
Lei Shu, *University of Texas at Austin*

Technical Program Chair

Marwan Omar, *Illinois Institute of Technology*

Technical Program Committee

Michael Harre, *The University of Sydney*
Lewis Tseng, *Clark University*
Mukhtar Ullah, *FAST NUCES Islamabad*
Sameena Naaz, *Jamia Hamdard*
Kourosh Khoshelham, *University of Melbourne*
Weijia Cao, *Aerospace Information Research Institute, Chinese Academy of Sciences*
Zachary Ziegler, *Harvard University*
Haidong Xie, *China Academy of Space Technology*

Publicity Committee

Shuang-Hua Yang, *University of Reading*
Dadmehr Rahbari, *Tallinn University of Technology*
James Duncan-Brown, *University of South Africa*

Preface

The 5th International Conference on Computing and Data Science (CONF-CDS 2023) is an annual conference focusing on research areas including computing technology, machine learning, computer science, and data science. It aims to establish a broad and interdisciplinary platform for experts, researchers, and students worldwide to present, exchange, and discuss the latest advance and development in computing technology, machine learning, computer science, and data science.

This volume contains the papers of the 5th International Conference on Computing and Data Science (CONF-CDS 2023). Each of these papers has gained a comprehensive review by the editorial team and professional reviewers. Each paper has been examined and evaluated for its theme, structure, method, content, language, and format.

Cooperating with prestigious universities, CONF-CDS 2023 organized three workshops in Chicago, Guildford and Auckland. Dr. Marwan Omar chaired the workshop “Adversarial Machine Learning”, which was held at Illinois Institute of Technology. Dr. Roman Bauer chaired the workshop “Computational Modeling of Complex System Dynamics” at University of Surrey. Dr. Alan Wang chaired the workshop “Workshop on Intelligent Computing for Medical Data Analysis 2023 (WICMDA 2023)” at University of Auckland.

Besides these workshops, CONF-CDS 2023 also held an online session. Eminent professors from top universities worldwide were invited to deliver keynote speeches in this online session, including Dr. Michael Harre from The University of Sydney, Dr. Festus Adedoyin from Bournemouth University, Dr. Lewis Tseng from Clark University, etc. They have given keynote speeches on related topics of computing technology, machine learning, computer science, and data science, and other information technology areas.

On behalf of the committee, we would like to give sincere gratitude to all authors and speakers who have made their contributions to CONF-CDS 2023, editors and reviewers who have guaranteed the quality of papers with their expertise, and the committee members who have devoted themselves to the success of CONF-CDS 2023.

Dr. Alan Wang
Dr. Anil Fernando
General Chairs of Conference Committee

Workshop

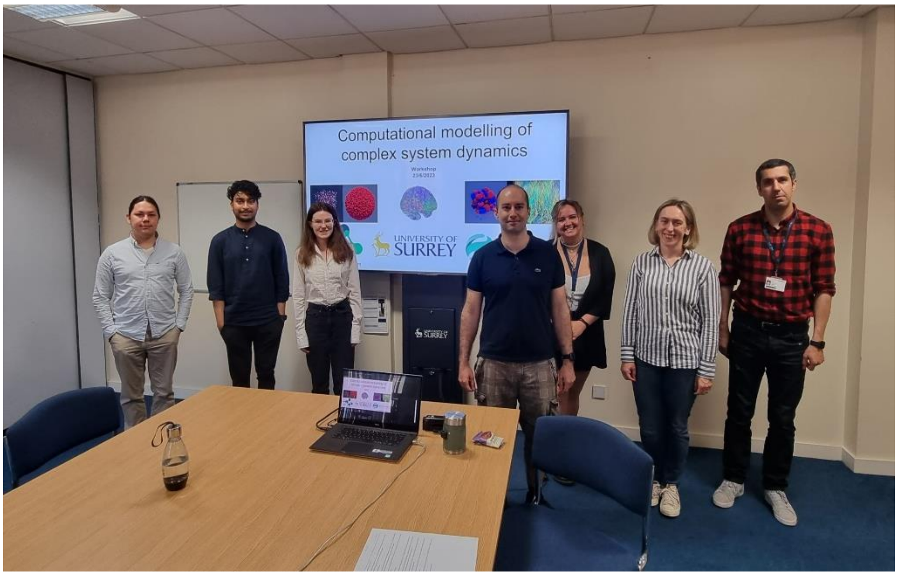
Workshop – Chicago: Adversarial Machine Learning

June 14th, 2023 (CDT)

ITM Department, Illinois Institute of Technology, USA

Workshop Chair: Dr. Marwan Omar, Associate Professor in Illinois Institute of Technology

Workshop – Guildford: Computational Modeling of Complex System Dynamics

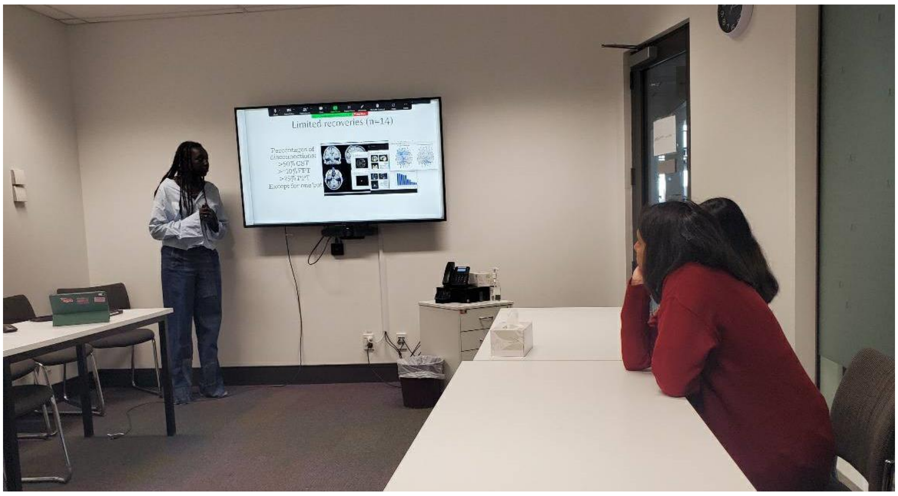


June 23rd, 2023 (GMT+1)

Surrey Space Center, University of Surrey

Workshop Chair: Dr. Roman Bauer, Lecturer in University of Surrey

Workshop – Auckland: Workshop on Intelligent Computing for Medical Data Analysis 2023 (WICMDA 2023)



July 13th, 2023 (GMT+12)

Faculty of Medical and Health Sciences and Bioengineering Institute, University of Auckland

Workshop Chair: Dr. Alan Wang, Associate Professor in University of Auckland

The 5th International Conference on Computing and Data Science

CONF-CDS 2023

Table of Contents

Structural analysis of U-Net and its variants in the field of medical image segmentation

Yanjia Kan

School of Artificial Intelligence, Xi'an Jiaotong-liverpool University, Suzhou, 215123, China

Yanjia.Kan20@student.xjtlu.edu.cn

Abstract. Medical image segmentation can provide valuable information for doctors, it has important research value in the medical field. Meanwhile, U-Net, as the fundamental networks for such tasks, brings a substantial improvement in the segmentation performance of traditional medical images. With the increasingly widespread use of U-Net, researchers have designed various U-Net variants according to different task requirements. However, most of the current summaries of U-Net variants are divided according to the direction of network applications, and the structural relationship between the variant networks and U-Net is not elaborated. Therefore, this paper classifies U-Net variants according to their network framework by elaborating the principles of U-Net structure. According to the U-Net network structure, it is divided into three main categories: backbone improvement, module addition and cross-network fusion. Further, the characteristics, advantages and disadvantages of different categories of variants are introduced, and the directions of the variants for U-Net optimization are analyzed. Finally, the article summarizes the current development direction of U-Net variants and provides an outlook on the future directions that can continue to be optimized.

Keywords: U-Net, medical image segmentation, U-Net variants.

1. Introduction

In medical tasks, the main purpose of image segmentation is to segment regions with medical research value from images. Medical images are various, such as magnetic resonance images, ultrasound images, etc. The segmented regions generally have special features. This feature can assist in clinical diagnosis and treatment, as well as provide valid evidence for pathological studies. Therefore, a medical image with accurately segmented lesions can greatly improve the accuracy and effectiveness of disease treatment in the later stage for medical personnel. However, in practical situations, due to the limitations of multiple parties such as acquisition equipment, deterioration of the lesion, and structural peculiarities of the organ, the image segmentation is very difficult. Medical images are complex and some images lack obvious features, especially in the segmentation edges where discriminative linear features may be missing. In addition, medical images are often affected by noise and volume effects, such as uneven gray scale and artifacts. Therefore, traditional segmentation tasks often require mature doctors to complete. However, human judgment is often influenced by many factors, resulting in fluctuations in the accuracy of segmentation. As a result, image segmentation field introduced various segmentation algorithms and neural networks, which greatly reduces the consumption of human and material and the losses caused by segmentation errors. As of today, neural networks have penetrated many levels of medical tasks with remarkable achievements. Meanwhile, with the continuous emergence of new technologies, the application of it also has more prospects.

Convolutional neural networks (CNN) have good feature extraction and generalization abilities, and image semantic segmentation is one of the important branches. The vigorous development of semantic segmentation is due to the FCN structure proposed by Jonathan Long and others [1]. Compared to CNN, FCN discards the structure used as classification output behind it and uses convolutional layer to replace its function. In addition, the network adopts a skip connection structure, which realize the conversion of network output from probability to image. Inspired by FCN, Ronneberger and others innovatively created U-Net based on FCN. As one of the variants of FCN, U-Net designs the network structure as a symmetrical network, and the skip connection part uses splicing operations instead of the pixel-by-pixel addition method of FCN, thereby greatly improving the network's feature extraction ability [2]. Therefore, with its simple and flexible characteristics and excellent segmentation ability, U-Net is preferred as the test standard for many segmentation tasks.

In view of this, this article takes U-Net as the core, introduces its network and typical network variants. Subsequently, based on its structure, the variants are divided into three categories (Figure 1). By explaining the U-Net variant structure, analyzing its optimization characteristics, and summarizing the optimization ideas of the U-Net network. Finally, the problems and challenges faced by U-Net are summarized, and the future development direction of the U-Net network is prospected.

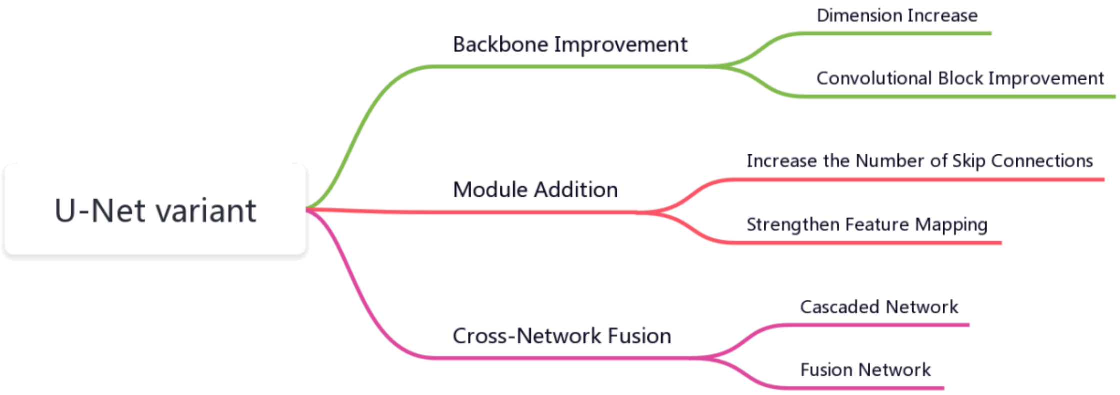


Figure 1. U-Net taxonomy.

2. 2D U-Net

U-Net is mainly adopting an encoder-decoder and skip connection structure to achieve fast and accurate end-to-end network training even with limited data. The encoder part is responsible for capturing contextual information while the decoder part is for mapping features to the required resolution. This includes continuous convolutional operations, bottleneck design, 4 downsampling, and 4 upsampling. In addition, to ensure that the network still retains low-level semantic features in deep structures, the network uses skip connections (Figure 2). This method combines the semantic features of the decoder at the same scale with the encoder’s deep features of to enrich the localization information in the mapping. Finally, the network also uses an overlapping tile strategy to solve the boundary information loss, uses data augmentation to solve the insufficient training data.

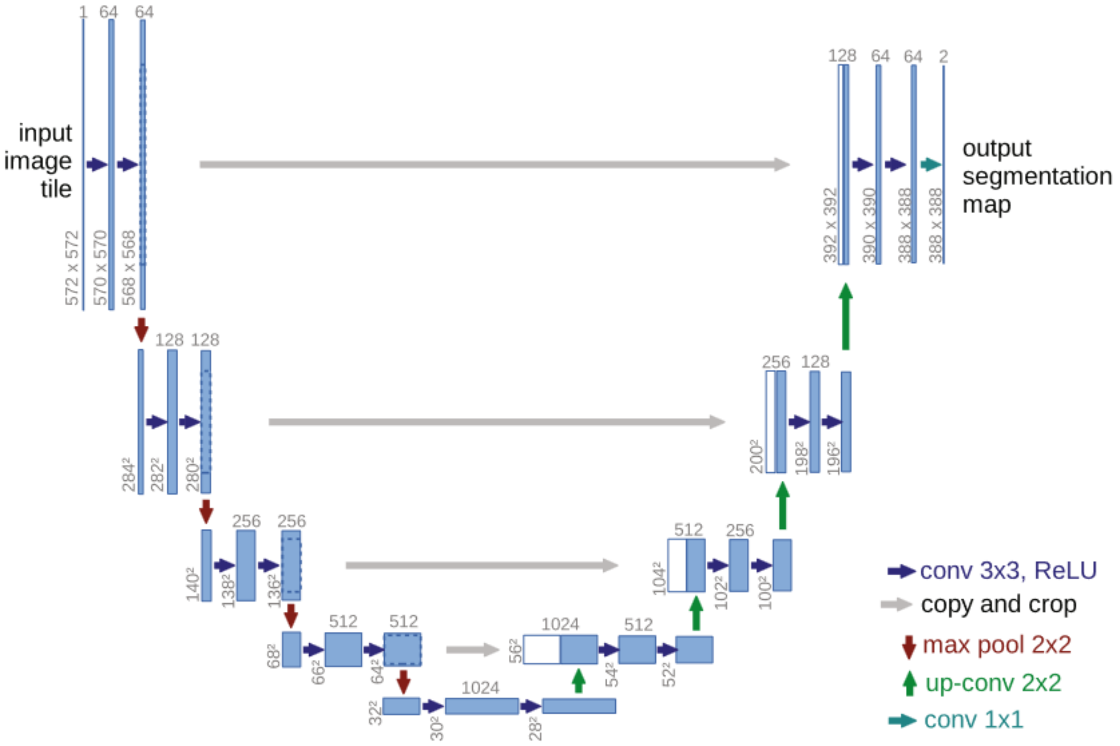


Figure 2. The architecture of U-Net [2].

U-Net, as an important network structure in the field of medical image semantic segmentation, is used to assist in several image analysis tasks. For example, magnetic resonance imaging analysis, computed tomography scan analysis, and ultrasound imaging analysis, etc. However, different images require different feature extraction methods. Therefore, in recent years, a lot of improved methods have been created to make U-Net have highly network suitability for specialized medical images. According to the unique network structure of U-Net, common improvement methods generally optimize the encoder-decoder structure and skip connection structure, resulting in various U-Net variants. These variants can be divided into three categories according to their optimized network positions: backbone improvement, module addition, and cross-network fusion.

3. Variant classification

3.1. Backbone improvements

In U-Net, the backbone network defines how the layers in the encoder are arranged, and its corresponding part is used to describe the architecture of the decoder. Therefore, backbone improvement mainly refers to the improvement of the network encoder-decoder part.

3.1.1. Dimension increase. U-Net is sometimes referred to as 2D U-Net because the network design is based on 2D images as input and output. However, medical images also include a lot of 3D image datasets. Traditional methods involve annotating data by slicing 2D images, which results in redundant data annotation between adjacent slices, tedious annotation process, huge calculation and other problems. Therefore, 3D U-Net, a 3D segmentation method which can converts 2D image operations to 3D image operations was proposed [3]. 3D U-Net retains the U-shaped network structure and makes adjustments to it. The upsampling and downsampling are set to three times, and the image channel remains the same in each deconvolution operation. Changing the channel operation is set in the first convolution after each sampling. Finally, batch normalization is added to accelerate convergence. 3D U-Net learned from sparsely annotated 3D images and provided dense 3D segmentation results, which were validated in the African clawed frog kidney confocal microscopy dataset with higher accuracy than 2D U-Net. V-Net is also used for 3D image segmentation [4]. V-Net replaces the upsampling and downsampling parts in U-Net with 2×2×2 convolution kernels with a stride of 2. It also restores four upsampling and four downsampling operations and changes the number of feature maps during convolution (Figure 3). In addition, V-Net introduces residual networks and a new objective function, the dice coefficient, to achieve end-to-end training for prostate MRI images.

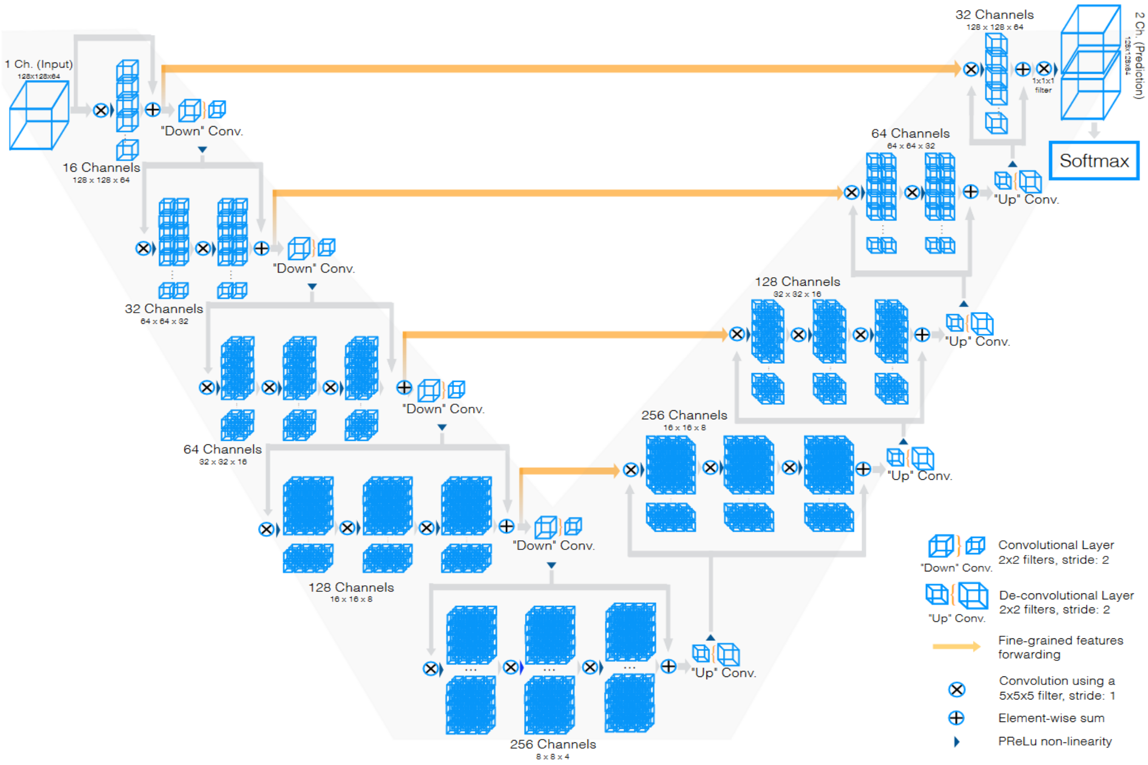


Figure 3. The architecture of V-Net [4].

3.1.2. Convolutional block improvement. The convolutional block of the U-Net network is also a structure that can be optimized. Since the introduction of ResNet, residual blocks have been widely used due to their excellent network performance enhancement capabilities. Since the residual block has a positive effect on avoiding spatial information loss and improving the accuracy of semantic segmentation networks. Therefore, the residual blocks are also used by V-net and ResU-Net to increase the extraction effect of features. Among them, ResU-Net, as a 2D network, combines with a weight mechanism based on the residual block and performs better than U-Net in solving retina segmentation problems (Figure 4) [5]. Ibtehaz further improved on the residual block and designed the MultiResUnet [6]. This network has MultiRes Block modules and Res Path modules. The MultiRes Block replaces the large convolution kernels, such as 5×5 and 7×7 convolution kernels in traditional residual blocks, with continuous 3×3 convolution kernels, and maintains image size by adding 1×1 convolution kernels. This reduces the calculation parameters while ensuring feature extraction. In addition, MultiResUnet also adds Res Path in the encoder-decoder. In each skip connection, Res Path introduces residual connections to connect feature graph to decoder through a convolution chain. This ensures that spatial information lost after each encoder is reversed through deconvolution can be transmitted to the decoder, reducing the semantic differences between corresponding levels. Compared with U-Net, MultiResUnet has significantly improved on challenging datasets, including endoscopic images, skin mirror images, etc.

Networks with residual blocks also include RU-Net and R2U-Net, designed by Alom and others [7]. The authors used recurrent convolutional layers (RCLs) to replace the original function and proposed four cell structures. Including the recurrent convolutional module used in RU-Net and the recurrent + residual module used in R2U-Net. This network achieves feature accumulation according to different strides, ensuring stronger feature representation. It also outperforms traditional U-Net in segmentation of multiple image datasets.

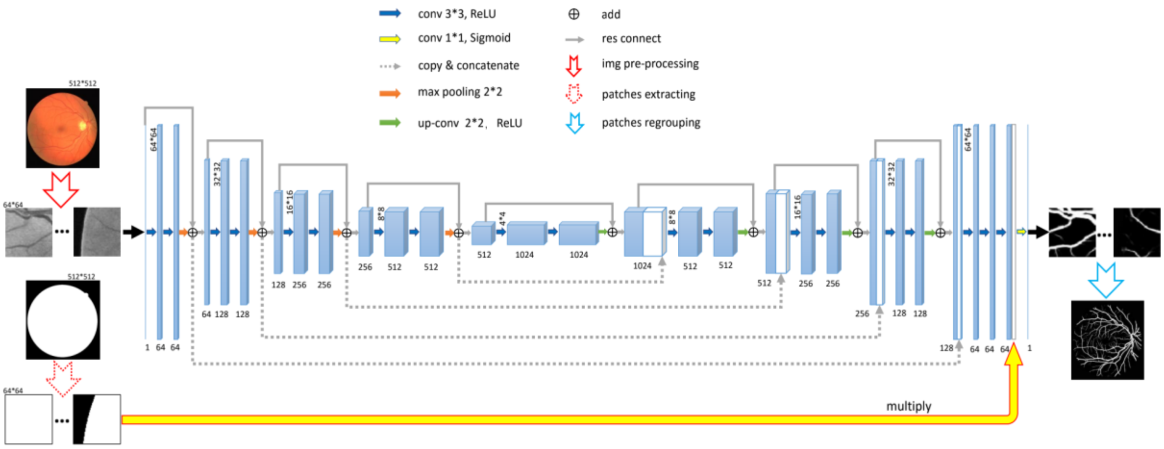


Figure 4. The architecture of ResU-Net [5].

The introduction of residual blocks greatly enhances the network's feature extraction capabilities, enriches the learnable feature quantity of the U-Net, and increases the depth and width of the network, improving its expression and accuracy. However, the large number of redundant features brought by residual blocks can cause the network to learn too many redundant features, increasing the complexity of the network, training burden, and time resource costs.

3.2. Module addition With the development of many functional modules, such as dense connection modules and attention modules, these modules have been widely used in multiple fields due to their outstanding specialty performance and excellent generalization ability. At the same time, skip connection modules play an important role in the U-Net network's semantic feature fusion. By adding and fusing modules, the U-Net network's feature learning ability can be further enhanced.

3.2.1. Increase the number of skip connections. U-Net combines shallow and deep features through skip connections, greatly improving the learning ability and the network’s segmentation accuracy. Therefore, by increasing skip connections’ number, the model can capture more semantic information and achieve better segmentation performance. U-Net++ is another U-Net architecture variant proposed by Zongwei et al., inspired by DenseNet, which enhances skip connections (Figure 5) [8]. Its structure reduces the semantic gap between corresponding layers. U-Net++ uses a dense skip connection network instead of traditional skip connections between U-Net layers. The network consists of multiple skip connection nodes and dense connections, and all feature mappings of the previous cell at the same level are received by each skip-connected cell after it. Therefore, a dense block can be thought for each layer. The semantic information loss in contraction path and expansion path is reduced by preserving feature mapping to a maximum extent through dense blocks.

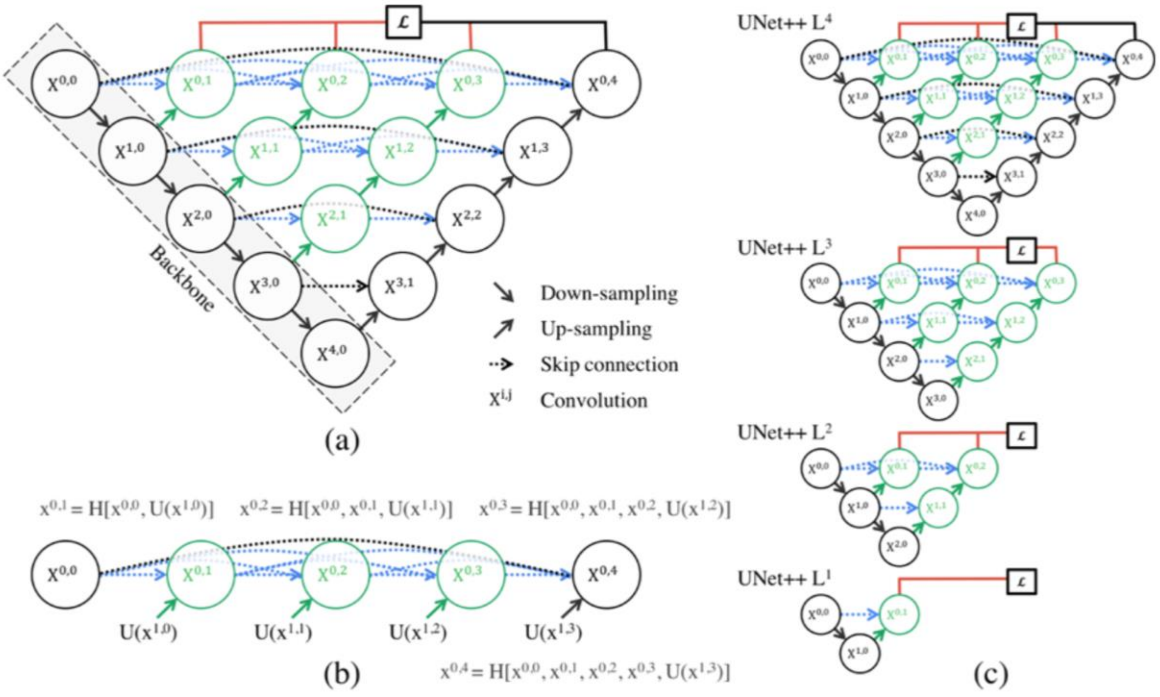

Backbone
Down-sampling
Up-sampling
Skip connection
Convolution
(a)
(b)
(c)
UNet++ L⁴
UNet++ L³
UNet++ L²
UNet++ L¹
x^{0,1} = H[x^{0,0}, U(x^{1,0})]
x^{0,2} = H[x^{0,0}, x^{0,1}, U(x^{1,1})]
x^{0,3} = H[x^{0,0}, x^{0,1}, x^{0,2}, U(x^{1,2})]
x^{0,4} = H[x^{0,0}, x^{0,1}, x^{0,2}, x^{0,3}, U(x^{1,3})]

Figure 5. The architecture of U-Net++ [8].

A similar method is U-Net3+ [9]. It is based on U-Net++ and combines smaller features of the same scale retained the decoder's features with larger ones. At the same time, the fine-grained semantics and coarse-grained semantics on the complete scale are captured. The dense connection module improves the disadvantages of residual blocks to some extent. This optimization solution not only solves the limitation of feature fusion at skip connections in U-Net itself but also greatly reduces the network parameters by pruning while preserving the maximum features of the network, ensuring learning depth and network speed. In addition to the dense connection structure, Bio-Net introduces a reverse skip connection structure on the basis of the original forward skip connection [10]. The network establishes bidirectional connections to link the encoder and decoder, and recursively implements the feature mapping between the encoder and decoder. In addition, Bio-Net does not require additional training parameters but its network performance exceeds that of traditional U-Net.

3.2.2. Strengthen feature mapping. In optimizing the skip connection process of the network, variants introduce specific functions modules to enhance feature mapping in skip connections. The common additions are the function of attention module. Attention U-Net introduces an Attention Gate module [11]. This module multiplies the coarse-grained information extracted by the network and the attention coefficients and then fuses the output with the upsampled feature map (Figure 6). This enables the network to learn images of different sizes and shapes, calculate feature importance for each pixel, and suppress invalid information while highlighting prominent features of specified targets. A similar network with a similar function is BCDU-Net proposed by Azad et al [12]. The network also adopts the idea of dense connections to let the network learn more features but also references a new extension module, LSTM module. This module controls feature transmission through gate states and preserves feature data that is significant for segmentation. The network strengthens feature mapping and effective information preservation by combining feature maps with the nonlinear function in the bidirectional Convolutional LSTM module. The network has also demonstrated its excellent network performance in several segmentation tasks.

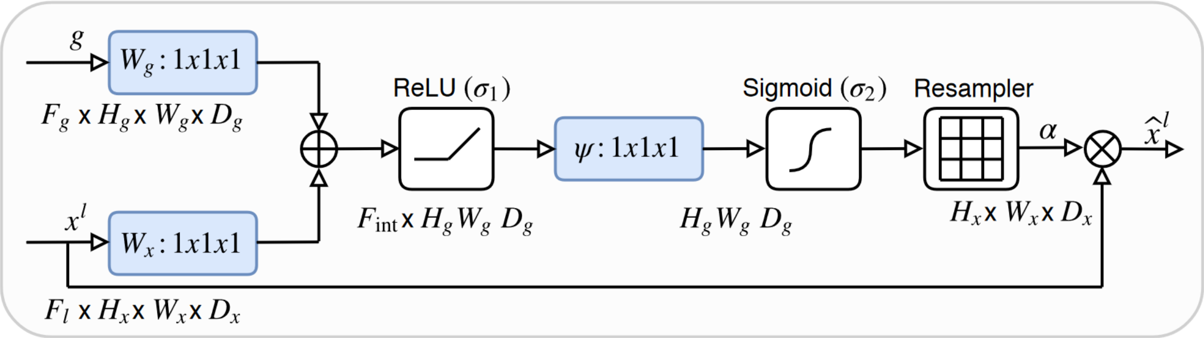


Figure 6. The architecture of Attention Gate [11].

Attention mechanism is also one of the methods to reduce network redundant features. This optimization approach simulates the process of human visual recognition of object features and focuses on enabling the network to make conditional choices to improve its accuracy. In addition, the parallel processing approach in the attention mechanism makes it not only has fewer parameters and better network performance, but also achieves a certain improvement in speed.

3.3. Cross-Network fusion

As one of the fundamental infrastructures of images semantic segmentation, the structure of U-Net has been used by researchers for feature processing of medical images. However, some U-Net variants introduce new network structures by applying other networks used in different fields to the U-Net.

3.3.1. Cascaded network. For tasks that require more processing or are more computationally demanding, a single U-Net network structure is often not enough. Therefore, DoubleU-Net was proposed (Figure 7) [13]. The module combines two U-Net structures, with the first one using VGG-19 as an encoder and both U-Nets using ASPP to capture contextual information. Due to VGG-19's lighter weight and compatibility with U-Net, the authors chose to incorporate VGG-19 into the network. Additionally, since deep networks can achieve more accurate segmentation, the authors added another U-Net to receive the results from the first network's feature output. The DoubleU-Net aims to solve challenging medical image segmentation problems, for which the authors used several datasets to test the network performance. A variety of medical imaging modalities such as colonoscopy, dermoscopy and microscopy are included and demonstrate the excellent network performance of DoubleU-Net. Similarly, the parallel U-Net was designed to better delineate the focal sites of ischemic stroke variant symptoms [14]. This network combines four 2D U-Nets and from CBV, CBF, MTT, and Tmax images extract valuable information about stroke lesion locations. Then, the U-Net probability map is used to determine the extent of the lesion at the pixel level to achieve accurate segmentation.

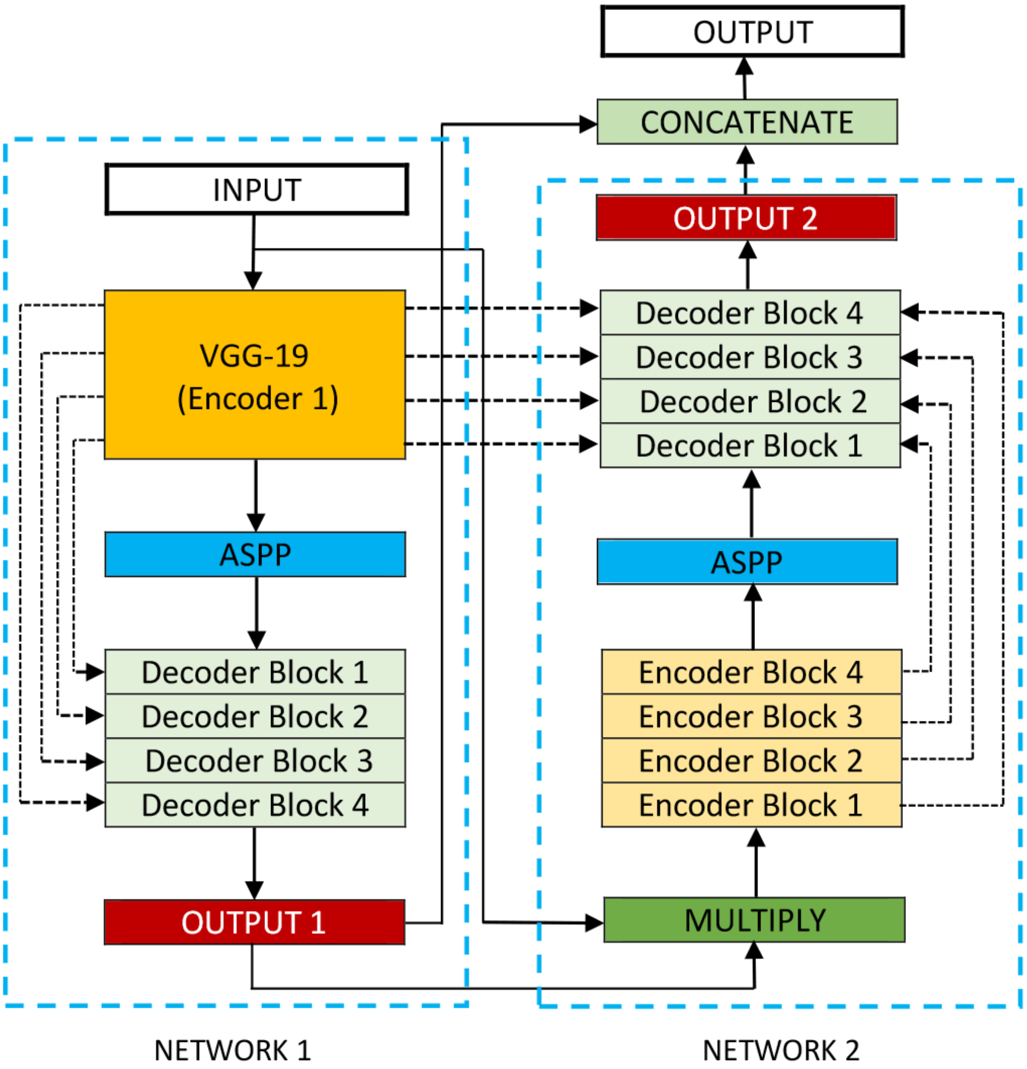


Figure 7. The architecture of DoubleU-Net [13].

3.3.2. Fusion network. In addition to using multiple parallel networks for image segmentation, specific networks with unique structures and functions can be decomposed and reconnected with U-Net's encoder-decoder parts to form a composite network composed of different network structures. TransUNet adopted the Transformer structure based on CNN features in the encoder part [15]. Since CNN can extract local details effectively and the Transformer structure can perceive global information well, Chen et al. combined them to design a fusion network. TransUNet introduces the Vision Transformer (ViT) and successfully applies it to full-size images (Figure 8). The network transforms images into sequences and then encodes global information, making effective use of shallow features to achieve high-precision image feature segmentation. Another example of network fusion is the Generative Adversarial U-Net [16]. The network addressed the issue of limited medical image labeling by introduced the Generative Adversarial Network (GAN) structure and designed a domain-impartial model that could be applied to various medical images. The network separates the U-Net generator parts and encoder, makes the overall network have the function of image generation by combining the generator and GAN. Moreover, the network optimizes the image generation process using conditional GANs and Wasserstein GANs. And it also achieves broad applicability to images with insufficient data.

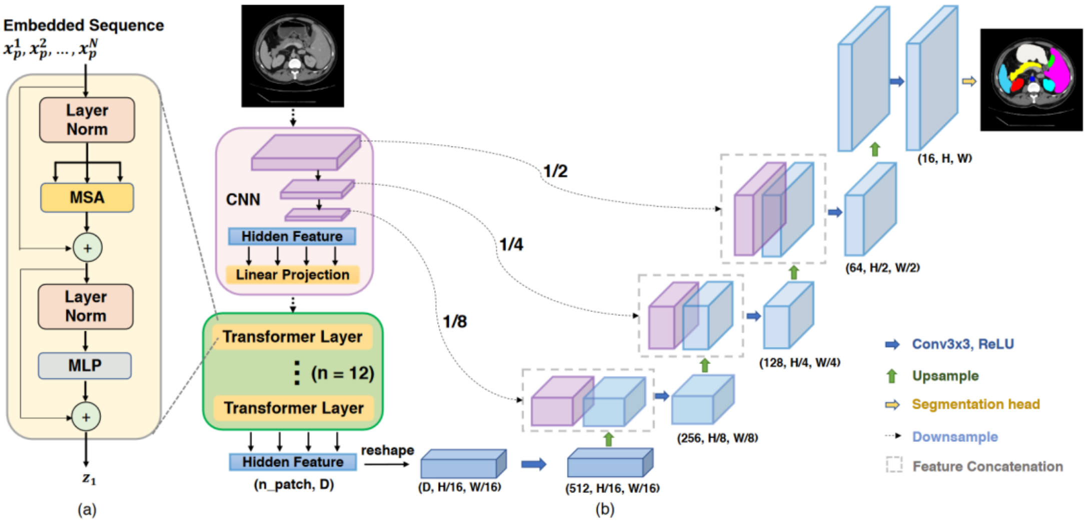


Figure 8. The architecture of TransUNet [15].

Cross-network fusion is a new optimization strategy proposed in recent years for U-Net networks. Technologies in the fields of CNN, RNN, and others are constantly innovating with the times. This optimization strategy is a practical approach to the network's flexibility and generalization. "Taking the strengths and weaknesses" into account not only solves the problems of the U-Net network itself but also extends its capabilities.

4. Conclusion

As an extremely important network in medical image segmentation, U-Net achieves its flexibility and generalization through its unique encoder-decoder structure and skip connections. Based on the demand for different medical image tasks, researchers have optimized the performance of U-Net networks in many directions, resulting in multiple U-Net variant networks. This article divides them into three categories based on the optimized structural parts: backbone improvement based on encoder-decoder structure, module addition based on skip connection structure, and cross-network fusion based on the entire network's functional structure. By discussing the network structures, the optimization strategies of U-Net variant networks are summarized.

It can be seen from the article that there is a demand for residual blocks, dense connections, and attention mechanisms in the future optimization and development of U-Net networks. These methods have an excellent effect on feature extraction and network learning capabilities. However, there is still room for development. While dense connections reduce the burden of network training to a certain extent compared to residual blocks, it still cannot achieve a balance between network training speed and performance. Therefore, how to solve the balance problem to achieve better optimization is a direction for development. Additionally, the effectiveness of the attention mechanism requires a large amount of data as a basis. At the same time, as the demand for segmentation networks in medical field increases, the attention features of some tasks will change due to factors such as time and space. Therefore, the controllability of attention mechanisms in terms of time and space is another area for optimization. In the end, multi-domain network fusion is a new direction for the development of U-Net variants. With the gradual complexity of image segmentation tasks in the medical field, the development of multifunctional composite networks will have extremely broad application prospects.

References

[1] Long, J., Shelhamer, E., & Darrell, T. Fully convolutional networks for semantic segmentation. 2015, *Computer Vision and Pattern Recognition*. 3431-3440.

[2] Ronneberger, O., Fischer, P., & Brox, T. U-net: Convolutional networks for biomedical image segmentation. 2015, *In Medical Image Computing and Computer-Assisted Intervention,* 234-241.

[3] Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., & Ronneberger, O. 3D U-Net: learning dense volumetric segmentation from sparse annotation. 2016, *In Medical Image Computing and Computer-Assisted Intervention* 424-432.

[4] Milletari, F., Navab, N., & Ahmadi, S. A. V-net: Fully convolutional neural networks for volumetric medical image segmentation. 2016, *In 3DV*. 565-571.

[5] Xiao, X., Lian, S., Luo, Z., & Li, S. Weighted res-unet for high-quality retina vessel segmentation. 2018, *In International Textile Machinery Exhibition* 327-331.

[6] Ibtehaz, N., & Rahman, M. S. MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation. 2020, *Neural networks,* **121**, 74-87.

[7] Alom, M. Z., Hasan, M., Yakopcic, C., Taha, T. M., & Asari, V. K. Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation. 2018, arXiv preprint arXiv:1802.06955.

[8] Zhou, Z., Rahman Siddiquee, M. M., Tajbakhsh, N., & Liang, J. Unet++: A nested u-net architecture for medical image segmentation. 2018, *In Deep Learning on Medical Image Analysis* 3-11.

[9] Huang, H., Lin, L., Tong, R., Hu, H., Zhang, Q., Iwamoto, Y., ... & Wu, J. Unet 3+: A full-scale connected unet for medical image segmentation. 2019 *In International Conference on Acoustics, Speech and Signal Processing* 1055-1059.

[10] Xiang, T., Zhang, C., Liu, D., Song, Y., Huang, H., & Cai, W. BiO-Net: learning recurrent bi-directional connections for encoder-decoder architecture. 2020 *In Medical Image Computing and Computer-Assisted Intervention*. 74-84.

[11] Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K., & Rueckert, D. Attention u-net: Learning where to look for the pancreas. 2018, *arXiv preprint arXiv:1804.03999.*

[12] Azad, R., Asadi-Aghbolaghi, M., Fathy, M., & Escalera, S. Bi-directional ConvLSTM U-Net with densley connected convolutions. 2019, *International conference on computer vision workshops*.

[13] Jha, D., Riegler, M. A., Johansen, D., Halvorsen, P., & Johansen, H. D. Doubleu-net: A deep convolutional neural network for medical image segmentation. *In Conference Board of the Mathematical Sciences* 558-564.

[14] Soltanpour, M., Greiner, R., Boulanger, P., & Buck, B. Ischemic stroke lesion prediction in ct perfusion scans using multiple parallel u-nets following by a pixel-level classifier. 2019 *In Biological Information and Biomedical Engineering.* 957-963.

[15] Chen, J., Lu, Y., Yu, Q., Luo, X., Adeli, E., Wang, Y., ... & Zhou, Y. Transunet: Transformers make strong encoders for medical image segmentation. 2019 *arXiv preprint arXiv:2102.04306*.

[16] Chen, X., Li, Y., Yao, L., Adeli, E., & Zhang, Y. Generative adversarial U-Net for domain-free medical image augmentation. 2021 *arXiv preprint arXiv:2101.04793*.

Cat classification based on improved ResNet50

Shipeng Sun

College of Mathematics and Statistics, Guangdong University of Technology, No. 161 Yinglong Road, 510000, Guangzhou, China

3120007004@mail2.gdut.edu.cn

Abstract. Cat species recognition holds significant potential in many fields. The primary objective of this research is to develop an automated algorithm for recognizing the presence of cats in images. The application prospects of this algorithm are diverse and include security, image search, and social media. Hence, this research has considerable practical value in various domains. In this study, we propose a cat image recognition algorithm based on the PyTorch, with ResNet50 as the foundational network architecture, and an attention mechanism (Efficient Channel Attention) integrated into the model for improved performance. We first introduced the Resnet network, and then introduced the combination of attention mechanism and Resnet in detail The proposed model achieved a 92.37% accuracy rate in classifying the 12 cat species, demonstrating its efficacy in accurately classifying and recognizing the collected images. The research conclusion of this paper has certain reference value.

Keywords: cat classification, attention, Resnet, efficient channel attention.

1. Introduction

With the rapid development of computer technology in the 21st century, artificial intelligence has pervaded various fields. Machine vision, an important branch of artificial intelligence, simulates human visual function to analyze and understand target images through feature extraction, ultimately achieving target classification and recognition. Machine vision technology has been applied to multi-font Chinese character recognition, Chinese medicine image recognition, car logo recognition algorithm research, and recently, cat species identification, which has broad application prospects in pet insurance, pet portrait, pet retail, pet health management, and other related fields. Researchers have conducted extensive research on the application of machine vision technology in cat classification and recognition.

Various classification methods have been studied, such as the eigenvalue recognition algorithm which is simple to operate, but with inadequate recognition rates [1]. The correlation coefficient recognition algorithm, which has high recognition accuracy but appears to have a narrow application range. The hierarchical classification algorithm has large calculations, and the support vector machine algorithm has high recognition rates but has difficulty in implementing large-scale training samples and requires manual selection of eigenvalues [2]. In comparison, the Convolutional Neural Network (CNN) has independent learning ability and good robustness. However, it requires a large amount of training data, and as the number of network layers deepens, the computational complexity and training model cycle increase. With the deepening of the network, the accuracy of the training set decreases, which affects the recognition effect and accuracy.

Based on the above analysis, this paper proposes a cat species recognition algorithm based on ResNet50, which fuse residual layer features and adds an attention mechanism to improve algorithm performance.

2. Mehtod

In the deep architecture of Convolutional Neural Networks (CNNs), the problem of gradient vanishing and explosion can impede training set accuracy as the network layers become increasingly deep. This issue can be resolved by employing residual networks, which allow for enhanced network performance despite increases in depth [3-4]. Specifically, a residual block structure, as depicted in Figure 1, is utilized, where ResNet includes both identity mappings and residual mappings. In this model, the image is convolved and processed through four residual modules to prevent overfitting, followed by one more convolutional layer to produce the output.

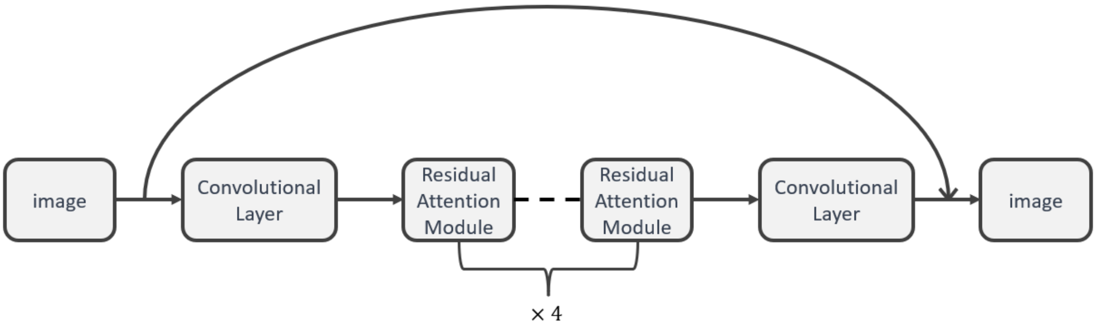


Figure 1. Main body framework diagram.

2.1. Backbone ResNet50

ResNet50 is composed of 49 convolutional layers and 1 fully connected layer. Specifically, ID BLOCK x2 in the second to fifth stages denotes two residual blocks that maintain the spatial dimensions, whereas CONV BLOCK refers to a residual block that introduces spatial scaling. Each residual block consists of three convolutional layers, yielding a cumulative count of 1 + 3 x (3+4+6+3) = 49 convolutional layers. The ResNet50 architecture expects the input images to have a size of 256 × 256 × 3, which is not compatible with the required input size of 224 × 224 × 3. To resolve this incompatibility, a preprocessing step is employed to normalize the input images and crop them to the required size [5]. Specifically, the mean value of each channel across all images in the training set is subtracted to normalize the images, and then the images are cropped to the specified size. After passing through consecutive convolutional operations in the residual blocks, the depth of the pixel matrix channels of the image's increases. The Flatten layer is then applied to transform the pixel matrix into a 2D tensor of shape batch_size × 2048, which is then fed into the fully connected layer (FC).

The softmax classifier subsequently outputs the corresponding class probabilities. ResNet50 architecture utilizes skip connections, which are implemented through shortcut connections to propagate the input across layers and add it to the output that has undergone convolution. This process facilitates the effective training of the lower layers of the network, leading to a remarkable increase in accuracy as the depth of the network increases [6-7]. The shortcut connection can be viewed as performing an equivalent identity mapping, which does not introduce additional parameters or computational complexity to the network. In this case, the model is effectively reduced to a shallow network, and the key challenge is to accurately learn the identity mapping function $H(x)=x$. Directly fitting such a latent function can pose a significant challenge. Let $H(x)$ and $F(x)$ represent the output of the residual network and the output after convolution, respectively, such that

$$H(x)=F(x)+x \tag{1}$$

$$F(x) = (\omega_3\delta(\omega_2\delta(\omega_1 x+b))) \tag{2}$$

where ω represents convolutional operations and δ denotes activation functions. The problem of learning an identity mapping function can be reformulated as the task of training a residual function F(*x*)=H(*x*)-*x* that is easy to optimize [8].

2.2. Residual attention module based on ECA attention module

Distinctive variations in physical characteristics such as color, eye shape and size, ear morphology, coat type, and body proportions are readily apparent across various feline breeds. A number of feline breeds are typified by orbicular and voluminous ocular structures, whereas others exhibit slenderer and reduced ocular configurations. Likewise, some breeds are characterized by relatively diminutive aural appendages, while others possess comparatively larger and more erect auricular attributes. Variations in coat texture and length are also evident among feline breeds, with some breeds displaying velvety and extensive pelage, while others exhibit compact and abbreviated fur coverage. Additionally, certain breeds are distinguished by squat and plump body morphologies, while others are identified by elongated and slender body proportions [9]. This model improves upon the ResNet50 architecture by integrating an attention mechanism, which enhances the model's ability to selectively attend to highly discriminative features, thereby mitigating the effects of confounding information.

The Efficient Channel Attention Module (ECA) is an improvement over SENet, as the dimensionality reduction in SENet can have side effects on the channel attention mechanism, and capturing all interdependencies between channels is not always necessary or efficient (Figure 2). The ECA module uses a 1x1 convolutional layer directly after the global average pooling layer, and removes the fully connected layer. This module avoids dimension reduction and effectively captures cross-channel interactions. Additionally, ECA model achieves good results with few parameters. The Efficient Channel Attention Module is a channel attention mechanism frequently employed in visual models. Its plug-and-play design allows for simple integration into existing architectures, enabling it to enhance the channel features of input feature maps without altering their size. Consequently, the ECA module serves to strengthen channel features, while maintaining the original size of the input feature map. The ECA-Net addresses the potential drawbacks of channel attention prediction caused by dimension reduction in the SENet, and the inefficiencies of capturing dependencies among all channels.

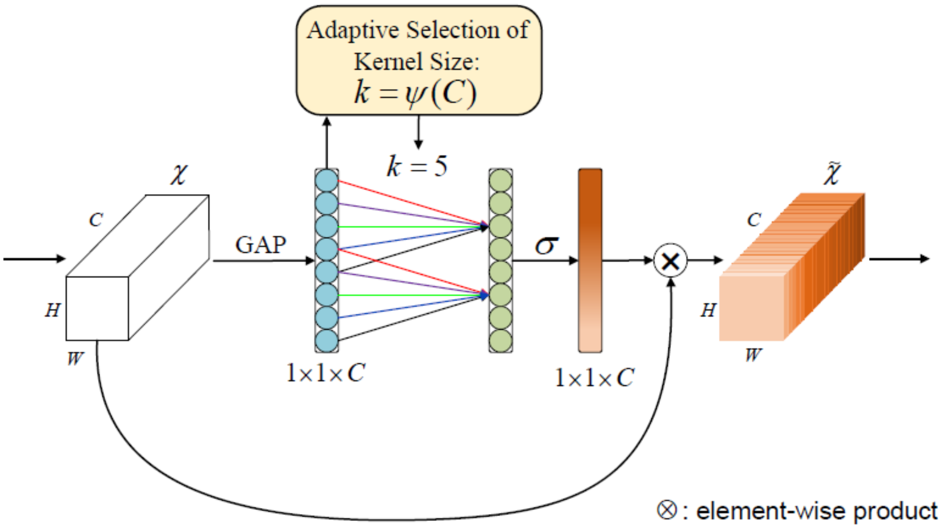


Figure 2. ECA model.

The ECA-Net achieves this by modifying the SENet architecture, replacing the fully connected layer (FC) responsible for learning channel attention information with 1x1 convolutional layers. This modification avoids the reduction of the channel dimension and reduces the overall number of parameters required for learning channel attention information [10-11]. Compared to the FC layer, the

1x1 convolutional layer has fewer parameters, making it a more efficient alternative. The ECA model follows a process in which an input feature map with dimensions of H * W * C undergoes spatial feature compression using global average pooling (GAP) on the spatial dimensions, yielding a feature map of 1 * 1 * C. Subsequently, a 1 * 1 convolution is employed to capture inter-channel dependencies and learn the relative importance of different channels, producing an output feature map with dimensions of 1 * 1 * C. Finally, a channel attention mechanism is integrated, whereby the channel attention feature map (1 * 1 * C) and the original input feature map (H * W * C) are element-wise multiplied on a channel-by-channel basis to produce a feature map with channel attention (Figure 3).

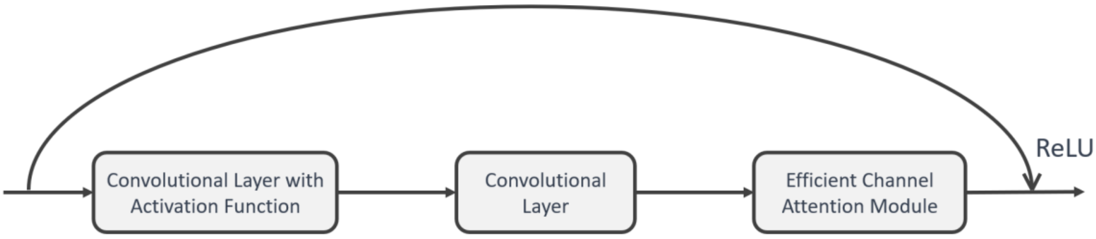

Convolutional Layer with Activation Function
Convolutional Layer
Efficient Channel Attention Module
ReLU

Figure 3. Residual block based on ECA attention.

3. Result

The aim of this experiment is to classify images of different cat breeds using deep learning algorithms. The dataset contains several thousand cat images, encompassing 12 different cat breeds. ResNet50 was selected as the base model and the ECA attention mechanism was added to improve the classification performance. The model was adjusted and optimized for hyperparameters during the training process.

3.1. Data set introduction

This paper employs the dataset provided by the Paddle platform's Cat Twelve-Classification Competition. The training dataset consists of 2,160 images of cats, divided into 12 categories. The testing dataset comprises 240 images of cats, without any annotation information (Figure 4).

In this study, the dataset was initially shuffled and partitioned into three distinct subsets: a training set, which accounted for 60% of the total dataset, a validation set (20%), and a testing set (20%). Uniform preprocessing techniques were implemented across all subsets to ensure consistency and accuracy of the validation and testing results. Firstly, the images were cropped to dimensions of 224x224 to ensure their compatibility with the input size of the model. The pixel values of the images were standardized, and finally, the dataset was divided into batches of size "32" to facilitate the training of the model in subsequent stages.

Figure 4. Data set diagram.

3.2. Hardware and software platform

The experimental hardware configuration used in this study consisted of a 64-bit Windows 10 operating system, a 2.60 GHz Intel i7 CPU, and an NVIDIA GeForce RTX 2060 graphics card based on the Pascal architecture. For software, PyCharm 2022.3.3 was used as the development platform, with the PyTorch open-source deep learning framework selected as the programming framework, with a version of 1.10.2. The program was designed using Python 3.6.13.

The loss curve demonstrates that with an increase in the number of training iterations, the loss function progressively decreases until it converges to a minimum value (Figure 5). The experimental results demonstrate that the proposed model has high classification performance and can effectively classify different cat breeds. Ultimately, the accuracy on the test set reaches 92.37%. Figure 6 shows the classification results for different cats. The numbers in "[]" are confidence (%), followed by the classification results for cats.

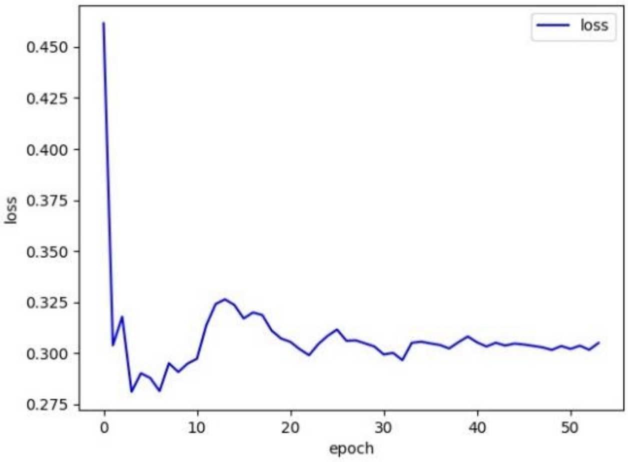


Figure 5. loss curve.

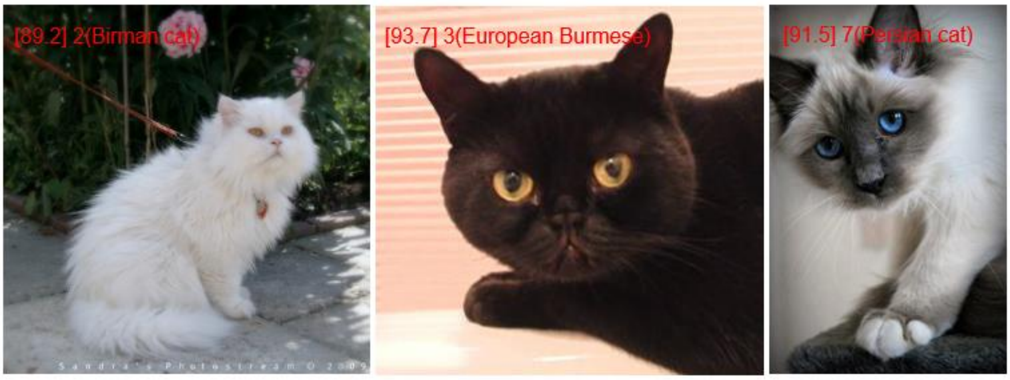


Figure 6. Classification result.

This study presents a novel deep learning network model based on ResNet50 for accurate recognition and classification of cat images, using the latest advancements in image classification techniques. The residual blocks incorporated in the model address the issue of network degradation, ensuring that the model's performance does not deteriorate with the increase in network depth. This proposed model outperforms the existing models in terms of its increased depth, faster convergence rate, higher precision, and superior generalizability. The feasibility of the proposed model has been demonstrated in practical applications of cat image classification, providing an effective solution for animal recognition.

4. Conclusion

The present study introduces an ECA attention mechanism to the ResNet50 architecture, resulting in a faster convergence rate, shorter training time, and improved performance. The residual network effectively addresses the issue of difficult training in deep networks, leading to a high accuracy rate of

92.37%. This model not only demonstrates its potential in cat breed image classification but also holds applicability in classifying other animals such as dogs. Although the ResNet50 model achieved the expected classification performance, the limited size of the dataset used in this study necessitates future work with a larger dataset to further improve the recognition accuracy.

References

[1] Edgar Solomonik, Grey Ballard. A Communication-Avoiding Parallel Algorithm for the Symmetric Eigenvalue Problem. 2017 *Sym. Para. Alg. Arch*. 111–121.

[2] An Zeng, Qi-Gang Gao, and Dan Pan. A global unsupervised data discretization algorithm based on collective correlation coefficient. 2011 *Conf. Ind. Eng. Appl. Intel. Sys.* 146–155.

[3] Susan Dumais and Hao Chen. Hierarchical classification of Web content. 2000 *Conf. Res. Deve. Infor. Ret*. 256–263.

[4] Glenn Fung and Olvi L. Mangasarian. Proximal support vector machine classifiers. 2001 *Conf. Knowl. Disc. Data Min.*, 77–86.

[5] Nestler E G, Osqui M M, Bernstein J G. Convolutional Neural Network, 2017, *Wire. Net.*, **201 (74)**,137-151.

[6] Weijie Liu, Weiwei Chen, and Xinmiao Dai. Capsule Embedded ResNet for Image Classification. 2021 *Conf. Com. Sci. Arti. Intel.*, 143–149.

[7] Kaiming He, Xiangyu, and Jian Sun. Deep residual learning for image recognition. 2016 *Computer Vision and Pattern Recognition,* 1-10.

[8] Wu Z, Shen C, Hengel A. Wider or Deeper: Revisiting the ResNet Model for Visual Recognition. Pattern Recognition, 2016 *Comp. Vis. Pat. Rec.*,1-12.

[9] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. 2018, *Comp. Vis. Pat. Rec*, 201-211.

[10] Santosh Kumar Mishra, Gaurav Rai, Sriparna Saha, and Pushpak Bhattacharyya. 2021. *Effic. Cha. Att. En.*, **21 3**, Article 49, 17

[11] Chen P. Efficient Channel Allocation Tree Generation for Data Broadcasting in a Mobile Computing Environment. 2003, *Wire. Net.,* 200-213.

Balance of rights in the protection of users' data interests in the era of big data

Chuanxiang Cong

Computer Science, dalhousie university, Nova Scotia, Halifax, Canada, B3H4R2

ch898183@dal.ca

Abstract. The protection of user data interests is of utmost significance in this day and age, when everyone's private life details and possessions are being digitised and stored in the cloud. Data now rules the world. This paper reviews the relevant literature in order to study the balance of rights in the protection of user data interests in the era of big data. It then suggests various countermeasures for the problem that it identifies. According to the findings of this study, the key factors contributing to the imbalance in the protection of user data rights are the progression of science and technology, the increase in the value of user data, and a lack of awareness regarding the protection of user personal data rights.

Keywords: data interest, big data, personal property, internet users.

1. Introduction

In the era of big data, with the continuous evolution of mobile terminal equipment technology and the development of mobile Internet applications, smart mobile terminals, such as smartphones and tablet computers, are gaining in popularity, and the number of Internet users worldwide continues to rise. Figure 1 displays that the number of Internet users in the globe increased by 4% between January 2021 and January 2022, reaching 4.95 billion.

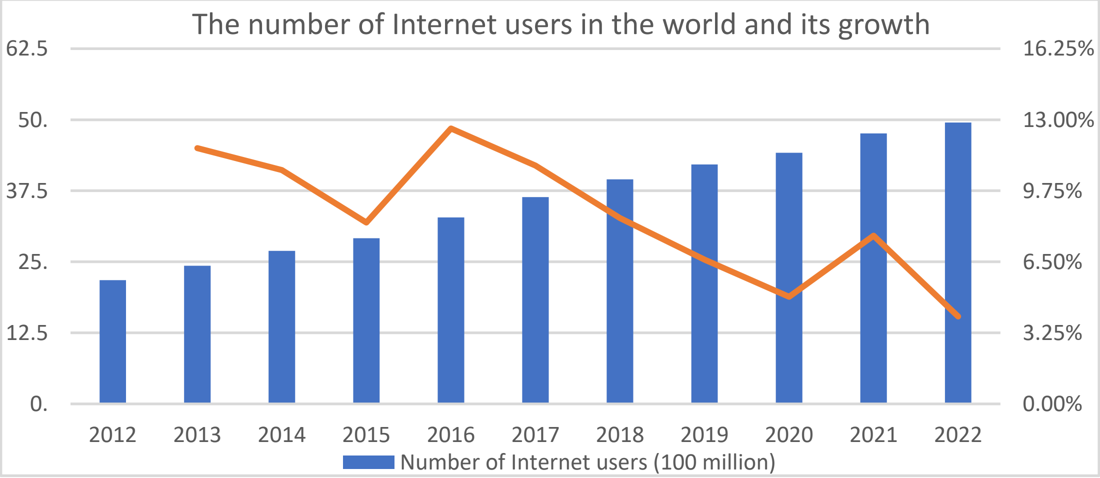


Figure 1. The number of internet users in the world and its growth [1].

A large number of users indicates that, in the era of big data, there are numerous types and quantities of user data. Moreover, the quantity of data left by Internet users is enormous. It is not difficult to conclude that, in the era of big data that we currently inhabit, the total quantity of user data has reached an astounding level.

Everyday network use necessitates the authorization of personal data. This can be reflected in a variety of privacy clauses in the application and a variety of login agreements. Personal information is utilised in virtually every aspect of daily existence. Through a review of relevant research, this paper examines the balance of rights in the preservation of user data interests in the era of big data and proposes a number of countermeasures. The paper intends to offer some helpful suggestions for this research field.

2. Performance of the imbalance of user data rights protection in the era of big data

The disparity in user data protection manifests itself in three ways: -It is the disclosure of fundamental personal information about the user. It refers to the personal name, age, and other demographic information involved in the dissemination of information, as well as the account information registered and utilised on the Internet. The second is the disclosure of personal behaviour data pertaining to users. In the process of information use and dissemination, real-time data such as users' clicking, forwarding, browsing, and sharing, as well as clothing, food, housing, and transportation, are at risk of being recorded, collected, used, and stored; these data reflect personal network communication behavior, information browsing behavior, entertainment consumption behavior, and other real behaviours. The third risk is the disclosure of an individual's predicted preferences. Using big data technology, personal preference information is summarised and predicted through the use of relational data, fuzzy calculation, and other methods in order to achieve the objectives of precise marketing, immediate analysis, and intelligent decision-making [2].

3. Reasons for the imbalance of user data rights protection in the era of big data

3.1. The value of user data has been continuously improved

In the current era of big data, any user data may become valuable and provide people with benefits. However, when criminals steal user data through improper means and misuse it, it will cause irreparable damage to the body and mind of the individual, as well as threaten the security of personal property and even the stability of the entire society. With the development of online media, the use value and exchange value of user information data have significantly increased in the era of big data, while the disclosure of user privacy information has become more lucrative [3].

3.2. Rapid development of technology in the era of big data

Rapid advancements in information technology have unquestionably made people's lives more convenient. The increasing sophistication of information technology has always been a double-edged sword for the general public. If the user's mind is impure, information technology will become a Damocles' sword dangling from the ceiling. Increasingly sophisticated network programmes and covert systems make it easier to capture user data in the background. Under big data, the background cloud storage is also practical for querying user behaviour data. Moreover, as a result of the above-mentioned extensively utilised characteristics of user data, user data has become more transparent and the protection of personal privacy data has become increasingly difficult.

3.3. Weak awareness of data rights among users

In the Internet age, the protection of others' privacy is exceedingly rare, and the awareness of network congestion's impact on the protection of personal privacy data is still quite low, so people cannot pay sufficient attention to privacy violations in everyday life. First, individuals do not know how to safeguard their own and others' privacy. Some individuals believe that privacy violations are common and have not been punished, so they invade the privacy of others. In addition, the public is unaware of how to safeguard their privacy from intrusion. When individuals' privacy is violated, they do not know how to halt it. They cannot ensure that their privacy will not be violated through formal legal channels, which contributes to the frequent and intractable theft of the public's personal information. People's lack of awareness regarding personal privacy protection not only contributes to the development of China's legal system, but also encourages the recurrence of violations, making it impossible to guarantee their safety.

3.4. The lag of privacy protection regulations leads to inadequate supervision

In the context of big data technology, the absence of privacy protection regulations makes data utilisation supervision fraught with hidden dangers, resulting in a privacy conundrum. Existing laws and regulations regarding network privacy protection are merely formalities, are overly general, and lack practical application. The majority of them protect only personally identifiable information, and their protection scope is limited. The original privacy protection principles, including "precise purpose, prior consent, use restrictions, etc., are challenging to implement in the context of big data technology. The legal system for the preservation and use of online personal data is imperfect, resulting in a lack of robust countermeasures for data use oversight [4].

4. Countermeasures to ensure the balance of rights in the protection of user data interests in the era of big data

4.1. Constructing relevant laws to ensure users' data interests

It is of the utmost importance to strengthen the legislation governing the preservation of personal information. Within the framework of extant laws and regulations, the first objective is to strengthen criminal law provisions. Expand the scope of the crime in question. Except for the personnel of state organs or relevant divisions, it is proposed that ordinary individuals with criminal responsibility be included, so that the subject of the crime is the general subject. As long as the crime continues to be committed, all severe offences will be punished. Clear the judgement of serious circumstances, serious circumstances will only exist when the act of unlawfully providing or obtaining other people's information is sufficient to pose a serious threat or cause serious losses to the personal personality rights and property rights of citizens. To assess the severity, it is necessary to consider four factors: the quantity of information, the significance of information, the influence of information, and the timing of behaviour [5]. The second objective is to enhance the tort liability law. It is proposed to change the rules of proof so that the infringer must prove that he is not at fault, and if he cannot do so, he is presumed to be at fault, thereby reducing the burden of evidence for the infringed. Providing punitive damages; If the infringer sells or illegally provides personal information to others, or steals or purchases, collects, stores, processes or uses personal information illegally, it is difficult to determine the infringer's loss, and if the infringer gains from it, compensation shall be made at twice the profit; thereby expanding the infringer's compensation range. Third, the Personal Information Protection Act should be promulgated as soon as feasible. Numerous departments involved in the protection of personal information should actively promote legislation or establish special institutions to promote legislation, introduce a special and unified law on the protection of personal information as soon as possible, standardise and coordinate the implementation of other relevant laws and regulations, and establish a comprehensive protection system for personal information security.

4.2. Constructing a powerful multi-subject supervision system

In the current era of big data, all subjects should do their part to safeguard the security of users' data rights and interests.

First, the government should do a good job with the design of the user data system repository, establish and develop a prevention and control mechanism for user data privacy that is compatible with the current era of big data, and play its leadership role to the fullest. Both central and local government agencies should increase their oversight of privacy protection in order to prevent the disclosure of personal information. The government should establish a mechanism for online notification and feedback regarding abrupt privacy incidents. According to the division of responsibilities, government departments must respond swiftly to sudden, significant privacy breaches or leaks and implement efficient coordination and emergency response mechanisms across departments, regions, and systems [6]. Multiple departments collaborate to address the privacy disputes of citizens caused by unethical businesses or institutions, to closely monitor the progress of significant issues, to continuously report the staged progress of investigations, and to promptly address a variety of privacy concerns. Expose the punishment results to companies or institutions with inadequate privacy protection, and effectively improve the service quality and work efficacy of online politics.

As a second step, it is important to promote the development of a multi-agent coordination and linkage mechanism for privacy protection. Multiple stakeholders, including the government, data corporations, and the general public, are involved in protecting individual privacy in the context of big data technology. Consequently, the government should direct multiple stakeholders to participate in the development of a collaborative governance linkage mechanism for privacy protection. Government departments should establish a linkage mechanism for privacy protection with data companies and other relevant institutions, share information, accept responsibility, collaborate, and coordinate services. At the same time, individuals are encouraged to actively contribute to the development of an interactive privacy protection mechanism. The operation of an effective multi-agent linkage mechanism will aid in the formation of a joint privacy protection force and reduce the likelihood of public privacy disclosure. Moreover, it is significant to construct a public privacy disclosure complaint platform and enhance the network complaint, reporting, and monitoring platform construction and privacy problem resolution service mechanism. Government departments should incorporate citizens' privacy protection into the government network supervision system, combine the specific requirements and application characteristics of privacy protection, and construct a public complaint reporting platform for privacy leakage in collaboration with businesses. Improve the privacy protection expression mechanism by utilising a stable and dependable platform for reporting complaints. The government should take the initiative to solve problems for the people, implement strict information supervision and a reward-and-punishment system for related industries that have customers' personal information, and end the bad behaviour of related industries or companies that disclose public privacy [7].

4.3. Strengthening the protection of users' data rights by technical means

In the age of big data, flexible technical means are an essential supplement to government oversight. The advancement of science and technology can also bolster the protection of user data rights to some extent. First, institutions need to increase capital investments in science and technology. Countries and businesses must increase capital investment in research and development of essential technologies for big data security, increase the proportion of capital investment allocated to research and development, or establish dedicated research funds. Encourage the research and development and innovation of personal information security technology, ensure information security at the technical level, enhance the quality of China's big data security technology products, and seize the opportunity to develop security technologies based on big data. The second goal is to enhance technical means. In the era of big data, a significant amount of individuals' personal information is stored and transmitted over computer networks. Technical means are the most effective method to close both human and technological loopholes [7]. It is necessary to strengthen the research, development, application, and promotion of new products and technologies, to continuously improve the performance of information system security equipment such as the firewall, intrusion detection system, anti-virus system, and authentication system, and to adopt technical means such as access filtering, dynamic password protection, login IP restriction, and network attack tracking method to enhance the access and audit functions of applications. The third step is to bolster technical specifications. Encrypt and safeguard the sensitive and vital data, and restrict access and viewing to only those who have been granted identity authorization or who have decrypted the data. Simultaneously, the system of multi-person management of important and key information is stipulated, and the authority of personal information holders is restricted, so that a single person cannot master all information and relevant personnel at each level can only master the corresponding limited information.

5. Conclusion

To balance the rights in the protection of user data interests in the era of big data, it is necessary to understand the characteristics of user data, which are numerous and widely used, leading to an imbalance in the protection of user data rights and interests, which are embodied in the basic data of users, personal behaviours of users, and user preferences. Through research, this paper concludes that the primary causes of the imbalance in the protection of user data rights are the advancement of science and technology, the increase in the value of user data, and the lack of awareness regarding user personal data rights protection. In this regard, if people want to better safeguard the rights and interests of users' data in the era of big data, the government must take the lead in macro-control, establish and strengthen relevant institutions, and implement and strengthen relevant legal systems. The second step is for all social disciplines to develop their own multi-subject supervision mechanism. Lastly, it is necessary to give priority to the application of science and technology; on the one hand, it might as well to use scientific and technological means to improve the security of user data. On the other hand, it is also a good way to break criminals' technical means and prevent user data leakage.

The literature chosen for this paper cannot encompass all ages and is insufficiently exhaustive, and this research methodology is immature. In addition, the contemporary data era is still evolving rapidly, and the data security industry will continue to improve.

References

[1] Consulting Research Report on Industry Competition and Investment Strategy in cmnet from 2022 to 2028, 2022. https://zhuanlan.zhihu.com/p/536700394

[2] The realistic dilemma and path choice of personal information protection in the era of big data, Journal of information, (12),155-159+154, 2019.

[3] User data: as privacy and as assets? -legal and ethical considerations of personal data protection, Editorial friend, (10), 74-79, 2019.

[4] On the disclosure of citizens' privacy under the network media environment-taking the disclosure of Facebook user data as an example, Research on Communication Power, (15), 221-222, 2020.

[5] Thinking about privacy dilemma facing big data technology, Jianghan Forum (08),65-70, 2020.

[6] Troy Segal, What is Big Data? Definition, How it works and uses, 2022. https://www.investopedia.com/terms/b/big-data.asp

[7] Tankard, C. Big Data Security, Network Security, 5-8, 2012.

Comparison and analysis of DQN performance with different hyperparameters

Zhuoxian Huang[1,*,†], **Jiayi Ou**[2, †], **Ming Wang**[3,†]

[1]College of Engineering and Physical Sciences, University of Birmingham, Birmingham, B15 2TT, UK
[2]Computer science and technology, Nanjing University of Aeronautics and Astronautics, Nanjing, 211106, China
[3]School of artificial Intelligence, Hebei University of Technology,Tianjin, 300131, China.

zxh131@student.bham.ac.uk
[†]All authors contributed equally.

Abstract. Deep Q-learning Network (DQN) is an algorithm that combines Q-learning and deep neural network, its model can adopt high-dimensional input and low-dimensional output. As a deep reinforcement learning algorithm proposed ten years ago, its performance on some Atari games has surpassed all previous algorithms, even some human experts, which fully reflects DQN's high research value. The tuning of hyperparameters is crucial for any algorithm, especially for those with strong performance. The same algorithm can produce completely different results when using different sets of hyperparameters, and suitable values can considerably improve the algorithm. Based on the DQN we implement, we test on number of episodes, size of replay buffer, gamma, learning rate and batch size with different values. In each round of experiments, except for the target hyperparameter, all others use default values, and we recorded the impact of these changes on training performance. The result indicates that as the number of episodes continues to increase, the performance improves steadily and degressively. The same conclusion is also applicable to the size of replay buffer, while other hyperparameters need to be given values to have optimal performance.

Keywords: DQN, performance, hyperparameters, comparison.

1. Introduction

Deep Q-Learning Network (DQN) is a deep reinforcement learning algorithm that combines Q-learning and deep neural network [1]. Since DQN algorithm can adopt high-dimensional state inputs and produce low-dimensional action outputs, it is frequently used to perform human-level control and even better than human, a common example is playing Atari games [2-3]. To optimize performance, some variants of DQN have been developed in the past ten years. However, appropriate tuning of hyperparameters is necessary for any variant, since hyperparameters control the actions of training algorithm directly and significantly affect the performance of deep RL models. At present, the relevant research is insufficient, so we conduct the experiment.

In this research, we compare the results from training with different hyperparameters such as number of episodes, size of replay buffer and gamma, then analyze how they affect the performance of the algorithm in the environment of OpenAI Gym's CartPole-v1. CartPole-v1 is one of the most classic environments for reinforcement learning, it has maximum scores of 500 instead of 200 from the older version and only has simple actions to take which is informative for producing intuitive training results.

2. DQN

DQN is a reinforcement learning algorithm that combines Q-learning and deep neural network. As an improved version of Q-learning, DQN uses Q function *Q(state, action* | θ) which is designed to approximate real *Q(state, action)*, but originally it suffered from problems such as training is unstable and cumulative reward does not converge when deep neural network is applied [4]. Thus, the concept of using experience replay to train the agent was proposed. When training neural network without experience replay, it is usually assumed that data are distributed independently and identically, but data have strong correlation, instability of neural network can occur if these they are used for sequential training. However, experience replay can break the correlation between data effectively, as it allows agent to store learned data into a database with certain capacity and train its neural network by randomly taking samples from database.

The DQN updates the Q-function using the Bellman Equation:

$$y_k = r_k + \gamma \cdot Q' \left(s_k, \pi' \left(s_k, a_k \middle| \delta^{Q'} \right) \right) \tag{1}$$

Moreover, soft update is also applied to our DQN algorithm to ensure that the target network updates for every episode. More specifically, target network's parameter θ_i^- uses current network's parameter θ_i to update according to the following equation:

$$\theta_{i+1}^- = (1 - \varepsilon) \cdot \theta_i^- + \varepsilon \cdot \theta_i \tag{2}$$

which can be simplified to:

$$\theta_{i+1}^- = \theta_i^- + \varepsilon \cdot (\theta_i - \theta_i^-) \tag{3}$$

where $0 < \varepsilon \ll 1$. By using soft update, DQN algorithm keeps stabilized, even though the target network updates for every episode. Similarly, as the soft update interval ε decreases, the stability increases, the speed of convergence decreases. Thus, an appropriate soft update interval ε makes training not only more stable but also faster. Our team sets $\varepsilon = 0.005$ as default value.

3. Experiments

3.1. Details

To conduct the experiment, we implemented a DQN algorithm and used CartPole-v1 from OpenAI Gym 0.15.7 as training environment. Compared to CartPole-v0, CartPole-v1 has higher maximum scores, which offers space for improved algorithms to present. Furthermore, we chose Python 3.8 and PyTorch 1.13.1 to build code because of better stability. CartPole is shown in Figure 1.

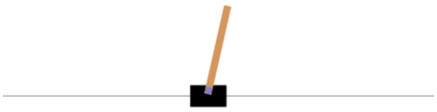


Figure 1. CartPole from Atari.

3.2. Performance comparison and analysis

3.2.1. Number of episodes. The number of episodes is a hyperparameter that determines how many times the DQN agent plays the game to train itself. Each episode consists of a sequence of states, actions, and rewards, and always ends with a terminal state. In our DQN algorithm, an episode ends if the pole falls, the cart reaches the edge, or the agent achieves a score of 500.

In supervised learning, a potential problem is overfitting, which occurs when a model learns the training data too well and performs poorly on new, unseen data. In deep reinforcement learning, overfitting can be a problem if the training environment is significantly different from the testing environment. However, since both environments in CartPole-v1 are the same, DQN algorithm should be well-fitted to the training data so that it can excel in this game [5].

In our experimental setup, we undertook the task of exploring the relationship between episodes and reward. Figure 2 depicts the outcome of this analysis, which also suggests that in the early stages of episode progression, the reward grows at a rapid pace and eventually stabilizes at later stages [6-7]. This pattern can be attributed to the fact that our network has been designed to learn from past experiences and continually improve its actions based on them.

As the network interacts with the environment and explores different scenarios, it gains a deeper understanding of the actions that lead to higher rewards and those that should be avoided. During the initial stages of training, the network may not have acquired sufficient knowledge to determine the optimal actions in every situation, which could result in lower rewards. However, as the network continues to learn and refine its actions and policy, its performance gradually improves, leading to higher rewards over time.

In summary, this experiment demonstrates that the network's ability to learn and adapt is a critical factor in maximizing the rewards it can achieve. By continuing to explore and refine its actions, the network can achieve optimal performance in a given environment.

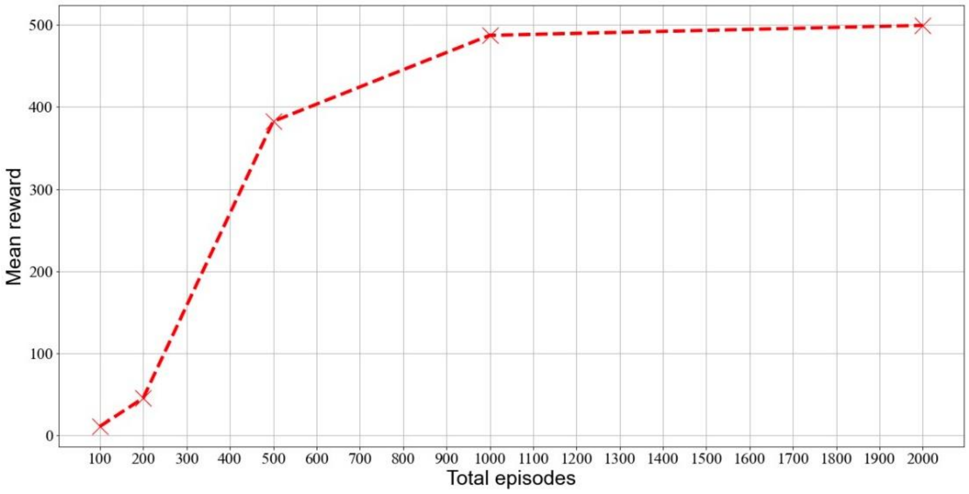


Figure 2. Mean Reward based on Total Number of Episodes.

3.2.2. Size of replay buffer. The replay buffer in a DQN network serves as a memory storage to store experience data generated from the agent's interaction with the environment. The replay buffer is then used to randomly select a subset of the stored experience data for training the DQN network. The primary functions of the replay buffer include reducing data correlation, improving data utilization efficiency, increasing sample diversity, relenting overfitting, and so on.

One of the primary advantages of using a replay buffer is that it reduces data correlation by removing temporal dependencies between consecutive samples. This allows for better utilization of data and prevents the DQN network from becoming biased towards certain types of experiences. Additionally, the replay buffer increases the diversity of samples by randomly selecting experiences, which helps prevent the DQN network from being stuck in local optima.

Another advantage of using a replay buffer is that it prevents overfitting by training the network on a random subset of the stored experience data rather than the entire dataset. This improves the generalization ability of the network and prevents it from memorizing the training data.

At the beginning, we set the size of replay buffer to 1000, and the agent didn't perform well. As a result, we started to train the agent with larger replay buffer such as 5000, 10000, 20000, and 40000. Mean rewards for all training were recorded and used for comparison.

As shown in Figure 3, the reward increases as the replay buffer increases because more historical experience is stored to learn, and the agent can choose actions more accurately in a certain state to obtain greater rewards. However, when the replay buffer reaches a certain size, the reward tends to stabilize. This is because the historical experience in the replay buffer is sufficient enough for the agent to learn the environment, in which case, the learning effect on new experience is no longer significant. At this time, increasing the replay buffer does not significantly affect the network's performance. An excessively large replay buffer will also bring storage and computational problems, leading to lower efficiency during training [8].

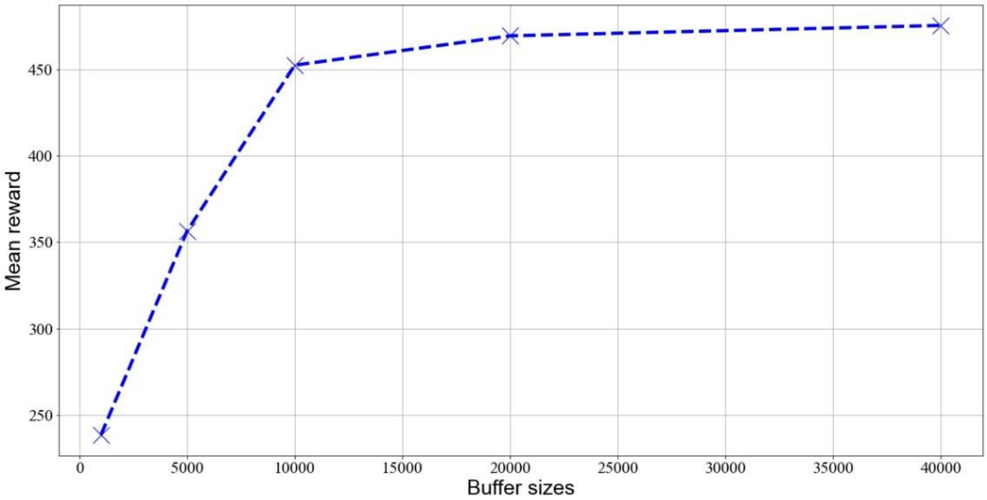


Figure 3. Mean Reward based on Buffer Size.

3.2.3. Gamma. In a DQN network, the γ value is a crucial hyperparameter that determines the trade-off between immediate and future rewards when validating an agent's performance. A higher γ value emphasizes future rewards, while a lower value emphasizes immediate rewards. Specifically, the γ value serves to discount future rewards, allowing the agent to balance the importance of immediate and short-term rewards against long-term rewards.

Typically, the γ value ranges from 0 to 1. When the γ value is close to 1, the agent gives greater weight to future rewards, which can lead to long-term planning and potentially better performance over the course of many steps. On the other hand, when the γ value is close to 0, the agent focuses more on immediate rewards, which may lead to a more reactive and opportunistic strategy. The discounted reward can be calculated by the following equation:

$$G = \sum_{k=0}^{\gamma^k} R_{t+k} \tag{4}$$

It is important to select an appropriate γ value based on the problem, as the optimal value may vary depending on the specific task or environment. Additionally, as the γ value increases, the agent may shift from being too focused on immediate rewards to being overly focused on future rewards, which may negatively impact training performance. Therefore, finding the right balance between immediate and long-term rewards is key to achieving optimal performance in a DQN network [9].

As depicted in figure 4, as the γ value increases, the mean reward generally increases, and the maximum reward is achieved at γ=0.97. However, when the γ value is too small, the agent may focus too much on immediate rewards, resulting in a short-sighted strategy that easily falls into local optimal solutions. As γ increases, the importance of long-term rewards is better accounted for, leading to a corresponding increase in the reward. However, there is a point where excessive emphasis on future rewards due to the large discount factor can lead to excessive exploration, resulting in poorer training performance [10]. Therefore, as the γ value continues to increase beyond this point, the reward will rapidly decrease. This experiment shows the importance of carefully selecting an appropriate γ value for the specific problem to achieve optimal training performance.

In terms of the magnitude of change, when the value of γ increases in the two intervals *[0.9, 0.91]* and *[0.93, 0.97]*, the reward increases very quickly. When the γ value is between *0.91* and *0.93*, although the reward also shows an increasing trend, the magnitude of increase is obviously not as large as in the other two intervals. However, when γ changes in the interval *[0.97, 1.0]*, the reward declines rapidly at an equally fast rate. The experimental results in the interval *[0.91, 093]* should not differ so much from those in the other two intervals theoretically, suggesting that this may be related to the specific task environment.

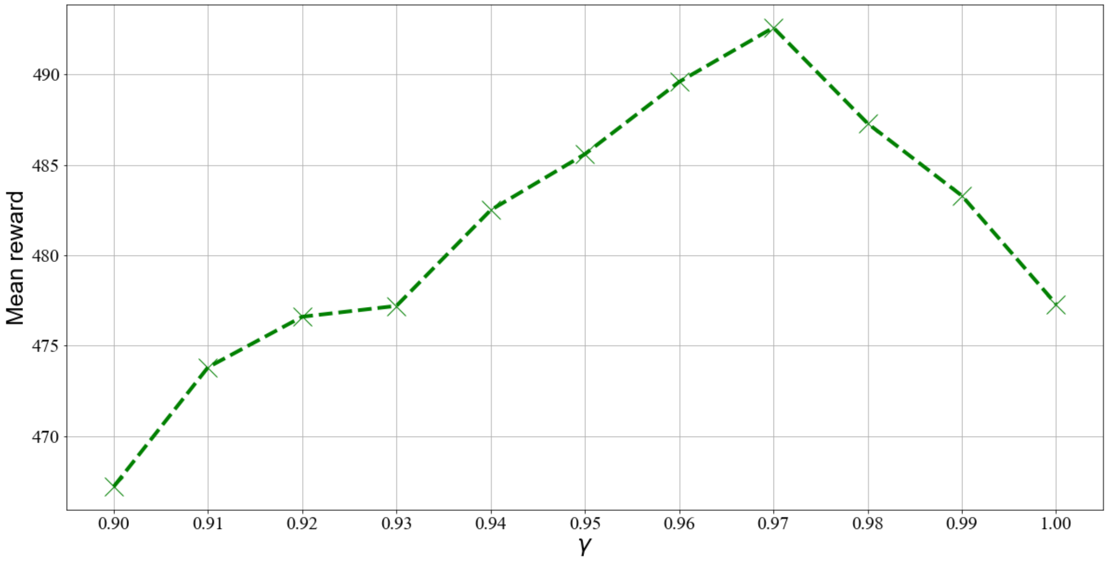


Figure 4. Mean Reward Based on Different Values of γ

3.2.4. Learning rate. Learning rate is a hyperparameter, acting like the stride length, to control the rate at which the algorithm learns and updates the parameters during the training process. If the learning rate is large, the model will be quickly modified in the beginning since each point is new to it. However, walking too fast means it may well ignore the minima and vibrate around it, thus being hard to converge. If learning rate is low, each step is too small to learn, so it will take longer time to descent and find the minimum value.

To better illustrate the performance, this paper used the concept of moving average reward, for example, average of the last hundred elements from rewards. Since the ADAM optimizer, which is an adaptive optimizer, is employed in our codes, we changed several values of its parameter, learning rate, including 0.00005, 0.0001, 0.0005. In figure 5, a comparatively high learning rate, 0.0005, seems to perform well at start with few samples, but as the problem becomes more complicated, the model fails to converge. By contrast, the learning rate of 0.00005 is too small to get a good outcome within 600 episodes. It is reasonable to assume a preferable learning rate is around 0.0001 in our DQN algorithm.

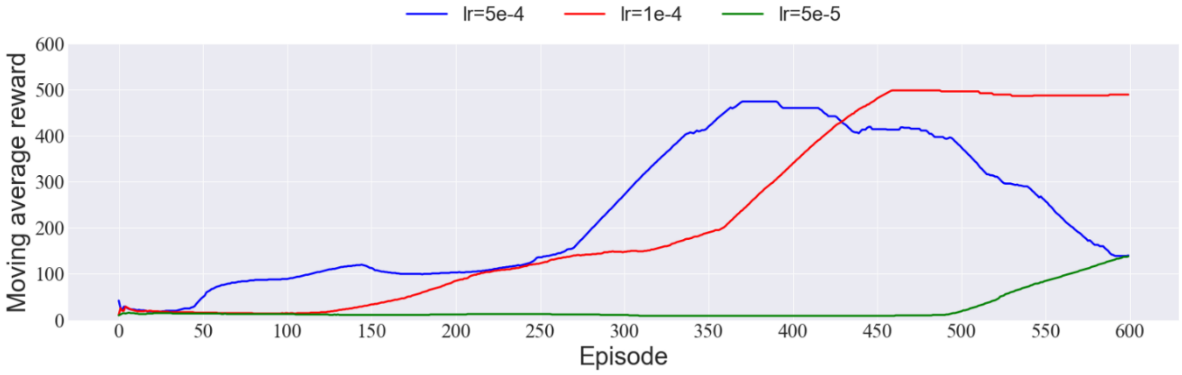


Figure 5. Comparison among the Moving Average Rewards of different Learning Rates.

Studies have found that even if ADAM optimizer is a kind of adaptive optimizer, some experiments show that adaptive approaches generalize more poorly than adaptive counterparts [6]. It may not be universally useful with problems with convergence to an optimal solution [4].

To study more about how learning rate affects DQN's performance, this paper also tests another strategy to adjust it. A common choice to manually adjust the learning rate is to decay it. The memorizing of noisy data is reduced by an initial high learning rate, while the acquisition of complex patterns is also enhanced by a subsequent declining learning rate [7]. In our algorithm, Cosine Annealing is employed to achieve that. However, figure 6 shows that it is not suitable to combine decaying learning rate with the ADAM optimizer in this algorithm. Researchers are advised to turn to some variant ADAM with long-term memory of previous gradients [4], or preferably, control the various parameters of the optimization iteration with a deeper understanding of the specific data and environment.

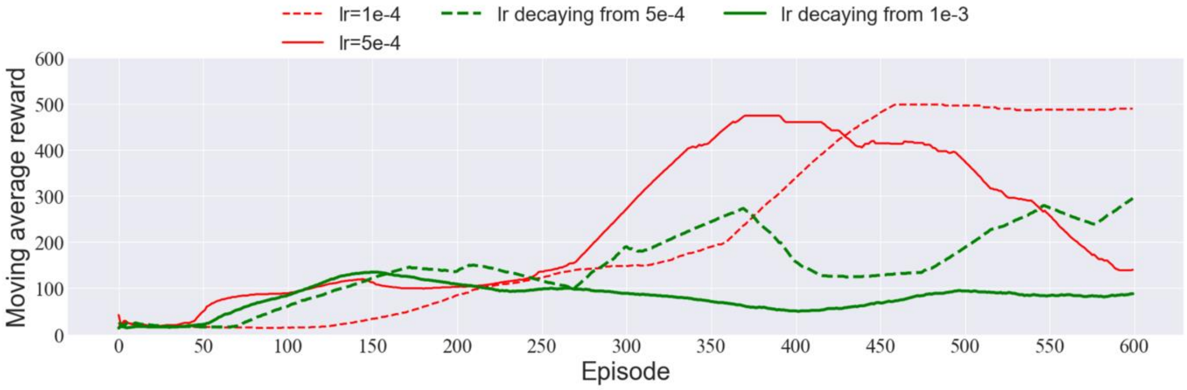


Figure 6. Comparison among the Moving Average Rewards of fixed Learning Rates and decaying Learning Rates.

3.2.5. Batch size. Batch size is the number of data samples captured in one training run. If the batch size is too small, it takes much time to train the model and potentially causes dramatic vibration. These updates need fewer calculations when the number of samples is modest. If the batch size is too large, different batches share similar gradient direction, thus easily falling into the local minima. However, with necessary dynamic adjustments, good generalization can be shown with big batches [1].

A method of updating this parameter with better performances in deep learning is to modify it proportionally to the quantity of data in Replay buffer [5]. The strategy can be transformed into the following equations:

$$k = \frac{len}{max} \tag{5}$$

$$batchSize = k \cdot basicSize \tag{6}$$

Where *k* is the scale factor, *len* is the current size of replay, *max* is the max size of replay buffer and *basicSize* is a predetermined value from which the batch size starts to be adjusted.

We compared the performances of a model using a fixed batch size of 128 and a model with that of 256, as well as one with a dynamically adjusted batch size whose basic size is 128. In figure 7 we can see that, with fixed batch size of 128, the moving average reward fluctuates despite the initial rise, while the other two converge to nearly 500. Besides, compared with using a constant big batch size of 256, proportionally adjusting it from 128 to 256 lessens calculations in the beginning but offers good overall performance.

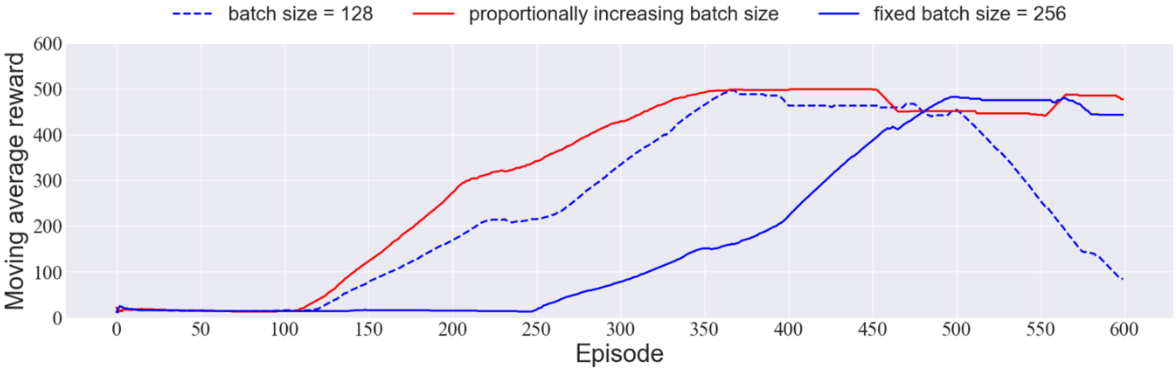


Figure 7. Comparison between Moving Average Rewards of fixed Batch Size and increasing Batch Size.

Furthermore, we compared several values of the basic batch size in figure 8, including 32, 64, 128, 256, 512. It demonstrates that 256 is a preferable choice in terms of basic batch size.

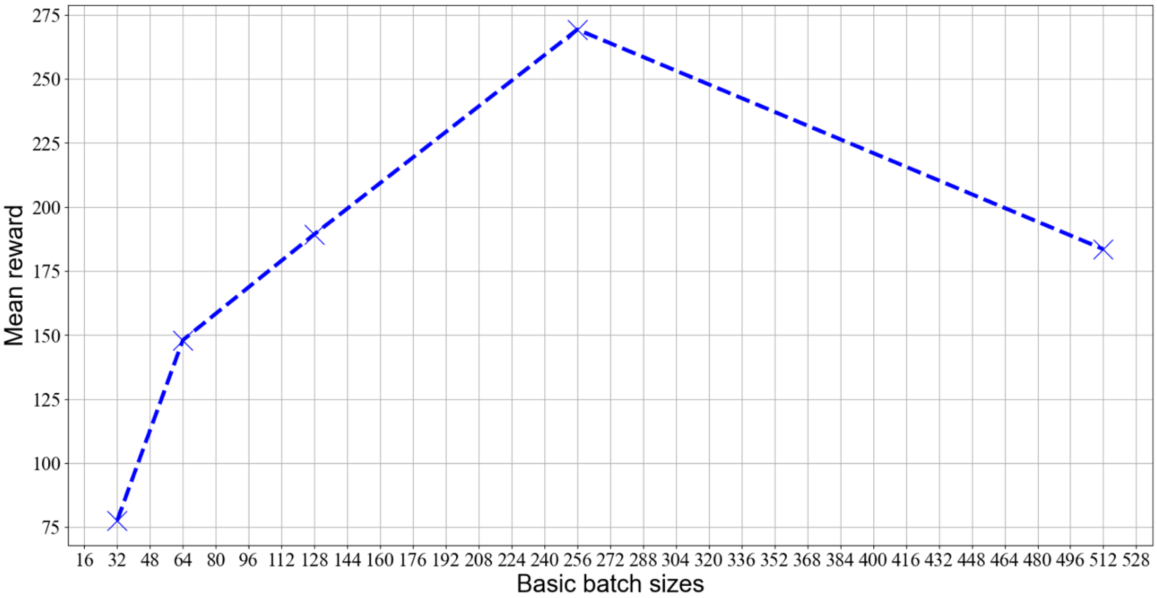


Figure 8. Mean Reward Based on Different Basic Batch Sizes.

4. Conclusion

This paper reconstructed the DQN algorithm codes according to previous work on classic DQN and soft updating. We trained it in the environment of CartPole-v1 with different hyperparameters and analyzed the performance. According to the testing results, the number of episodes and the size of replay buffer play important roles in the DQN algorithm. The more episodes, the better the performances. Expanding the replay buffer can contribute to the improvement until it exceeds 10000. In addition, gamma can improve our agent's performance most when its value is 0.97. In our environment, learning rate should not be too big or too small, a preferable choice being 0.0001, as decaying is of no necessity to be combined with the ADAM optimizer for better outcomes. What's more, the batch size is advised to be adjusted or increased in proportion to the current size of replay, where the basic size can be 256.

References

[1] Hoffer, E., Hubara, I., & Soudry, D. Train longer, generalize better: closing the generalization gap in large batch training of neural networks. 2017 ArXiv (Cornell University). https://arxiv.org/pdf/1705.08741.pdf

[2] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. Playing Atari with Deep Reinforcement Learning. 2013 *Cornell University*. http://cs.nyu.edu/~koray/publis/mnih-atari-2013.pdf

[3] Mnih, V., Kavukcuoglu, K., Silver, D., Hassabis, D. Human-level control through deep reinforcement learning. 2015 *Nature*, **518(7540)**, 529–533.

[4] Reddi, S. J., Kale, S., & Kumar, S. On the Convergence of Adam and Beyond. 2018 *International Conference on Learning Representations* **38(12)** 233-244.

[5] Smith, S. G., Kindermans, P., Ying, C., & Le, Q. V. Don't decay the learning rate, increase the batch size. 2018 *International Conference on Learning Representations.* 312-324.

[6] Wilson, A. C., Roelofs, R., Stern, M., Srebro, N., & Recht, B. The Marginal Value of Adaptive Gradient Methods in Machine Learning. 2017, *Neural Information Processing Systems,* **30**, 4148–4158.

[7] You, K., Long, M., Wang, J., & Jordan, M. I. How Does Learning Rate Decay Help Modern Neural Networks. 2019, ArXiv Cornell University. https://arxiv.org/pdf/1908.01878.pdf

[8] Zhang, S., & Sutton, R.. A Deeper Look at Experience Replay. 2017 Cornell University. https://arxiv.org/pdf/1712.01275.pdf

[9] Li H , Kumar N , Chen R , et al. Deep Reinforcement Learning. 2018 I*EEE International Conference on Acoustics, Speech and Signal Processing*,349-359.

[10] Lillicrap T P, Hunt J J, Pritzel A et al. Continuous control with deep reinforcement learning. 2015 *Computer science*, 101-115.

Research on robot path planning methods

Tianhao Chen[1,3,†], Guanhong Jiang[2,†]

[1]Department of Computer science and engineering, Shanghai University,200444, China
[2]Department of Science, Hong Kong Baptist University, Hong Kong, 999077, China

[3]chentianhao1106@shu.edu.cn
†All authors contributed equally.

Abstract. Mobile robots are used extensively across a variety of industries and fields. How to discover a start-to-end path without colliding has become a hot topic in recent years due to the complexity and uncertainty of the workplace. In various environments, a path planning technique should demonstrate high efficiency and speed. And this can reduce the energy consumption of the robots and greatly increase their working efficiency. This paper will conclude the presently popular path planning algorithm. Based on the different features of these algorithms, they are divided into three types: traditional path planning algorithm, neural-work-based algorithm, and sampling-based algorithm. Based on the new papers in these years, detailed introduction of the algorithms and their variants will be given. At the end of the paper, the thesis is summarized and the future research trend is prospected.

Keywords: path planning, traditional algorithm, neural-work-based algorithm, sampling-based algorithm.

1. Introduction

Mobile robots have been widely applied in different industry and field. Due to the uncertainty and complex of the working environment, how to find a start-to-end path without collision becomes a hot topic nowadays. An effective path planning method should show high efficiency and speed in different environment [1]. And this can significantly improve the working efficiency of the robots and reduce the energy consumption of it.

In recent years, many review articles on path planning have been published. For example, Kruse et al. surveyed the socially-aware trajectory planning [2-3]. It puts greater emphasis on robot behavior during the navigation. Chik et al. divided path planners for robot navigation into a global planner and a local planner. Although a few researches have been made on the robot path planning, few have been done to make a comprehensive study on traditional and state-of-art path planning algorithms. Thus, a detailed survey on different path planning algorithms is presented.

This paper will conclude the presently popular path planning algorithm. Based on the different features of these algorithms, they are divided into three types: traditional path planning algorithm, neural-work-based algorithm, and sampling-based algorithm. Based on the new papers in these years, detailed introduction of the algorithms and their variants will be given.

2. Traditional methods

Path planning is a crucial area of study in robotics, with consequences for automated guidance, impediment avoidance, and best path selection. The goal of this section is to analyze traditional algorithms from two different angles, which are based on obstacle avoidance algorithm and based on graph network construction, and give a comparison and analysis.

2.1. Methods based on obstacle avoidance

Artificial Potential Field (APF), a popular obstacle avoidance algorithm, is a passive obstacle avoidance method in which the robot generates a virtual potential field around obstacles in its environment and guides it away from regions of the high potential field. The theory behind APF is that a robot is attracted to a goal and rebounded by obstacles in its environment [4]. The algorithm generates a potential field around the robot, where targets are assigned low potential values and obstacles are assigned high potential values. The robot then turns to the lowest potential value, which will Orient it toward the goal and away from the obstacle. APF has a wide range of applications in robotics, including mobile robots, unmanned aerial vehicles (UAVs), and autonomous underwater vehicles (AUVs).

The bug algorithm is a direct reactive obstacle avoidance method proposed by Lumelsky and Stepanov. By following the contour of the obstacle to bypass the obstacle, the robot advances toward the desired destination. Two appreciated variants are Bug1 and Bug2. The basic defect method is Bug1. To avoid obstacles, it navigated in a direct motion. When encountering an obstacle, the robot rotates in any direction until it finds a way out. While this is simple, it can lead to invalid routes and obstacles. The wiggly bug method is called Bug2. The Bug1 algorithm works by moving along the edge of an obstacle when it is encountered until it reaches the goal point. The advantages of this algorithm are that it is simple to understand and easy to implement, and it can also achieve good results in simple environments. However, it has the disadvantage that it may go around obstacles, resulting in an increased path length [5]. The Bug2 algorithm is a modified version of the Bug1 algorithm, which detects whether there is a shorter path to the goal point when moving along the edge of an obstacle. The advantage of the proposed algorithm is the ability to avoid circling obstacles, thus reducing the path length. However, compared with the Bug1 algorithm, the Bug2 algorithm is more complex to implement, and path jams may occur in complex environments.

Vector Field histogram (VFH) is a commonly used algorithm for robot obstacle avoidance. It was originally proposed by Borenstein and Koren as a method to help robots navigate in dynamic environments. The VFH algorithm works by creating a 2D histogram of the environment around the robot, which represents the likelihood of an obstacle at each location [4]. Using this histogram, the algorithm then generates a polar histogram representing the free-space orientation of the robot. Finally, the robot chooses a steering direction that maximizes free space and minimizes the possibility of collision with obstacles. The theory behind VFH is based on the idea of a vector field, a mathematical representation of the forces acting on an object in a physical system. In the case of VFH, the vector field represents the force exerted by the robot as it moves through the environment.

Compared with these three methods, the APF method can quickly deal with large-scale environments, and can deal with multi-objective path planning problems. And using the APF method, robots can avoid obstacles and find the shortest path, but it is easy to fall into local optimal solutions, and some skills are needed to avoid this situation. The APF method may produce a "vortex effect" and cause the robot to fail to reach the target point. Because of this, the APF method is most suitable for dealing with large-scale environments and multi-objective path planning problems. Since the APF method uses an artificial potential field to describe the interaction between the robot and the environment, it can quickly deal with large-scale environments and can deal with the path planning problem of multiple target points simultaneously [6]. When dealing with multi-objective path planning problems, APF is better than Bug Algorithm and VFH. The Bug algorithm is suitable for complex environments and can deal with obstacles, gaps, and other situations that appear in the environment. At the same time, the Bug algorithm is relatively simple and easy to implement, however, this algorithm may produce a "surround effect", which causes the robot to fail to reach the goal point. In addition, the Bug algorithm requires a high initial estimation of the environment and needs to know the terrain and other information about the environment in advance. This is why Bug works best when dealing with complex terrain and situations where there are obstacles and gaps. The Bug algorithm uses the distance between pairs of points (robot and goal point) and the distance between the robot and obstacles to plan the path, so it can handle environments with complex terrain and problems such as obstacles, gaps, etc. The bug algorithm is better than APF and VFH when dealing with an environment with complex terrain and problems such as obstacles and gaps. Different from the remaining two, the VFH method can quickly and accurately detect obstacles in the environment, select the best path, and can be applied to complex environments, and can deal with various obstacles in the environment. However, this method has high requirements for environment modeling and needs to accurately estimate the position and shape of obstacles. When dealing with multi-objective path planning problems, the VFH method may only have local optimal solutions. The VFH method is most suitable for handling application scenarios that need to detect obstacles in the environment quickly and accurately. The VFH method uses lidar to scan the environment, then generates a histogram from the scanned data, and selects the best path based on the histogram. Because the VFH method can quickly and accurately detect obstacles in the environment and select the best path, it is more excellent than APF and Bug Algorithm in application scenarios that need to detect obstacles quickly and accurately in the environment.

In summary, the APF method is suitable for dealing with large-scale environments and multi-objective path planning problems. The bug algorithm is suitable for complex environments and situations that need to deal with obstacles and gaps in the environment. The VFH method can detect obstacles in the environment quickly and accurately and is suitable for dealing with complex environments. Which approach you choose depends on the specific scenario and problem requirements.

2.2. Methods based on graph network construction

Grid-based methods are a family of algorithms used in robotics and autonomous systems for planning the motion of a robot or vehicle. They operate by representing the robot's environment as a grid of cells, where each cell corresponds to a location in the space, and each cell is assigned a value that represents the accessibility or cost of that location. The history of grid-based methods can be traced back to the 1980s when researchers started to develop algorithms for robot path planning. In the early days, grid-based methods were simple and often used a binary representation to represent the obstacles in the environment. Over time, researchers developed more sophisticated grid-based methods that could represent the environment more accurately and efficiently. Another possible method is Voronoi Diagram. Paths are created by Voronoi Diagram methods along the diagram, which divides the world into regions based on how close barriers are to one another. To maximize its escape from obstructions, the robot moves along Voronoi edges.

Both the Grid-Based method and Voronoi Diagram are methods used for spatial analysis, but they each have some differences and advantages and disadvantages. The grid-based method is to divide the spatial region into uniform grids and treats each Grid as a unit. The advantage of this approach is that it can handle large-scale data because the computation of each lattice is relatively independent and can therefore be processed in parallel. Moreover, since the size of the lattice is fixed, the results of the proposed method are also predictable for data with different resolutions. The disadvantage is that since the size of each cell is fixed, it may lead to defects or errors in the analysis results, especially when the data density and distribution are uneven in different spatial regions. So, when the data is evenly distributed in the space or when the amount of data is large, the Grid-Based method can better process the data, because it can evenly divide the space area into grids and processes, which can take full advantage of parallel computing. In contrast, Voronoi Diagram is a geometrically based method that partitions a spatial region into polygonal cells centered at data points, where each cell contains all points that are at an equal distance to the nearest data point. The advantage of this method is that it can capture the characteristics of the data more accurately because it can adapt to different data distributions depending on the distribution of the data points. In addition, since the proposed method uses geometric shapes, it can better express spatial relationships and spatial similarities. The drawback is that the computational complexity of the method is high, so it may not be able to handle large-scale data. So, when the data is unevenly distributed in space, the use of the Voroni Diagram can better capture the characteristics of the data, and when the data has a complex spatial distribution, the use of the Voroni Diagram can better represent the spatial relationship and spatial similarity.

To put it simply, traditional algorithms are frequently simple to comprehend and apply, making them a common option for many route-planning tasks. These algorithms have been thoroughly investigated and implemented in numerous situations, showing their efficacy in solving path-planning issues. They can also be modified or combined to better fit the needs of a given application. However, some conventional methods might produce inefficient or less secure routes for the automaton. Local minima or fluctuations in some algorithms, like the Artificial Potential Field technique, can impede the robot's progress. Additionally, some methods might be computationally difficult, particularly in settings with lots of obstacles or high-resolution models.

3. Neural-network based methods

3.1. Reinforcement learning

Reinforcement learning (RL) is an effective method that helps the agent make the best choice based on the past actions and results without any provided samples to learn in advance. In the field of path planning, this method uses feedbacks from the environment as the reward to stipulate the best path. Figure 1 shows a widely used RL model, the actor critic model.

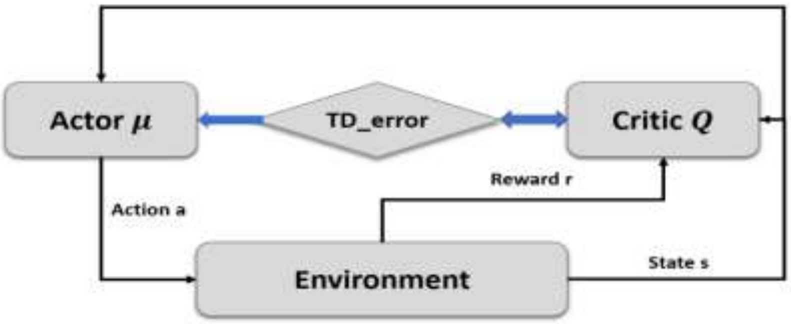


Figure 1. The actor critic model.

RL and its variants have been widely applied for robot path planning, especially in real-time tasks. Zhang et al. propose a path planning model called SG-RL. It uses the Simple Subgoal Graphs (SSG) to look for the best paths, and reinforces the decision-making process with the Least-Squares Policy Iteration (LSPI) method [7]. It’s proved that this model can better adapt for the dynamically changing environment and achieve good performance on large-scale maps, since SSG resolves the limitations of sparse reward and local minima trap for RL agents and thus LSPI can be applied to deal with slight changes from the environment.

To raise the ability of the robots to perform more complex missions, GyeongTaek proposes a Goal-Conditioned RL method (Figure 2). It uses the bi-directional memory editing to raise its robustness and a sub-goal dedicated network to ensure the agent is fully controllable [8]. The reward function is redesigned to help shape the shortest path on propose for the optimal performance.

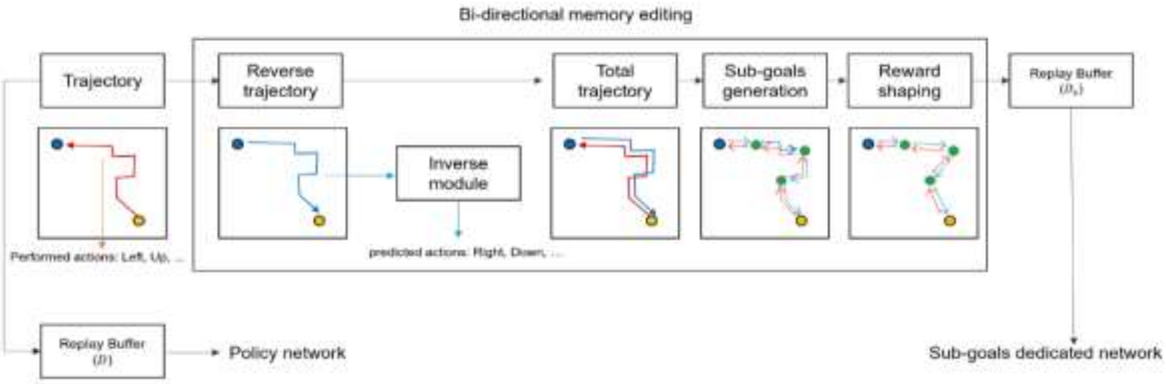


Figure 2. Illustration of the GC-RL method.

Deep Reinforcement Learning (DRL), a method combining RL and DL (Deep Learning) has also been a hot topic in path planning. Bouhamed et al. designed a Deep Deterministic Policy Gradient (DDPG) based DRL approach which exhibits stronger real-time learning ability from surrounding environment. In autonomous robot exploration tasks, Yuhong Cao et al. propose a neural network named ARiADNE, combining the attention mechanism and DRL [9]. It can learn spatial dependencies from the partial map and predict possible gains for unexplored areas, which assist the agent to make non-myopic movement decisions. It is proved to outperform most state-of-the-art methods. Migual et al. compare two deep Q network (DQN) methods, the D3QN and rainbow algorithm in path planning tasks, and make the conclusion that the rainbow DQN performs better in most tasks, suggesting its feasibility in DRL approach design.

3.2. Deep learning

Compared to RL methods, Deep learning (DL) method learn to plan path based on extracted features from samples. It is suitable for tasks having a large number training samples.

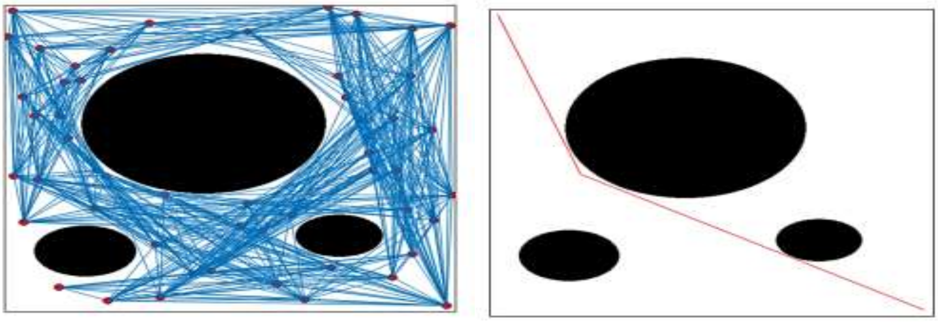
Many scholars apply DL method in mobile robots, among which the Convolutional Neural Network (CNN) is the most widely adopted [10]. To improve the algorithm’s efficiency in large and complex environment, Janderson et al. propose a CNN encoder to overcome the limit that the model can only extract features from linear information, thus reducing data dimensionality, namely useless paths in the environment. This model is proved to decrease the execution time by 54.43%. Shen et al. propose a Coverage Path Planning Network (CPPNet) [11]. A CNN network with graph-based input and output. The edge value of the output graph is the probability of belonging to the TSP tour, and a greedy search is used to find the best TSP tour. Liu et al. combine the CNN and RRT* algorithm, and design a learning-based algorithm, which regards the environment as a RGB image input to predict unexplored environment, assisting the RRT* planner to make faster path stipulation.

4. Sampling-based methods

Sampling-based method find the feasibility of the path through collision detections, avoiding the detailed expression of the environment. Sampling-based algorithm connects all of the feasible nodes and find a start-to-goal feasible path based on it [12-13]. The most widely used sampling-based approaches are probabilistic roadmaps (PRM) and random-exploring trees (RRT).

4.1. PRM algorithm

PRM algorithm includes two phases: a learning phase and a query phase. In the learning phase, it constructs the roadmaps with nodes and edges representing feasible paths and uses a local fast planner to calculate them. Then in the query phase, it will find the optimal one given the start and goal configurations.



(a) Generating nodes and find feasible path (b) The best path offered by the algorithm

Figure 3. PRM algorithm working diagram.

PRM algorithm performs well in high-dimensional search space, but in some tasks, it is not that stable (Figure 3). To resolve this limitation, Yang et al. propose a sample adjustment method during the construction of the roadmaps. This post-processing method will adjust the randomly generated nodes to meet the soft constraints, which influence the behavior of the agent, required by the problems. To better adapt for the dynamic environment, Ahmed et al. improve the PRM algorithm by dividing the domain of motion and deal with the relevant path in each of it. Chen et al. propose a modified algorithm called P-PRM, which introduces in the concept of the potential field in its planning area. The new part is adopted to choose valuable nodes to avoid collision for the sampling points, and cost function is specifically designed to avoid local optimum. Compared to traditional PRM algorithm, this method has higher efficiency and faster execution time. Ankit et al. propose a method called HPPRM to solve narrow passage problems [14]. It distributes nodes through segmenting roadmaps into high and low potential areas and reduce the dispersion of sample set during roadmap construction. It is proved that it has greater success rate and lower calculation cost than traditional methods.

4.2. RRT algorithm

RRT algorithm was proposed by Lavalle et al. in 1998. It constructs random trees to achieve a start-to-end path without collision. The RRT regards the start point as the root and find the best path through costly regeneration (Figure 4).

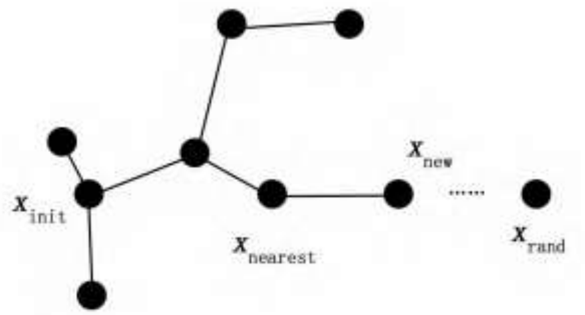


Figure 4. shows the process of the RRT.

The RRT algorithm has a great adaption to the environment, therefor it’s applied to real-time task. To improve the performance of the RRT algorithm in different dynamic environment and multi-query tasks, Daniel et al. proposes a method, AM-RRT*, which can extend the RRT-based sampling approach and use an assisting metric to store beneficial results [15]. Experimentations demonstrate the improved method’s effect in execution time reduction compared to RT-RRT*. Thomas et al. propose a Grounding-aware RRT* algorithm, which can be applied to marine tasks, to improve its ability to avoid collision. Previous data from both the environment and navigation experience is encoded and transferred to

RRT*'s cost function, so that path deviation can be penalized and a better path alternation is therefore made. Meng et al. combine the deep learning with the RRT algorithm to widen the appliable field of RRT planner. A new algorithm named NR-RRT is presented. It implements a neural network sampler to improve the safety of possible chosen state, and use the bi-directional search strategy to fasten the execution time. It is tested that this algorithm performs better trade-off between efficiency and safety than other state-of-order method [16]. Energy consumption of RRT algorithm remains a hot topic in the research of RRT algorithm. To have better efficiency, Pedram et al. propose the information-geometric RRT* (IG-RRT*) algorithm. The problem is better resolved by reducing the large number of nodes needed to deal with. Only a part of these nodes is chosen to be run on, and a smoothing algorithm, which can be seen as an optimization function, is applied to adjust the path planning result.

5. Conclusion

In conclusion, path planning is a critical component of robotics and autonomous systems, and there are various methods available for its implementation. This essay has introduced both traditional algorithms, such as Bug Algorithms, VFH, APF, and Grid-based methods, as well as neural-network based algorithms, such as DRL and CNN. Traditional algorithms have been in use for many years and have proven to be effective in many applications. They are often simpler and more interpretable than neural-network based algorithms, making them a popular choice for certain applications. However, traditional algorithms may struggle in complex, dynamic environments, and may require extensive tuning to achieve optimal performance. Neural-network based algorithms, on the other hand, are increasingly popular due to their ability to learn complex behaviors and adapt to changing environments. They have shown impressive results in many applications, particularly in robotics and autonomous vehicles. However, they can be computationally expensive and difficult to interpret, which may limit their use in some applications.

The future of path planning lies in the integration of both traditional and neural-network based algorithms, with each method being used for its strengths. The use of artificial intelligence-based approach in path planning is rapidly evolving, and it promises to revolutionize the way we interact with robots and autonomous systems in the future.

References

[1] Tang, Z., & Ma, H. An overview of path planning algorithms. 2021 Earth and Environmental Science. 804 (2), p. 022024.

[2] Li, X., Hu, X., Wang, Z., & Du, Z. Path planning based on combination of improved A-STAR algorithm and DWA algorithm. 2020 International Conference on Artificial Intelligence and Advanced Manufacture, 99-103.

[3] Boots, B., Sugihara, K., Chiu, S. N., & Okabe, A. Spatial tessellations: concepts and applications of Voronoi diagrams 2009, John Wiley & Sons.

[4] Chen, W., Wang, N., Liu, X., & Yang, C. VFH based local path planning for mobile robot. In 2019 China Symposium on Cognitive Computing and Hybrid Intelligence, 18-23.

[5] Xu, Q. L., Yu, T., & Bai, J. The mobile robot path planning with motion constraints based on Bug algorithm. 2017 Chinese Automation Congress, 2348-2352.

[6] Chen, Y. B., Luo, G. C., Mei, Y. S., Yu, J. Q., & Su, X. L. UAV path planning using artificial potential field method updated by optimal control theory. 2016 International Journal of Systems Science, 47(6), 1407-1420.

[7] Zeng, J.; Qin, L.; Hu, Y.; Hu, C.; Yin, Q. Combining Subgoal Graphs with Reinforcement Learning to Build a Rational Pathfinder. 2019 Application. Science, 9, 323.

[8] Lee, GyeongTaek. A Fully Controllable Agent in the Path Planning using Goal-Conditioned Reinforcement Learning. 2022 10.48550/arXiv.2205.09967.

[9] Cao, Y., Hou, T., Wang, Y., Yi, X., & Sartoretti, G. ARiADNE: A Reinforcement learning approach using Attention-based Deep Networks for Exploration. 2023 ArXiv, abs/2301.11575.

[10] Y. Zhang, J. Zhao and J. Sun, Robot Path Planning Method Based on Deep Reinforcement Learning, 2020 International Conference on Computer and Communication Engineering Technology, 49-53.

[11] Z. Shen, P. Agrawal, J. P. Wilson, R. Harvey and S. Gupta, CPPNet: A Coverage Path Planning Network, 2021 OCEANS. 1-5.

[12] J. Liu, B. Li, T. Li, W. Chi, J. Wang and M. Q.H. Meng, Learning-based Fast Path Planning in Complex Environments, 2021 IEEE International Conference on Robotics and Biomimetics. 1351-1358.

[13] E. M. Ahmed, H. E. Abd El Munim and H. M. Shehata Bedour, An Accelerated Path Planning Approach, 2018 International Conference on Computer Engineering and Systems, 15-20.

[14] A. A. Ravankar, T. Emaru and Y. Kobayashi, HPPRM: Hybrid Potential Based Probabilistic Roadmap Algorithm for Improved Dynamic Path Planning of Mobile Robots, 2020 IEEE Access, 8 221743-221766.

[15] D. Armstrong and A. Jonasson, AM-RRT*: Informed Sampling-based Planning with Assisting Metric, 2021 IEEE International Conference on Robotics and Automation, Xi'an, 10093-10099.

[16] Pedram, Ali Reza & Tanaka, Takashi. A Smoothing Algorithm for Minimum Sensing Path Plans in Gaussian Belief Space. 2023 IEEE Transactions on Robotics 32(5).

Research of different feature detection and matching algorithms on panoramic image

Jindong Xiao

School of Software Engineering, Shenzhen University, Guangzhou, China

2020151007@email.szu.edu.cn

Abstract. Image stitching is the process of combining numerous photos to make a panorama. The technique of image stitching has rapidly advanced and grown to be a significant area of digital image processing. Many image stitching methods have been proposed and studied in prior study. In this paper, the image stitching process is implemented using different algorithms. For keypoints detection, the algorithms of Harris corner detection, SIFT(Scale-Invariant Feature Transform), SURF(Speeded Up Robust Feature) and ORB(Oriented FAST and Rotated BRIEF) algorithms are applied, then use different methods (i.e., Brute Force, etc.) for feature matching. The RANSAC(Random Sample Consensus) method is used to calculate a homography matrix from matched feature vectors and use it to warp the images. Image blending and cropping methods are proposed to enhance the image quality. Given groups of the self-captured images, experiments have been down to shown the performance of different techniques.

Keywords: panoramic mosaic, SIFT, Harris, SURF, ORB, blending, warping, homography.

1. Introduction

Image stitching is the process of integrating multiple images with overlapping fields of vision to create a segmented panorama image. This method is a crucial component of digital picture processing and is widely used in fields such as remote sensing, aerospace, virtual reality, and medical imaging. The process involves feature detection and extraction, feature matching, using RANSAC(Random Sample Consensus) to estimate homography matrix, image warping, alignment, and image blending and cropping. In previous years, significant progress has been made in each of these steps. For example, algorithms like Harris corner, SIFT(Scale-Invariant Feature Transform), SURF(Speeded Up Robust Feature), and ORB(Oriented FAST and Rotated BRIEF) have been developed for feature detection and extraction for satisfying the increasing demands of time and precision, while image blending and cropping techniques are used to enhance image quality, for obtaining an image with excellent stitching which clear, has no black edge [1].

In this paper authors apply different algorithms of feature matching, and explore the methods of image blending and cropping, after the estimation of homography matrix using RANSAC and the image warping between two images. Meanwhile, we consider the order of the stitching by making the central image to be fixed and warping the images under the analysis of the positions for better image stitching results.

The remainder of the paper is divided into the following parts. Section 2 shows the methods of feature detection and descriptor extractions, feature matching algorithms, image stitching process, and image blending and cropping methods. Section 3 illustrate our image matching methodology by using the example given the input groups of images, the difference of the results is explored and analyzed. We offer our findings and suggestions in Section 4.

2. Method and technology

2.1. Problem description

Panoramic image stitching, in which many photos are stitched together to generate a larger, wider image, is a useful approach in many industries. Yet, there are numerous obstacles in picture alignment and fusion that must be overcome in order to achieve high-quality panoramic image stitching. Accurate feature point recognition and matching, handling various perspective distortions and lighting variations, and handling lens distortion are important considerations. This paper focuses on the stitching of multiple pictures using feature point detection and matching algorithms and evaluates the performance of several approaches. This procedure is implemented using Python and OpenCV.

2.2. Input data

The input data is linear multiple images captured by same camera with overlapping areas. (Figure 1-Figure 3). Three data sets are selected and captured in different condition. Figure 1 and Figure 2 are indoor images set. Figure 3 is outdoor images set. Three sets of images have different lighting conditions, shooting angles, and sceneries, which also help to evaluate the adaptability and universality of the algorithms. The number of images in each set are all above 10 (about 10 to 12 images).

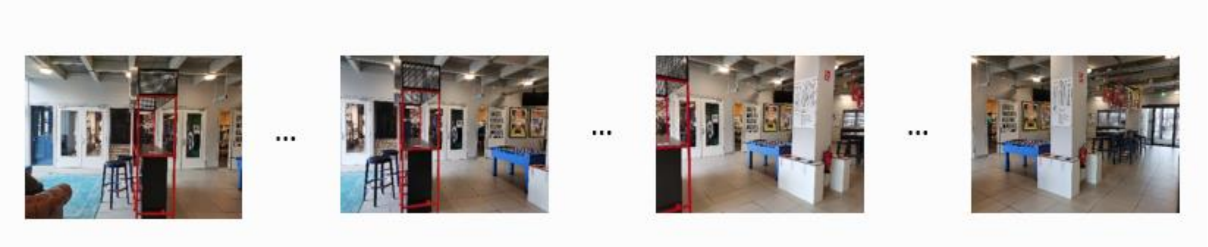


Figure 1. Indoor images set 1.

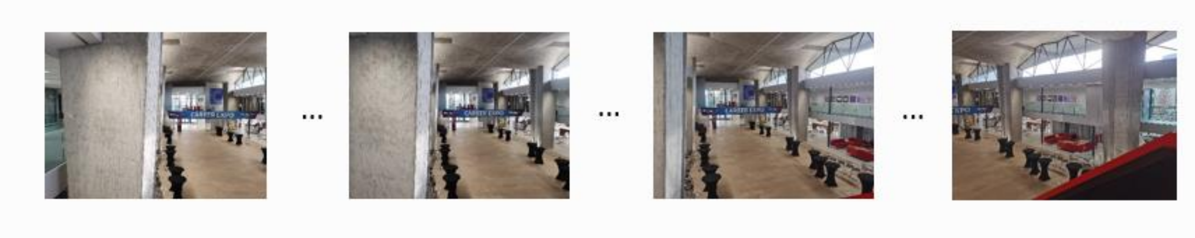


Figure 2. Indoor images set 2.

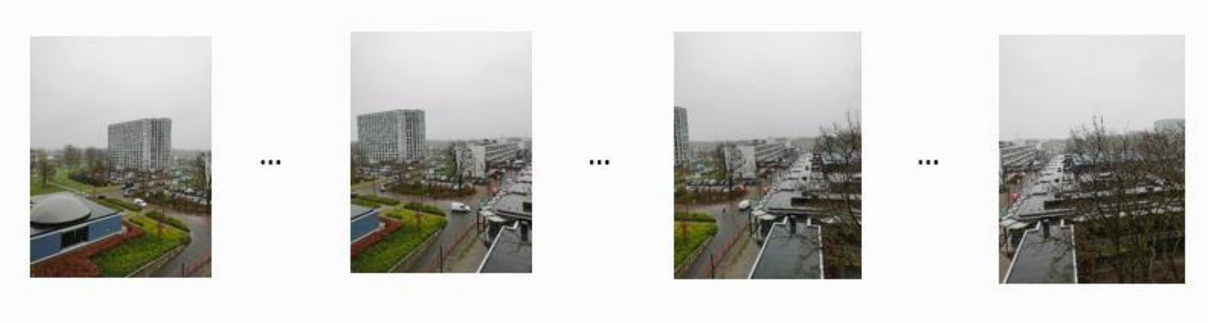


Figure 3. Outdoor images set.

2.3. Feature detection and extraction

2.3.1. Harris corner detector. The Harris uses the intensity fluctuation in a local neighborhood to identify points. A small area close to the feature should display a significant intensity change compared to windows that have been moved in either direction [2]. There are procedures of Harris corner detector in table 1.

Table 1. The procedures of Harris.

The procedures of Harris
1) Gradient calculation: Use a filter like Sobel, Scharr, or Prewitt to calculate the image gradients in the x and y directions.
2) Structure tensor computation: Using the gradients, compute a 2x2 structure tensor for each pixel. The local image structure surrounding the pixel is encoded by the structure tensor, which is a matrix.
3) Corner response calculation: Use the Harris corner detector to determine the corner response function for each pixel.
4) Non-maximum suppression: Suppress non-maximum responses in the image to produce a set of local maxima that correspond to corner locations
5) Thresholding: Choose the most prominent corners by applying a threshold to the corner response values.

2.3.2. SIFT algorithm. SIFT (Scale-Invariant Feature Transform) is a well-liked feature detection approach in computer vision and is used for a range of applications like picture matching, object recognition, and 3D reconstruction. It mainly has four stages to get the set of image features [3].

A. Scale-space peak selection

Using scale-space peak selection, the system can identify features at various scales. The image is convolved with a succession of Gaussian filters at progressively larger sizes to produce the scale-space representation. By removing adjacent levels from the Gaussian pyramid, the SIFT method creates a difference-of-Gaussian (DoG) pyramid from the scale-space representation as Figure 4. The DoG pyramid acts as a scale-invariant feature detector and draws attention to areas of the picture that have high contrast and curvature.

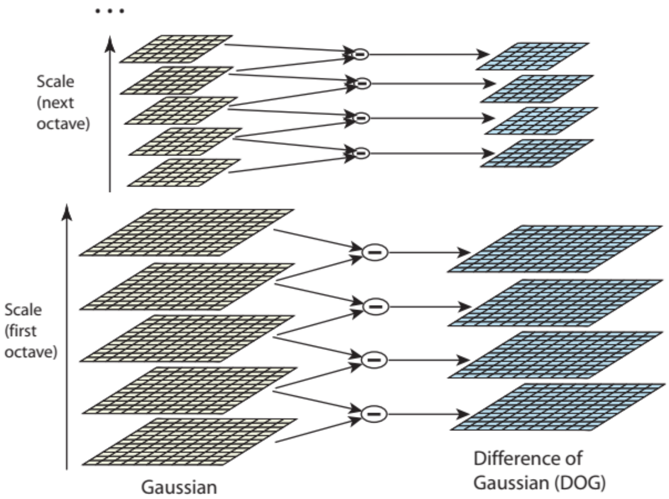


Figure 4. Difference-of-Gaussian (DoG) pyramid from the scale-space representation (from [2]).

B. Keypoint localization

SIFT recognizes probable keypoints as local DoG function scale and space extrema at each level of the pyramid. A pixel's 26 immediate neighbors in the level it is now in and 9 immediate neighbors in the levels above it is used to determine the local extrema. By calculating the difference between the pixel and its neighbors and determining whether the pixel is the maximum or minimum in its neighborhood, a comparison is made.

C. Orientation assignment

By calculating the orientation of the gradient histogram, the main orientation can be determined, which gives the SIFT algorithm rotational invariance [4].

D. Keypoint descriptor

SIFT generates scale, orientation, and translation invariant key point descriptors by computing a gradient orientation histogram of the picture around the key point location.

2.3.3. SURF algorithm. The SURF (Speeded Up Robust Feature) algorithm approximate the DoG (Difference of Gaussians) useing box filters instead of image Gaussian averaging, because integral image convolution using squares is quicker [5]. Additionally, the SURF algorithm accelerates calculations by using fast approximations of the Hessian matrix and descriptors through the use of integral images [6]. There are mainly 2 steps in SURF:

A. Interest point detection

In the SURF algorithm, the original image is converted into a composite image, where I_(x,y) represents the total amount of pixels in a rectangle whose top-left corner is (0,0) and bottom-right corner is (x,y) (Figure 5) [7]. This is achieved by using only four array references, thus allowing for efficient computation of the total pixel sum in any rectangular region through the use of integral images.

$$I_{\Sigma}(x,y)=\sum_{i=0}^{i\leq x}\sum_{j=0}^{j\leq j}I(x,y) \tag{1}$$

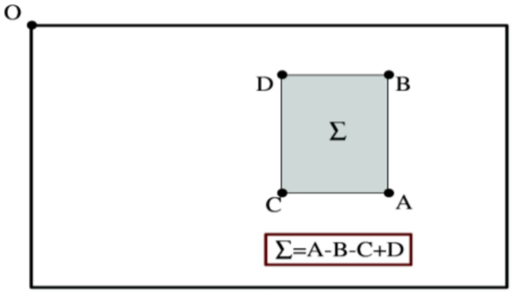


Figure 5. Using integral images.

The SURF detector relies on the Hessian matrix's positive determinant, which is calculated by convolving the picture with an appropriate kernel to obtain the second-order partial derivatives of a function. Figure 6 [6] illustrates the convolution of the integral image with a box filter.

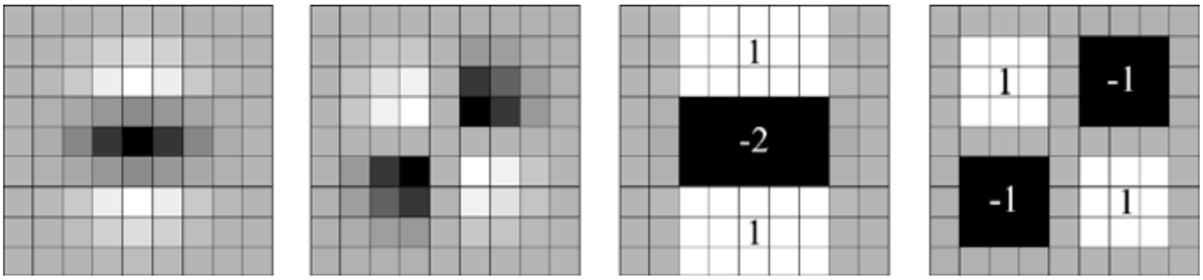


Figure 6. The integral image convoluted with box filter.

B. Interest point description

There are two steps in the construction of the SURF descriptor:

a. The orientation assignment.

For each keypoint, it is necessary to determine its main orientation. Haar wavelet responses can be computed in the image region surrounding the keypoint to provide rotation invariance, and the main orientation can be determined using these responses.

b. Extract the descriptor.

Once the main orientation of a keypoint is determined, it is necessary to compute its local feature descriptor. The SURF algorithm uses a technique called accelerated integral images to calculate the local feature descriptor.

2.3.4. ORB algorithm. ORB (Oriented FAST and Rotated BRIEF) is a swift binary descriptor that combines Binary Robust Independent Elementary Features (BRIEF) keypoints with FAST(Features from Accelerated Segment Test) detectors to create an efficient and robust method for feature detection and description [8]. There are two main steps of ORB algorithm:

A. Oriented FAST corner detection

ORB initially uses the FAST corner detector to find keypoints in the image. FAST is a method for detecting corners characterized by its fast computation speed and stability across images of different scales.

B. rBRIEF description

After obtaining the Oriented FAST keypoints, the ORB algorithm uses an enhanced version of the BRIEF algorithm. BRIEF is a binary vector descriptor composed of a series of 0s and 1s, offering a compact and efficient representation of the features [9]. ORB encodes the local region around keypoints using BRIEF descriptors. The BRIEF descriptor is combined with the orientation data of the keypoints in ORB's rotation-invariant BRIEF descriptor, which achieves rotation invariance.

2.4. Feature matching

The matching procedure compares descriptor data between matching points in two images. The locations of identical features are recognized as matched pairs if the features in the input images match. Matching algorithms such as Brute Force (BF), Fast Library for Approximate Nearest Neighbors (FLANN), and K-Nearest Neighbors matcher (KNN) [10] are used in this study.

A. BF matcher:

The BF matcher considers all possible matches and selects the best matches from the initial set. It compares one feature from the first image to all features in the second image by measuring the distance between them [10].

B. FLANN matcher:

The FLANN uses a custom algorithm library that can efficiently search for nearest neighbors in high-dimensional feature space. It employs random KD tree and k-means tree algorithms to conduct a prioritized search. The random KD algorithm swiftly locates the nearest points to a given input point by conducting parallel tree searches. The priority search K-means tree algorithm segments data into regions and reorganizes them until each leaf node contains more than M elements. It selects the initial center randomly., making it faster than the BF matcher for large datasets [10].

C. KNN matcher:

KNN matcher is a matching algorithm that finds the best matches between features in two images based on the k-nearest neighbors. It displays the k-best matches, where k is determined by the user. The algorithm generates lines from the features in the first image to the matching best match in the second image after stacking the two photos horizontally [10].

2.5. Image stitching using homography

Once we have obtained the feature matching information for all images, we can utilize this for image matching. During the image matching process, RANSAC algorithm is employed to estimate a homography matrix.

RANSAC (Random Sample Consensus) [11] is an iterative model fitting algorithm used to estimate model parameters from data with a lot of outliers (such as noise or incorrect matching points). Table 2 is the mainly procedure of RANSAC.

Table 2. RANSAC algorithm procedure.

RANSAC algorithm
1) Select n data points randomly from them;
2) Estimate parameter x to calculate the transformation matrix;
3) Use this data point to fit a model;
4) The remaining data points' distance from the model should be calculated. An outlier point is one when the distance is greater than the cutoff. An intra-office point is one where the value does not exceed the threshold. Consequently, identify the model's corresponding intra-office point value.

After the homography matrix, we apply a warping transformation to stitch the images. The steps are as follows:

A. Get the height and width of two images.
B. Extract the coordinates of four corners from each image.
C. Apply homography transformation to the corner points of the source image, which is the image to be warped.
D. Concatenate the corner points of two images, and then find the minimum x and y coordinates as well as the maximum y coordinate, which are used for image stitching.
E. Since we consider the central image is fixed, we assume that if the top-left corner of the source image has a coordinate less than 0, then it is stitched to the left side of destination image; otherwise, it is stitched to the right side.

2.6. Blending and cropping

A. Blending

Non-binary alpha mask is used to blend two images. The method is to select a suitable size window to blur the seam of image stitching and use non-binary alpha mask to blend two images. the value of the alpha mask image is between 0 and 1, representing the proportion of the weight of each pixel. For each pixel, the original image and overlay image pixel values are linearly blended according to the weight in the alpha mask. Using following formula to compute the value of each pixel:

$$F(i,j) = \omega_1 A(i,j) + \omega_2 B(i,j) \quad (2)$$

An important thing of non-binary alpha mask is to choose an appropriate size of window, it makes resulting images smooth but no ghosting. One eighth of destination image is chosen in this project.

Obviously, ω_1+ω_2=1. At the seam of image stitching, the value of ω_1 and ω_2 are both between 0 and 1. Otherwise, ω_1=1 and ω_2=0, meaning that it's the A image part. ω_1=0 and ω_2=1, meaning that it's the B image part.

B. Cropping

There are also some black edges because of warping transformation. So it needs to implemented a function to crop image. There are several steps for cropping in table 3.

Table 3. The procedure of cropping.

The procedure of cropping
1) Determine the corners' minimum and maximum x and y coordinates (4 corners of the warped image and 4 corners of the destination image). 2) Determine the translation vector t (t = [-xmin, -ymin]) that will be used to get the displacement of the image. 3) The warped picture is stitched to the left side of the target image if the x-coordinate of the top-left corner of the warped image is smaller than 0. Otherwise, is stitched to the right side. 4) Then using these corners, the edge of cropped image can be calculated without black edge.

Thus, the overall process of image stitching is as follows:

Table 4. Image stitching procedure.

Image stitching
1) Read the input image; 2) Detect key feature points of images 1 and 2, using algorithms of Harris corner, SIFT, SURF, ORB respectively, and calculate feature descriptors; ▪ Build Harris corner, SIFT, SURF, ORB generator; ▪ Detect Harris corner, SIFT, SURF, ORB feature points and calculate descriptors; ▪ Return the set of feature points and remember the corresponding description feature. 3) Set up matcher; 4) Use BF and KNN to detect SIFT feature matching pairs from images 1 and 2, K=2; 5) When the matching point pairs after screening are larger than 4, the view transformation matrix is calculated, and H is the view transformation matrix of 3 × 3; 6) Match all the feature points of the two images and return the matching result; 7) If the return result is empty, it proves that there is no feature point matching the result and exits the program; 8) Otherwise, the matching result is extracted; 9) Transform the view Angle of image 1, and the result is the transformed image; 10) Judge if the warped image 1 is on the left or the right side of the source image, stitch the images.

3. Experiment

3.1. Platform introduction

The platform configuration parameters used in this report are the compiler and version pycharm, using OpenCV in the python language as a framework, creating a virtual environment using Anaconda, and installing Python 3.10.0 and 3.6 in the virtual environment for testing different algorithms.

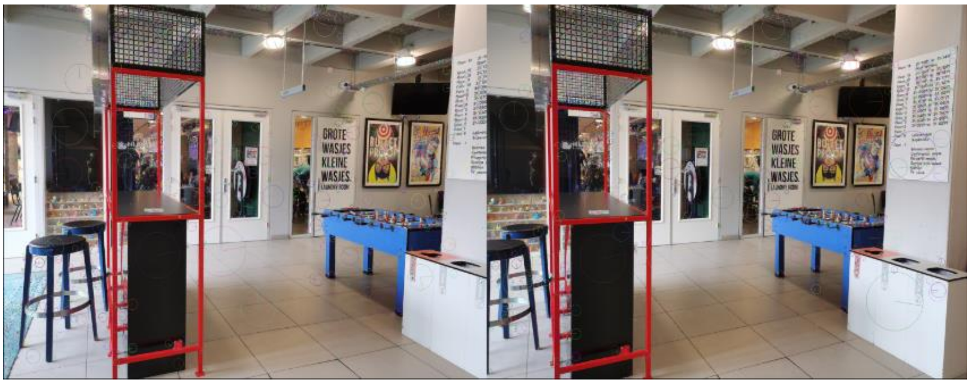
3.2. Results illustration

3.2.1. KeyPoints detection

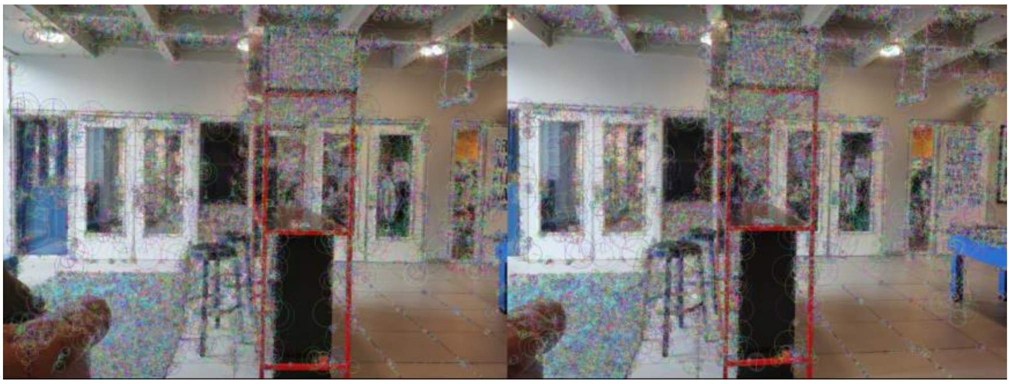
Figure 7 shows the results of the keypoints detection of the 1st and 2nd images of indoor image set 1, applying different feature detection methods.

Harris corner detection:

- SIFT:



- SURF:



- ORB:

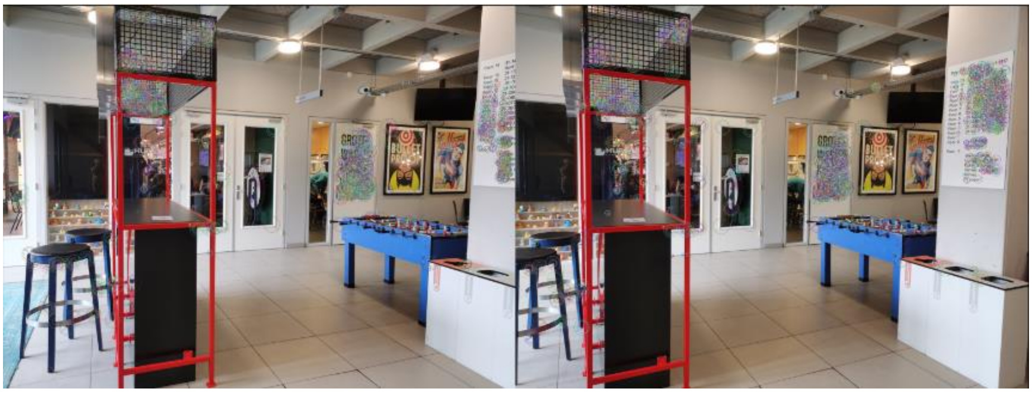
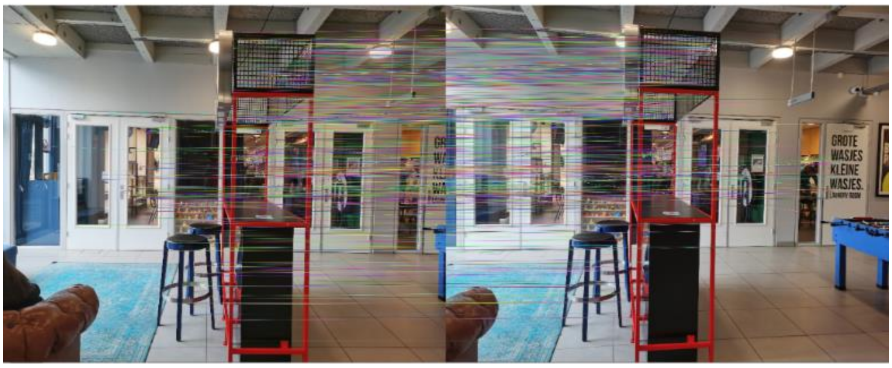


Figure 7. Keypoints detections results (indoor images set 1).

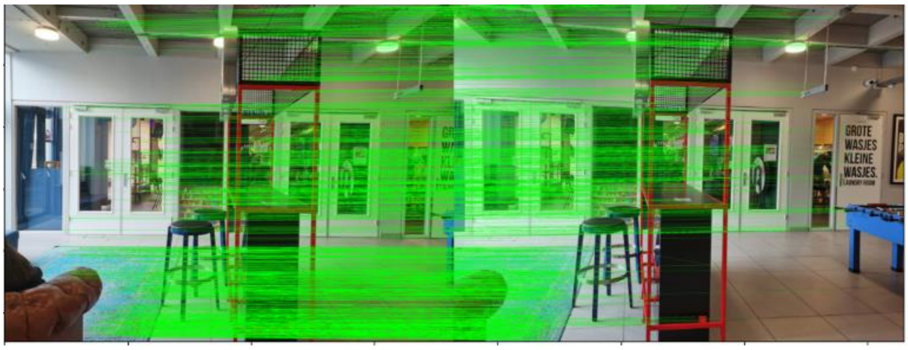
3.2.2. Feature matching

Figure 8 shows the results of the feature matching of the 1st and 2nd images of indoor image set 1, given the detected keypoints and corresponding descriptors obtained in 3.2.1.

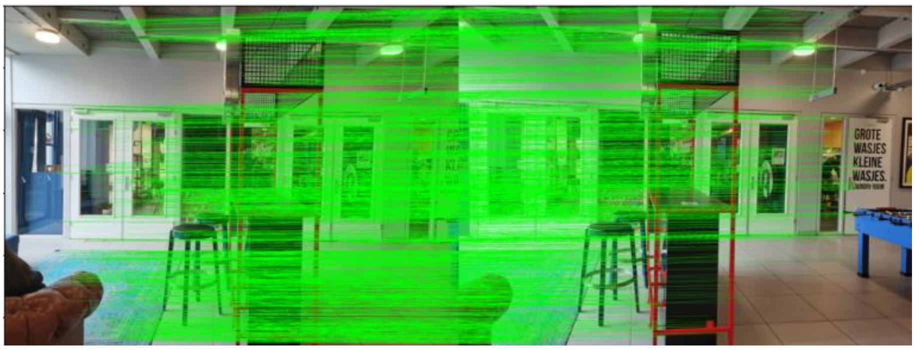
Harris corner detection + SIFT descriptors:



- SIFT:



- SURF:



- ORB:

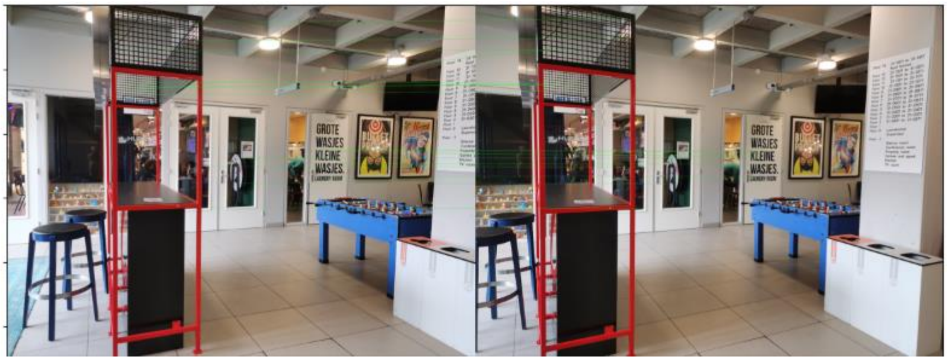
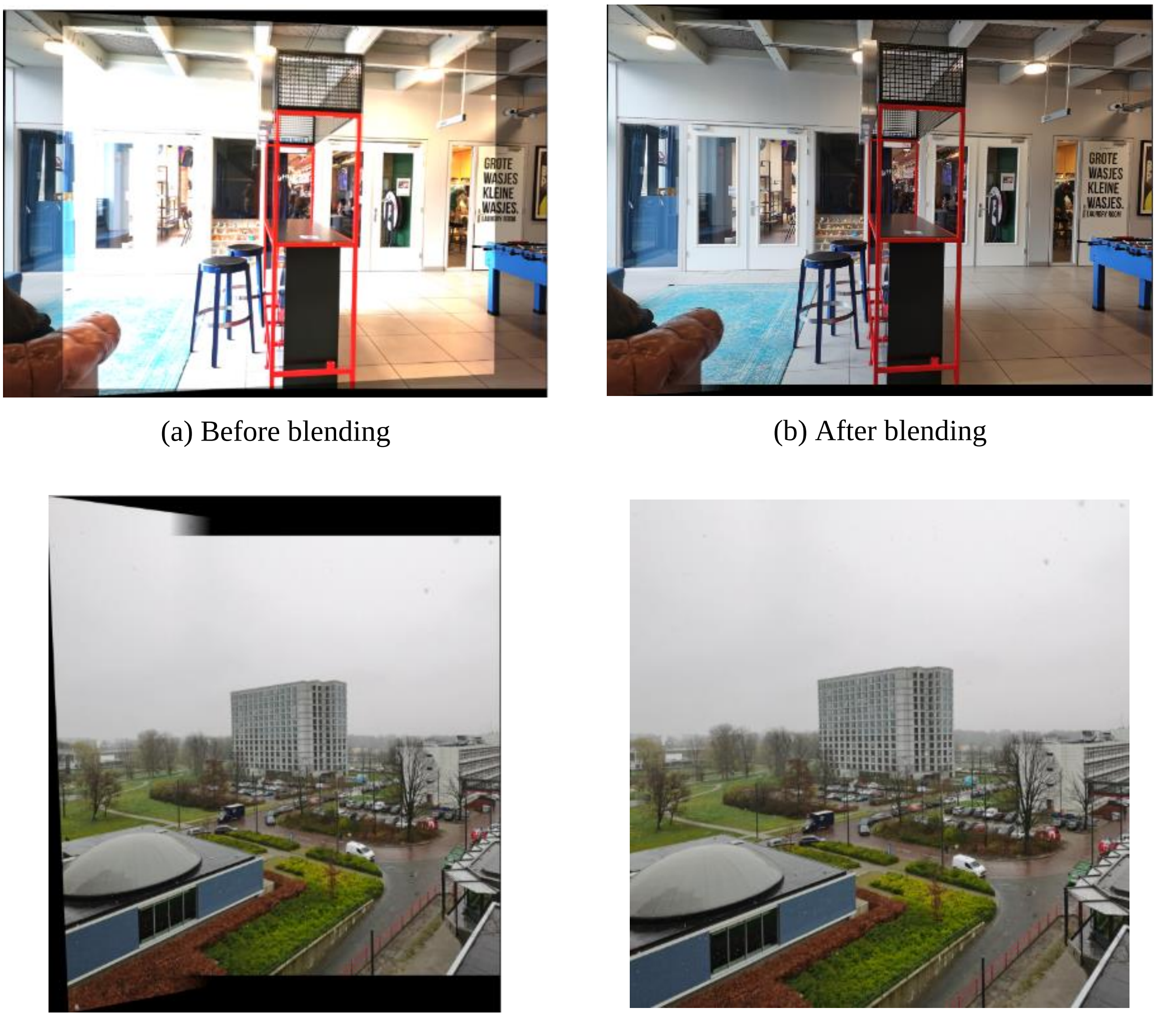


Figure 8. Feature matching results (indoor images set 1).

3.2.3. Image stitching

To obtain a well-stitched image which clear, smooth edge and high resolution, we apply image blending and cropping techniques to enhance image quality. Figure 9 shows the results of comparison with whether image blending and cropping are conducted.



(a) Before blending

(b) After blending

(a) Before cropping

(b) After cropping

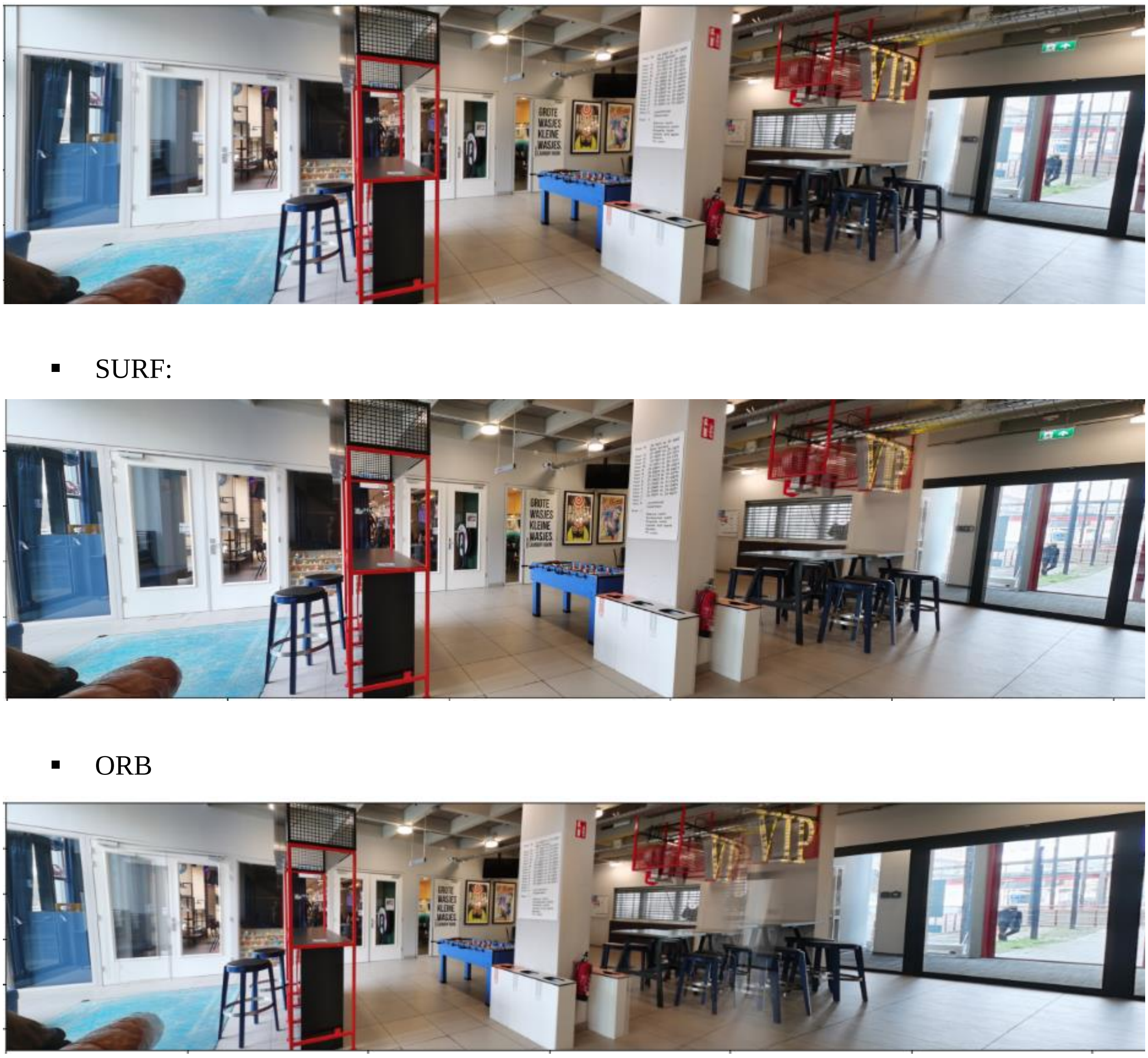
Figure 9. Image blending and cropping results (indoor images set 1).

The results shows that image turns to smooth, because the total weight of all pixels in final images equals one after blending, instead of adding two images directly. It is obvious that black edge is eliminated after cropping, indicating that the four corners of the final image have been correctly identified.

3.2.4. Panorama results

Following the image stitching process described in Table 4, the panorama results using different detection and extraction algorithms and matching methods are shown in Figure 10.

- SIFT



- SURF:

- ORB

Figure 10. Panorama results (indoor images set 1).

From the results, we can see that the SIFT performs better than SURF and ORB, and SURF results better than ORB. The different of the 3 methods is reflected by the stitching result of the VIP area among the 3 panoramas.

3.2.5. Comparison and analysis of results

In this experiment, the feature matching accuracy percentages of four feature matching methods were analyzed. The accuracy percentage is calculated as (number of correct matches / total number of matches) * 100. RANSAC was used to identify correct matches and incorrect matches. Meanwhile, the processing time is measured to compare the efficiency of different algorithms. The data used consisted of the first two images from Indoor images set 1, FLANN matcher is used to match points. The table records the number of feature points and matched points searched in images 1 and 2, the number of correct matches after processing with the RANSAC algorithm, as well as the accuracy and processing time.

Table 5. Compares the feature matching accuracy percentages of four algorithms (the number of searched feature points was controlled).

	feature points (first image)	feature points (second image)	Match points	Correct matches (After ransac)	Matching rate	processing time
Harris (SIFT descriptors)	5000	5000	2543	1439	56.58%	2.6307s
SIFT	5001	5000	1426	641	44.95%	3.3216s
SURF	4713	5145	1425	735	51.58%	4.2654s
ORB	5000	5000	657	159	24.20%	2.0938s

Table 6. Compares the feature matching accuracy percentages of four algorithms (the number of searched feature points was not controlled).

	feature points (first image)	feature points (second image)	Match points	Correct matches (After ransac)	Matching rate	processing time
Harris (SIFT descriptors)	7036	5841	3312	1574	47.52%	3.6250s
SIFT	23668	20637	6088	2923	48.01%	23.5718s
SURF	35626	34833	10154	5406	53.24%	36.2369s
ORB	500	500	54	42	77.78%	1.5249s

In Table 5, For each algorithm, the number of searched feature points was controlled to be around 5000 to compare the search time. Harris with SIFT descriptors has best performance on matching rate (56.58%), followed by SURF (51.58%) and SIFT (44.95%). The matching rate of ORB only is 24.20%, which is lowest among four algorithms. While the ORB algorithm was the fastest in terms of processing time and SURF is slowest.

In Table 6, the number of searched feature points was not controlled in order to allow each algorithm to fully demonstrate its performance. It is obvious that ORB has fastest speed (1.5249s) and matching rate (77.78%). Compared with ORB performance in Table 6, it is more accurate when the number of features is small. Besides, Harris preforms better when feature points was controlled, which means given an appropriate threshold, the Harris corner detection algorithm can identify more accurate corners and eliminate some false detections.

Considering that the experimental results may be affected by factors such as image content, transformation, and noise, outdoor dataset was also tested (Table 7).

Table 7. Compares the feature matching accuracy percentages of four algorithms (Outdoor image set).

	feature points (first image)	feature points (second image)	Match points	Correct matches (After ransac)	Matching rate	processing time
Harris (SIFT descriptors)	21719	18852	4683	3144	67.14%	18.3387s
SIFT	74797	58616	7022	6230	88.72%	185.1894s
SURF	54856	56016	6332	5038	79.56%	78.6469s
ORB	500	500	57	44	77.19%	1.6513s

In the experimental results, SIFT took the longest time but had the highest accuracy, SURF was more than twice as fast as SIFT while maintaining a certain level of accuracy, the corner detection accuracy was not high but the processing time was greatly reduced, and ORB remained the fastest detection algorithm with the least number of detected feature points.

Overall, using SIFT and SURF can find more feature points and consume more time. However, if a sufficient number of feature points is needed to ensure the quality of image stitching, they can achieve higher accuracy than the ORB algorithm.

4. Conclusion

In this paper, SIFT, SURF, ORB, and Harris corner detection is employed as the techniques for feature extraction and matching. For feature matching, BF matching and FLANN matching is employed, then using the RANSAC algorithm to estimate the homograghy matrix, design transformation methods to align the images, and finally using image blending and cropping techniques to obtain a smooth and high-resolution panorama. In the above process, it is concluded that SIFT algorithm performs better than SURF and ORB in terms of image results. SIFT might be more suited for uses where high precision is essential and when computing expense is not a limiting factor. Meanwhile, SURF might be more suited for real-time applications where both accuracy and speed are crucial. ORB may be preferred over SIFT and SURF depending on the particular application requirements and limits, due to its higher processing speed and reduced memory requirements. In practical applications, the results may be affected by factors such as image content, transformation, and noise. Therefore, when choosing a suitable feature matching algorithm in practical applications, multiple factors need to be considered, such as algorithm performance, processing time, application scenarios, etc.

References

[1] Zhaobin Wang, and Zekun Yang. Review on image-stitching techniques. 2020, *Multimedia Systems* **26:** 413-430.

[2] Sánchez J, Monzón N, Salgado De La Nuez A. An analysis and implementation of the harris corner detector. 2018, *Image Processing on Line.*

[3] Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. 2014, *International Journal of Computer Vision* **60**, 91–110.

[4] Yan Ke and R. Sukthankar, PCA-SIFT: a more distinctive representation for local image descriptors, 2004. *IEEE Conference on Computer Vision and Pattern Recognition*. 137-149.

[5] Bay, H., Tuytelaars, T., Van Gool, L.: SURF: speeded up robust features. 2006 *European Conference on Computer Vision*. **3951**, 404–417.

[6] Herbert B., Andreas E., Tinne T. and Luc Van G.: Speeded up Robust Feature (SURF), 2008 *Computer Vision and Image Understanding,* **110 (3):** 346- 359.

[7] Utsav S., Darshana M. and Asim B.: Image Registration of Multi-View Satellite Images Using Best Feature Points Detection and Matching Methods from SURF, SIFT and PCA-SIFT 1(1): 2014 *European Conference on Computer Vision* 8-18.

[8] E. Rublee, et al. ORB: An efficient alternative to SIFT or SURF. 2011 *International conference on computer vision*, 1-11.

[9] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. Brief: Binary robust independent elementary features. 2010, *European Conference on Computer Vision*, 1-10.

[10] S. A. Bakar, X. Jiang, X. Gui, and G. Li, Image Stitching for Chest Digital Radiography Using the SIFT and SURF Feature Extraction by RANSAC Algorithm Image Stitching for Chest Digital Radiography Using the SIFT and SURF Feature Extraction by RANSAC Algorithm, 2020, *European Conference on Computer Vision 1-12*.

[11] Fischler, Martin A., and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. 2020 *Communications of the ACM* **24.6,** 381-395.

Analysis and application research on key issues of channel coding

Jiatai Huang

Glasgow school, University of Electronic and Science Technology of China, Chengdu, 611731, China

2020190904022@std.uestc.edu.cn

Abstract. Channel coding plays a crucial role in enhancing the reliability and efficiency of communication systems, particularly when transmission channels are disrupted by noise and interference. This paper presents an in-depth review of various channel coding techniques, their applications, and future research directions. Key topics discussed include prevalent channel coding methods, such as repetition codes, convolutional codes, LDPC codes, turbo codes, and polar codes. The paper also delves into the selection of suitable channel coding parameters and their applications in digital TV, mobile and satellite communications, unmanned aerial vehicle data links, speech communication, and underwater acoustic channels. Moreover, the paper explores the performance analysis and comparison of different channel coding techniques, shedding light on their strengths and weaknesses. Lastly, the paper identifies emerging trends and challenges in channel coding research, providing valuable insights for researchers and practitioners in the field of communication systems. By examining these techniques and future directions, this comprehensive overview aims to contribute to the development of more robust and efficient channel coding schemes for a wide range of communication applications.

Keywords: channel coding, error correction, modulation, convolutional codes, LDPC codes, polar codes, wireless communication, IoT, 6G, machine learning.

1. Introduction

Channel coding is a vital technique for boosting the reliability and efficiency of communication systems, especially when noise and interference disrupt transmission channels [1]. Since Shannon's groundbreaking work on information and coding theory in 1948, research on achieving reliable transmission over unreliable channels has persisted, with channel coding playing a central role [2].

The emergence and popularization of 5G communication further underscore the importance of channel coding [3]. This article investigates the specific applications of channel coding and the efficacy of various coding methods to prepare for future challenges and opportunities in the field [4].

Shannon's second coding theorem, articulated in his seminal paper "A Mathematical Theory of Communication," established that there exists a coding method capable of enabling transmission at a rate R nearly as high as the channel capacity C, while still maintaining an extremely low transmission error rate [5]. This theorem has inspired researchers to explore different encoding schemes that approach this theoretical limit, known as the Shannon capacity [6]. Essentially, Shannon's theory suggests that a coding scheme exists that can achieve exceedingly low error probabilities for transmission rates less than or equal to the channel capacity, and researchers continue to work towards developing such schemes. The quest for these optimal encoding schemes is an active area of research in information theory, as they hold the potential to revolutionize communication by enabling faster and more reliable data transmission [7].

Significant advancements have been made in linear Gaussian channel models, such as Turbo codes and LDPC codes [8]. However, developing encoding schemes that approach theoretical limits for wireless communication remains challenging due to the nonlinear, time-varying properties of wireless channels. Current global research efforts focus on enhancing coding techniques, such as LDPC, turbo, polar, and product codes, to address communication system challenges. Moreover, researchers investigate integrating multiple coding schemes to improve performance in the face of channel impairments [9]. In China, the development of customized channel coding techniques for domestic communication systems, including satellite and wireless communication, is an active area of exploration [10].

2. Introduction to basic knowledge

2.1. Fundamentals of channel coding

In communication systems, channel coding is a strategy implemented to bolster the reliability of data transmission when faced with noisy channels. It involves adding redundant information to detect and correct errors. Coding schemes vary in complexity and performance, with some adding simple redundancy, while others use more complex algorithms for higher error correction capabilities. The primary goal is to enable reliable communication with minimal errors, and researchers continue to explore new coding schemes to approach the theoretical limit of communication capacity. This improves the efficiency and reliability of communication systems, making them more suitable for modern applications.

2.2. Common channel coding techniques

2.2.1. Repetition code. Repetition code is a simple error-correcting code that works by transmitting each bit of a message multiple times. The sender duplicates each bit of the message, and the receiver performs a majority vote to determine the correct bit. The idea behind the repetition code is that the redundant bits can be used to detect and correct errors that may occur during transmission.

2.2.2. Convolutional codes. Convolutional codes operate by encoding an ongoing stream of input data through the utilization of a shift register and a collection of code coefficients. The shift register maintains a limited quantity of preceding input bits, shifting them in a temporal order as new input bits are introduced. The code coefficients serve to generate the output bits, which are subsequently transmitted over the communication channel. Upon reception, the decoder employs specific algorithms to ascertain the most probable sequence of input bits responsible for creating the received code sequence. By comparing the received code sequence to all potential input sequences, the decoder selects the sequence with the highest likelihood [1].

2.2.3. LDPC codes. LDPC (Low-Density Parity-Check) codes represent a category of linear error-correcting codes employed in digital communication and data storage systems. Their operation is grounded in the principle of parity-check matrices. The generator matrix of an LDPC code is characterized by its sparse nature, containing a minimal density of ones. When transmitting a message utilizing LDPC codes, the message undergoes encoding through multiplication with the generator matrix. Subsequently, the resulting codeword is conveyed over the communication channel. At the receiving end, the obtained codeword undergoes error checking by being multiplied with the parity-check matrix. This parity-check matrix assists in pinpointing the error locations within the received codeword [6].

LDPC codes are known for their excellent error-correction performance, low complexity, and design flexibility. They are widely used in various communication standards, including Wi-Fi, WiMAX, and DVB-S2. LDPC codes can achieve near-capacity error-correction performance with relatively low complexity, making them suitable for high-speed data transmission applications.

2.2.4. Turbo codes. Turbo codes, a class of error-correcting codes in wireless communication systems, are renowned for their exceptional data transmission and error correction capabilities. The operational principle encompasses three primary stages: encoding, interleaving, and decoding. Encoding employs parallel concatenation of convolutional encoders, while an interleaver reorganizes input data to minimize error correlation among parity bits. Decoding is an iterative procedure utilizing soft-input soft-output decoders, which exchange extrinsic information to ultimately reconstruct the original data sequence with elevated reliability. By employing parallel concatenated codes, interleaving codewords, and iterative decoding, turbo coding bolsters the reliability of digital communication systems, refining the decoding process and facilitating increased data rates.

2.2.5. Polar codes. Polar coding, a pioneering error-correcting code in digital communication systems, was devised by Erdal Arikan and employs binary tree structures and channel polarization to convert original data bits into highly reliable bits. The working principle is based on channel polarization, separating the original data bits into two sets: highly reliable bits and less reliable bits. A binary tree is constructed with the original data bits at the leaves, and the tree's structure is created using a recursive kernel matrix called the Arikan matrix. This matrix is applied iteratively to the input bits, generating intermediate nodes in the tree and eventually reaching the root node, which contains the combined reliability information of all original data bits. Through successive applications of the Arikan matrix, the channels become increasingly polarized, enabling efficient error correction by focusing on correcting the less reliable bits. Due to its low complexity and flexibility, polar coding has been adopted for 5G wireless communication, improving system reliability, efficiency, and data transfer rates, making it a significant advancement in the fields of information theory and communication systems [7].

2.3. Channel coding parameter selection method

Selecting appropriate channel coding parameters is vital for optimizing error-correcting codes in digital communication systems. Various methods can be employed, such as simulation-based, analytical, and machine learning techniques. Simulation-based approaches assess coding schemes' performance over channels, while analytical models use statistical channel models to predict performance. Machine learning algorithms examine extensive data sets to discern correlations between channel characteristics and coding scheme effectiveness. Typically, a combination of these methods is implemented to choose coding parameters that fulfill system requirements, enabling reliable communication even in environments with noise and interference.

3. Channel coding application case study

3.1. Channel coding in digital tv signal transmission

In digital TV signal transmission, channel coding techniques such as forward error correction (FEC) [5], convolutional coding, and turbo coding are essential for maintaining signal reliability amidst noise and interference. FEC adds redundancy to the data stream for error detection and correction, while convolutional and turbo coding use advanced algorithms to encode data, bolstering its resilience against noise and interference. Preserving signal quality in digital TV transmission is critical due to potential factors like atmospheric conditions and electrical interference that may compromise signal integrity.

3.2. Channel coding in mobile communications

Channel coding in mobile communications is essential for reliable data transmission across noisy wireless channels, utilizing techniques like convolutional, turbo, LDPC, and polar coding to introduce redundancy for error detection and correction. The selection of a coding method depends on factors such as data rate, latency, and error-correction requirements. When designing a coding scheme, it is crucial to choose the appropriate code rate, word size, and decoding algorithm. Modern mobile systems, including 4G and 5G, frequently employ adaptive channel coding schemes, adjusting parameters according to channel conditions to maintain high-quality connections and optimize resource use.

3.3. Channel coding in satellite communications

DVB-S2 satellite systems employ a forward error correction system using cascaded BCH and LDPC codes, characterized by a sparse parity-check matrix for efficient decoding algorithms with near-Shannon-limit performance. This approach reduces demodulation thresholds and allows longer codes with manageable decoding complexity. Advanced Satellite Broadcasting System (ABS-S) utilizes LDPC codes with stronger error correction capabilities and shorter frame lengths, eliminating the need for BCH codes. Turbo codes, adopted in satellite communications due to their excellent error-correcting performance, employ parallel concatenation of convolutional codes with an interleaver for iterative decoding. Polar codes, a newer capacity-achieving class of channel codes with low encoding and decoding complexity, are being considered for future satellite communication systems. As encoding methods evolve, digital satellite radio and television systems continue to improve [3].

3.4. The data link of unmanned aerial vehicle

China is rapidly advancing its UAV data link systems, focusing on channel encoding technology to enhance high-speed communication. Comparing LDPC codes, convolutional codes, and Turbo codes in Rayleigh and Rician fading channels, simulations show LDPC codes outperform others in both short and long code scenarios. By implementing LDPC codes, UAV data link systems can achieve higher data rates, improved error-correction capabilities, and better overall performance. This is essential for real-time monitoring and control in applications such as surveillance, disaster management, and agriculture. As the UAV industry grows, optimizing advanced channel encoding techniques remains crucial for UAV data link systems' success.

5G technology encompasses three main use cases: enhanced mobile broadband (eMBB), ultra-reliable and low latency communication (URLLC), and massive machine-type communication (mMTC). In the eMBB context, a code excelling in performance for long block lengths and fast decoding is desired. To meet this requirement, the 3GPP has already adopted LDPC codes for data channels and Polar codes for control channels. For the URLLC scenario, a code with small packet sizes, low code rates, no error floor, resilience over fading channels, manageable decoding complexity, and a user-friendly rate matching mechanism is required. Turbo codes are not suitable for this scenario due to their increased decoding complexity and subpar performance at low code rates with shorter block lengths. LDPC codes also underperform for short block lengths and low code rates. In contrast, Polar codes provide exceptional performance with a range of code rates and code lengths through straightforward puncturing and code shortening mechanisms. They can achieve 99.999% reliability, feature low-complexity decoding algorithms, and consume minimal power, making them well-suited for both URLLC and mMTC scenarios [8]. As a result, Polar codes emerge as a strong candidate for 5G NR scenarios, particularly in the context of URLLC and mMTC use cases. Their unparalleled performance, adaptability in code rates and code lengths, reduced decoding complexity, and low power consumption render them an ideal solution for the demanding requirements of 5G networks

3.5. Speech communication

In speech communication, polar code is a suitable channel coding scheme for speech communication due to their capacity-achieving property, low complexity, and ability to efficiently use the channel. The structured nature of speech signals allows Polar codes to select only noiseless channels, reducing the probability of error and improving efficiency. Additionally, Polar codes can be constructed with various code rates and lengths to meet the specific requirements of the speech communication system.

Numerical simulations demonstrate that Polar codes outperform LDPC codes over both AWGN and Rayleigh channels, making them a promising choice for speech communication applications.

3.6. Underwater acoustic (UWA) channels

Underwater acoustic (UWA) channels present significant challenges for wireless communication due to their rapidly changing nature, extended multipath delays, and substantial frequency-dependent attenuation. Ensuring effective error-correction coding is essential for UWA communication systems, particularly when dealing with short or medium-length bit inputs. Various coding techniques have been investigated, including Convolutional Codes, RS block codes, Turbo codes, and Non-binary LDPC coding [4].

Protograph LDPC codes are recommended for implementation in UWA communication systems due to their linear encoding and decoding complexity, coupled with their uncomplicated structure. These attributes lead to a reduction in overall channel coding time, facilitating real-time and reliable UWA communication. As a subset of LDPC codes, Protograph-based LDPC codes possess swift encoder structures and have found applications in ultra-wideband (UWB) communication, space communication, and partial response (PR) channels in magnetic recording systems. These codes demonstrate exceptional error-correction capabilities in channels with inter-symbol interference (lSI) and have been proven to outperform MacKay's LDPC codes in UWA channels. The introduction of Protograph LDPC codes aimed to enhance encoding and decoding speed while maintaining high error-correction performance in UWA communication.

4. Performance analysis and comparison of channel coding

4.1. Performance comparison of different channel coding techniques

Performance comparison of different channel coding techniques is crucial for selecting the most suitable method for enhancing reliability and efficiency in communication systems. Convolutional, turbo, LDPC, and Reed-Solomon coding each have their own advantages and drawbacks. Assessing their performance involves evaluating error-correction capabilities, coding efficiency, complexity, and decoding delay. Coding gain measures error-correction, with higher values indicating improved SNR and better performance. Coding efficiency depends on the rate, with higher rates signifying efficient data transmission but reduced error-correction capabilities. Complexity reflects the computational resources required, with lower complexity being preferable. Decoding delay evaluates the time needed to decode received data, with shorter delays being more desirable [3]. Convolutional coding offers simplicity and low complexity but lower coding gain. Turbo and LDPC coding provide higher coding gain and efficiency but with increased complexity, making them suitable for high-data-rate applications. Reed-Solomon coding, a non-binary cyclic code, excels in error detection and correction for data storage and broadcast applications. The selection of an appropriate coding technique depends on specific system requirements and desired trade-offs between performance factors.

4.2. Comparison of channel coding and modulation methods

Channel coding and modulation are two important techniques used to enhance wireless communication. Channel coding adds redundancy to the transmitted data for error correction, while modulation adjusts the signal to optimize transmission.

A key factor in comparing channel coding and modulation is their impact on system performance, including error rate, spectral efficiency, complexity, and power consumption. Convolutional coding is a simple technique that provides moderate error correction, while turbo and LDPC coding offer higher error correction at the cost of increased complexity. Modulation schemes vary in spectral efficiency and error correction capabilities, with higher-order schemes providing greater spectral efficiency but requiring more signal-to-noise ratio (SNR) and being more susceptible to errors. By comparing and selecting the appropriate combination of channel coding and modulation techniques, wireless communication systems can achieve optimal performance and efficiency.

4.3. BER Analysis in channel coding applications

BER analysis is used to evaluate the performance of different channel coding schemes by measuring the rate of bit errors in received data and comparing it to the theoretical error rate predicted by the coding scheme [6]. The BER depends on factors such as the SNR of the received signal, the coding rate, and the complexity of the decoding algorithm. Higher SNR results in lower BER, higher coding rate results in higher spectral efficiency but also higher BER, and more complex decoding algorithms provide better error-correction capabilities but require more computational resources. BER analysis can help in selecting the appropriate coding scheme for a given application and optimizing parameters to achieve a specific BER at a lower SNR, improving overall performance of the communication system.

5. Future development of channel coding

5.1. Application prospects of channel coding

In IoT applications, channel coding plays a critical role in enabling reliable data transmission over noisy wireless channels. For instance, in smart city applications, channel coding is used to transmit sensor data and enable remote control of devices such as traffic signals and streetlights.

In agriculture, it can optimize irrigation, fertilizer application, and pest control, leading to increased crop yields and reduced waste. In connected vehicles, it improves vehicle safety, traffic flow, and enables autonomous driving. About smart energy, channel coding facilitates real-time monitoring and efficient distribution of energy resources. In home automation, it enables remote control and monitoring of smart home devices for convenience, security, and energy efficiency. In addition, it helps track pollution levels, predict natural disasters, and implement mitigation strategies in environmental monitoring. For, retail and supply chain, it also contributes to enhanced operational efficiency and customer experience through inventory tracking and automated checkout systems. Furthermore, channel coding is also used in wearable devices and medical implants, where reliability and low power consumption are crucial factors. In 6G communication, channel coding will be essential for high-speed, low-latency communication required by emerging applications such as virtual and augmented reality, autonomous vehicles, and tactile internet. To meet stringent performance requirements, advancements in channel coding will include new modulation and coding techniques, enhanced channel estimation methods, and integration with technologies like Massive MIMO and beamforming. Machine learning and artificial intelligence will play a significant role in optimizing channel coding schemes, adapting to changing conditions and minimizing complexity. Additionally, 6G channel coding will need to support distributed processing in edge computing, ensure data security and privacy, and adapt to heterogeneous networks comprising satellite, terrestrial, and aerial systems [8]. These advancements will facilitate seamless connectivity and optimal performance in the 6G era.

5.2. Future development direction of channel coding technology

Two significant trends in channel coding research are the development of low-latency coding schemes and coding schemes optimized for specific communication channels. Low-latency coding schemes are critical for real-time applications, and researchers are exploring new techniques such as polar codes, LDPC codes, and turbo codes. Coding schemes optimized for specific communication channels, such as wireless and satellite channels, have also been developed to achieve maximum performance. The trend is towards more integrated and efficient communication systems where source and channel coding are optimized together. Future developments may focus on finding more effective ways to combine source and channel coding and incorporating optimization techniques such as machine learning and artificial intelligence.

Adaptive channel coding, including rate-compatible LDPC codes, is gaining significance in addressing fluctuating channel conditions in wireless communication systems. Researchers are also exploring the potential of turbo codes and polar codes to enhance system reliability and efficiency. With the ongoing evolution of wireless communication technology, it is anticipated that more sophisticated adaptive coding schemes will emerge to satisfy the requirements of future systems.Channel coding plays a crucial role in modern communication systems by enabling reliable transmission of digital information over noisy channels. However, challenges such as efficient error-correcting code design, balancing coding rate and decoding complexity, and optimizing coding schemes for emerging applications such as wireless and quantum communication systems still need to be addressed. Furthermore, the trend towards machine learning-based communication systems presents new opportunities and challenges for channel coding research. To meet these challenges, researchers need to collaborate across multiple disciplines and develop innovative coding schemes that can meet the growing demands of modern communication systems. The future development of channel coding technology in wireless networks may involve further optimization of existing coding schemes, development of new coding schemes, integration of coding and modulation techniques, and exploration of new coding techniques such as machine learning-based coding and quantum coding [7].Future research in channel coding will concentrate on addressing the challenges posed by emerging wireless applications, focusing on optimizing existing coding schemes and developing new ones with improved error correction performance. Integrating coding and modulation techniques, along with exploring novel approaches such as machine learning-based coding and quantum coding, will play a pivotal role in meeting the demands of future wireless networks. Collaboration across multiple disciplines will be essential to drive innovation and create cutting-edge coding schemes that can keep up with the rapidly evolving landscape of wireless communication.

6. Conclusion

Channel coding is a critical aspect of contemporary communication systems, enabling digital information to be reliably transmitted over noisy channels. The field of channel coding has witnessed significant advancements in recent decades, with numerous error-correcting codes proposed and extensively employed in practice. However, challenges remain, including designing efficient codes that counteract channel noise and interference while maintaining low decoding complexity. Furthermore, the implementation of channel coding in emerging applications, such as wireless and satellite communications, presents novel challenges and opportunities. The trend towards machine learning-based communication systems also introduces new possibilities for channel coding research. In summary, ongoing research and development in channel coding are essential to satisfy the requirements of modern communication systems and emerging technologies.

References

[1] Arora K, Singh J, Randhawa Y S. A survey on channel coding techniques for 5G wireless networks[J]. Telecommunication Systems, 2020, 73: 637-663.

[2] Indoonundon M, Pawan Fowdur T. Overview of the challenges and solutions for 5G channel coding schemes[J]. Journal of Information and Telecommunication, 2021, 5(4): 460-483.

[3] Kurka D B, Gündüz D. DeepJSCC-f: Deep joint source-channel coding of images with feedback[J]. IEEE Journal on Selected Areas in Information Theory, 2020, 1(1): 178-193.

[4] Dai J, Wang S, Tan K, et al. Nonlinear transform source-channel coding for semantic communications[J]. IEEE Journal on Selected Areas in Communications, 2022, 40(8): 2300-2316.

[5] Choi K, Tatwawadi K, Grover A, et al. Neural joint source-channel coding[C]//International Conference on Machine Learning. PMLR, 2019: 1182-1192.

[6] Bourtsoulatze E, Kurka D B, Gündüz D. Deep joint source-channel coding for wireless image transmission[J]. IEEE Transactions on Cognitive Communications and Networking, 2019, 5(3): 567-579.

[7] Farsad N, Rao M, Goldsmith A. Deep learning for joint source-channel coding of text[C]//2018 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, 2018: 2326-2330.

[8] Choi K, Tatwawadi K, Grover A, et al. Neural joint source-channel coding[C]//International Conference on Machine Learning. PMLR, 2019: 1182-1192.

[9] Zarcone R, Paiton D, Anderson A, et al. Joint source-channel coding with neural networks for analog data compression and storage[C]//2018 Data Compression Conference. IEEE, 2018: 147-156.

[10] Balsa J, Domínguez-Bolaño T, Fresnedo Ó, et al. Transmission of still images using low-complexity analog joint source-channel coding[J]. Sensors, 2019, 19(13): 2932.

Research on panoramic mosaic of multiple images

Haoxuan Liu

University of Science and Technology Beijing, Beijing, China

42021214@xs.ustb.edu.cn

Abstract. With the continuous improvement of the manufacturing process of mobile phone cameras, the demand for panoramic photos by the industry and the public is not limited to ordinary cylindrical panoramic images, so a series of accurate panoramic image stitching technologies have been derived. However, these techniques often fail to take into account the stitching effect and speed, and there is still a technical gap in current multi-image panoramic stitching techniques. This article mainly compares the accuracy and efficiency of various feature extraction algorithms to select the most suitable algorithm for obtaining feature point pairs and estimating the homography matrix. Finally, a dual band hybrid algorithm is used to distort and fuse multiple images into panoramic photos. The results show that the panoramic images generated with multiple images using the algorithm provided in this paper are very good, even though there is a little bit of light and dark interlacing and ghosting.

Keywords: multi-image, panorama mosaic, feature extraction, image blending.

1. Introduction

With the rapid development of computer graphics related technologies, the requirements for new technologies in various fields are also increasing. Panoramic stitching integrates image registration, image fusion and other technologies. It also involves optics, computer graphics, stereo vision and other scientific fields. Panoramic stitching is a hot research direction at present, and its development quality plays a vital role in medical, material, military, aerospace and other fields. At present, the panoramic image shooting function provided by mobile phones or cameras only generates a "flat view" of the scene, without stereoscopic and realistic sense, so the development of panoramic splicing technology will also bring benefits to photography. So far, many algorithms have been developed in the field of panoramic mosaic, but these algorithms cannot give consideration to both speed and accuracy, and there is also great room for improvement in image fusion. Therefore, the research and improvement of panoramic image mosaic technology is of both academic and practical significance.

The image mosaic requires numerous processing techniques, including boundary stitching, mixing, transform estimates across pictures, image warping into mosaic surfaces, and so forth [1]. Panoramic mosaic technology processes multiple images with overlapping parts, estimates the transformation matrix (a.k.a. Homography) of the coordinate system of two images through the coordinate matching of feature points between adjacent images, and then uses the transformation matrix to "distort" one image to another image, so that we can intuitively see that the field of vision of the entire image is broadened. Because the material pictures of the panorama are taken at the same position with different rotating states of the camera, it has a strong stereoscopic sense and excellent visual experience.

This paper mainly studies the selection and optimization of feature point matching and image fusion technology in panoramic mosaic algorithm, so as to realize a set of panoramic mosaic algorithm pipeline with high efficiency and accuracy, and highlight the problem of how to build a smoother and more realistic panoramic image with more material pictures. First, we will introduce the process of feature matching using SIFT. Next, we will use the Direct Linear Transform [2] algorithm to calculate the Homegraph and use the Random Sample Consensus [3] algorithm to filter the interior points to optimize the calculation results of Homegraph. After warping each image to the plane of the central image, select the appropriate fusion algorithm for image fusion to make the panorama look more natural.

2. Methods

2.1. *Data collection*

The experimental data is a group of pictures taken by the camera of the mobile phone around the horizontal line. In the process of obtaining the material, try to keep the camera position unchanged (the optical center of the camera corresponding to each picture almost coincides), only change the rotation angle, and shoot at the same time to ensure that the brightness and light line remain unchanged, otherwise it will lead to serious blurring effect of the mosaic panoramic image.

The process of obtaining data in this experiment is low cost and convenient, and the quality of the data obtained is relatively good due to the advantage of the advanced camera function of the current smart phone. Figure 1-3 are a part of the captured images.

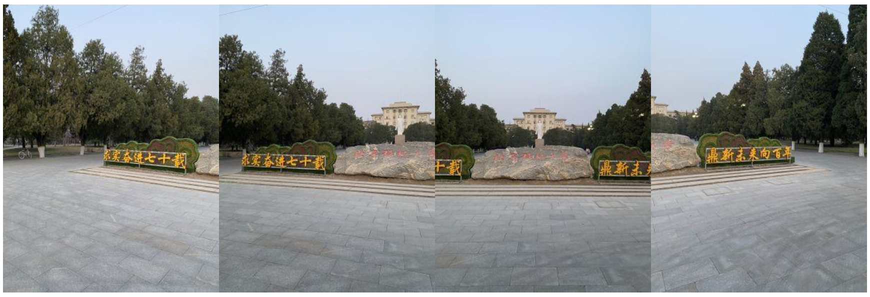


Figure 1. Source Images (Stone, 13 in total).

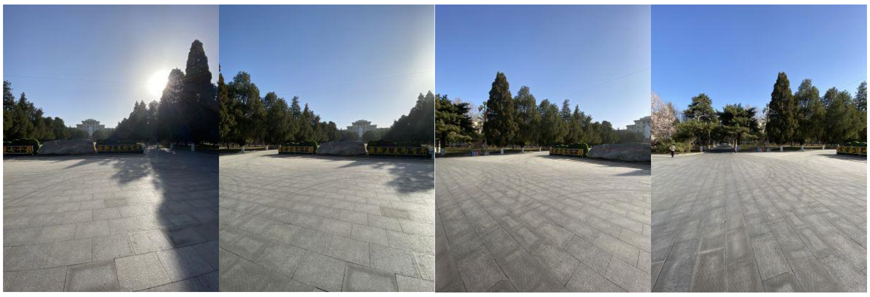


Figure 2. Source Images (Sunshine, 5 in total).

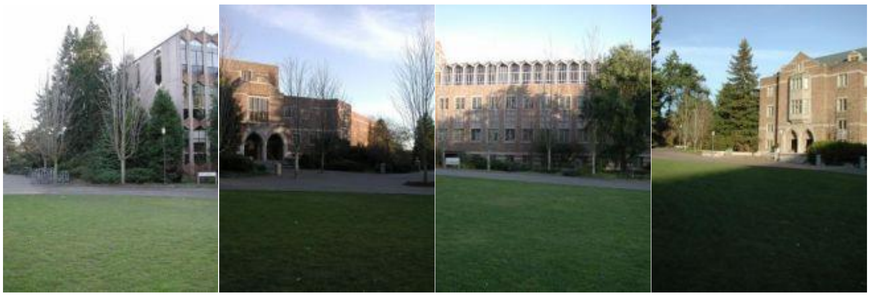


Figure 3. Source Images (Different Brightness, 15 in total).

2.2. Algorithm

The majority of image stitching algorithms follow a similar process: after estimating the warping or transformations needed to align the overlapping images, the aligned images are composited into a single canvas [4]. More specifically, it can be divided into the following steps: feature extraction and matching, Homegraph computation and RANSAC optimization, Homegraph computation from each image to the central image, image mosaic and fusion. The specific process is shown in Figure 4.

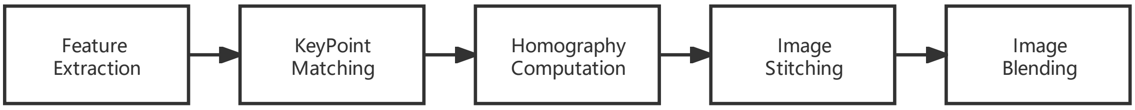


Figure 4. General process of panoramic splicing.

2.2.1. Feature extraction and matching

- SIFT Algorithm

SIFT is a feature extraction algorithm with scale invariance, and it can be concluded that SIFT algorithm has the following advantages: First off, despite changes in rotation, zoom, and illumination in the input photos, the inclusion of invariant features permits accurate matching of panoramic image sequences [5]. Table 1 shows the main steps of the SIFT algorithm.

SIFT has good stability and invariance, which means that it can ignore interference of affine transformation and noise. The use of SIFT algorithm in the multi-photo panoramic mosaic project also benefits from its distinguishing information quickly and accurately and generating a large number of eigenvectors. Figure 5 shows the generation process of SIFT descriptors.

Table 1. SIFT detector algorithm.

SIFT algorithm
1) Scale-space extremum detection: search for image positions on all scales. Identify potential points of interest that are invariant to scale and rotation by means of Gaussian differential functions.
2) Key point localization: At each candidate location, the location and scale are determined by fitting a fine-grained model. Key points are selected based on their stability. Low-contrast candidates and edge candidates are eliminated in the keypoint localization step.
3) Orientation determination: One or more orientations are assigned to each keypoint location based on the local gradient orientation of the image. All subsequent operations on the image data are transformed with respect to the orientation, scale and position of the keypoints, thus providing invariance with respect to these transformations.
4) Key point description: The gradients local to the image are measured at selected scales in the neighborhood around each key point. These gradients are transformed into a representation that allows for relatively large local shape distortions and illumination changes.

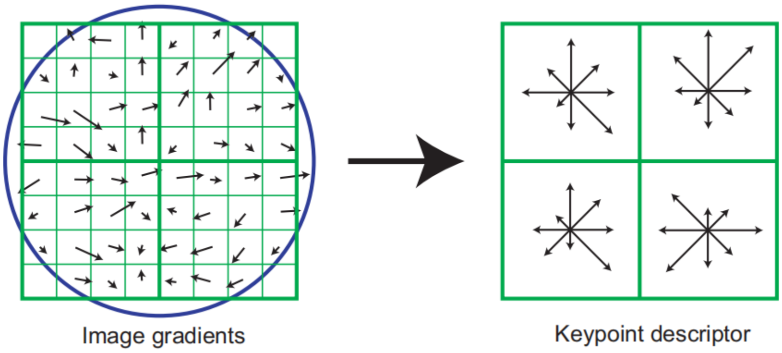


Figure 5. Keypoint descriptor created by gradient magnitude and orientation.

- KnnMatch Algorithm

This paper uses the knnMatch algorithm of k=2 as the descriptor matching algorithm. It finds the two feature points closed to the original image feature points, and only when the Euclidean distance between the two feature points is less than a certain value, will the matching be considered successful.

2.2.2. Homography computation and optimization

Although the cylindrical coordinate projection method is relatively easy to realize in the link of mosaic and image fusion, it is often only applicable to pictures that are far away from the camera and pictures that do not require strong perspective effect. Therefore, in this section, the method adopted by the project is to calculate the Homegraph matrix between adjacent pictures, and "twist" the pixel points in all pictures into the same plane through Direct Linear Transform coordinate transformation, as to form a panoramic view with strong perspective effect.

- Direct Linear Transform

Figure 6 describes the relationship between the coordinates of a point on the plane X_{π} on different projection planes. According to the relevant knowledge of Homegraph $X_1 \equiv HX_2$, we can get the expression with the shape of $\mathrm{Ah} = 0$ through the transformation based on linear algebra. The A matrix is decomposed by SVD, and the column vector corresponding to the minimum singular value is obtained, which is the 9x1 column vector composed of the elements in the Homegraph matrix.

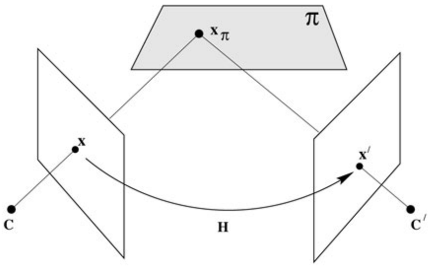


Figure 6. A homography H links all points x_{π} lying in plane π between two camera views.

- RANSAC algorithm

RANSAC is a frequently used algorithm for optimizing computer vision parameters. It can calculate the mathematical model parameters of data based on a set of sample data containing abnormal data, and obtain valid sample data. It is used to optimize the result of feature point matching in panoramic stitching, and to obtain the fundamental matrix in stereo vision. Table 2 shows the execution process of the RANSAC algorithm.

Table 2. RANSAC optimization algorithm.

RANSAC algorithm
1) Randomly select the number of sample points that can estimate the minimum number of points of the model (in the Homography Estimation algorithm, since the Homography matrix has 8 degrees of freedom, at least 4 pairs of feature points need to be selected here to generate the A matrix)
2) Calculate the model parameters (elements of the Homography matrix)
3) Bring all data into this model and count the number of inliers (inliers: use the Homograph matrix point to multiply the homogeneous coordinates of the point in the source image to obtain the estimated coordinates of the corresponding point in the target image, and then calculate the difference between the estimated coordinates and the accurate coordinates. If the difference is less than the threshold value, it means that the point is an interior point, otherwise it is an exterior point)
4) Based on the number of interior points, maintain an optimal (maximum number of inliers) model (i.e., the Homegraph matrix). When the number of inliers in the new loop exceeds the currently maintained number of inliers, update the maximum value of inliers and the estimated Homegraph matrix.
5) Repeat the above steps until the convergence (the number of inliers reaches the preset value) or the iteration reaches a certain number of times

2.2.3. *Image stitching and blending*

- Image stitching

One of these images must be transformed into a common field using the view transformation matrix H after the RANSAC model fitting comparison is finished. Give each of the photographs a different perspective in this case [6].

After computing the homography between each two adjacent images, it is necessary to warp each image onto the plane of the central image. The method used here is to dot multiply all the homography from the side image to the middle image to obtain the coordinate system conversion matrix from the side image to the central image, and then calculate the coordinates of each pixel point of the current image under the central image coordinate system through dot multiplication, as shown in Figure 7.

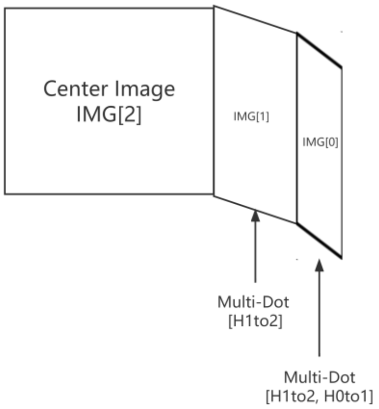


Figure 7. Warp each image into center image plane.

- Image blending

The image frames on the mosaic image are warped progressively and quickly in accordance with the camera's position. After the seam is found, due to factors such as image noise, illumination, exposure, and model matching error, direct image synthesis can produce significant edge traces at the stitching of the overlapping regions of the image. Adaptive multi-band blending is used to produce stitching results that are aesthetically acceptable. The weight matrix and the image's Laplacian pyramids are kept and modified accordingly [7].

The idea of the multi-band blending algorithm is to directly decompose the two images to be spliced into Laplacian pyramids, and merge the latter half with the first half. In the project, because the material images are all taken at the same time and place, the difference between the images is not significant, so the algorithm used is two-band blending.

In this work, a common mean value fusion algorithm is also used for comparison. By comparing effect and time consumption of the resulting images generated by the two sets of methods, it is determined which image fusion algorithm is more appropriate to use. Figure 8 shows the basic principles of the two fusion algorithms.

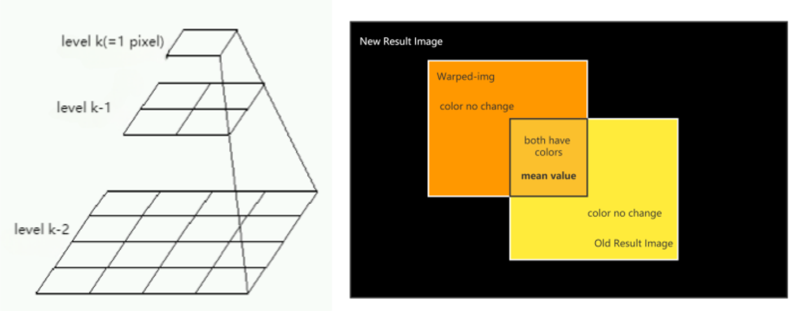


Figure 8. Two-band blending and uniform blending used in project.

3. Experiment

3.1. Platform introduction

In this report, the running environment of the project is Python 3.10.10 64-bit, and the external libraries are OpenCV-Python 4.7.0.22 and numpy 1.22.4. The IDE hosting the project is Visual Studio Code 1.71.2.

3.2. Results

3.2.1. Feature extraction and keypoint matching. The collection of image materials taken by mobile phones offers more detailed features that can be recognized by feature extraction algorithms due to the growing availability and sophistication of digital imaging technology (digital cameras, computers, and photo editing software), as well as the popularity of the Internet [8].

Table 3. The comparison results of image 1(Stone, 13 in total).

Detector	Key points	Match rate	Computation time(s)
SIFT	5237	36.10%	0.2834
ORB	6162	28.87%	0.3115
KAZE	4094	54.28%	1.0987
AKAZE	2544	57.57%	0.1611
BRISK	10973	26.59%	0.1685

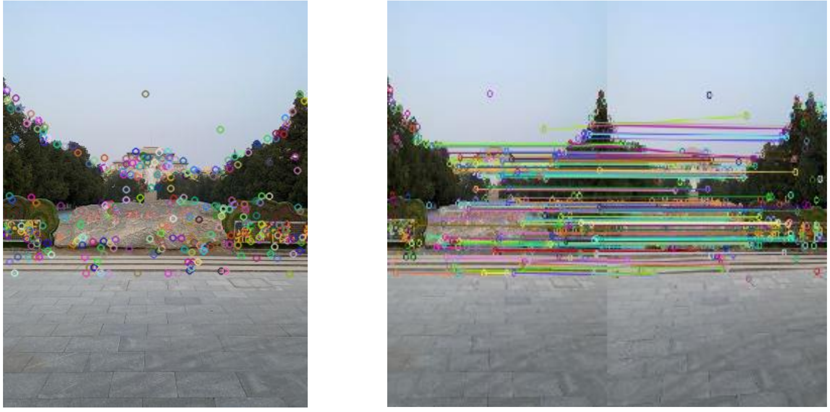


Figure 9. Feature Extraction and Keypoint Matching in image series 1.

Table 3 shows the number of feature points generated by different feature extraction and matching algorithms. The processing results from image series 1 (shown in Figure 9) indicate that the size relationship between the number of feature points generated by each image matching algorithm is: $\mathrm{BRISK} > \mathrm{ORB} > \mathrm{SIFT} > \mathrm{KAZE} > \mathrm{AKAZE}$. The matching results are: $\mathrm{AKAZE} > \mathrm{KAZE} > \mathrm{SIFT} > \mathrm{ORB} > \mathrm{BRISK}$. In terms of calculation time: $\mathrm{AKAZE} < \mathrm{BRISK} < \mathrm{SIFT} < \mathrm{ORB} < \mathrm{KAZE}$, but the calculation time of these algorithms in this dataset is mostly less than 2 seconds, so there is little impact on the project in terms of speed.

From the above experimental data, it can be concluded that algorithms that consume more time have higher accuracy, but the calculation time is not directly related to the number of feature points generated. However, algorithms with a larger number of feature points generally have a lower matching degree (some of the generation of feature points is invalid). Based on the above analysis, this multi-image panoramic mosaic project selects the SIFT algorithm with balanced indicators as the feature extraction algorithm, and its corresponding image matching algorithm is the KnnMatch algorithm in the FLANN library, where $\mathrm{K} = 2$.

3.2.2. Stitching. Figure 10 and Figure 11 are the results of direct stitching and the results obtained using two band blend (pixel RGB average fusion) processing, respectively. It can be clearly seen from the images that the results obtained using the direct stitching method have relatively obvious black seams, while using the two-band blend method can greatly eliminate the seams. The reason for this phenomenon is that the image fusion using pyramid decomposition is processed in different frequency bands, enabling a fusion effect that is closer to human visual characteristics [9].

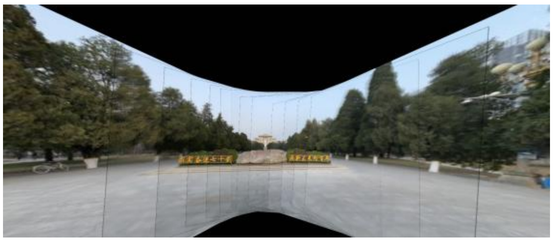


Figure 10. Results of direct splicing.

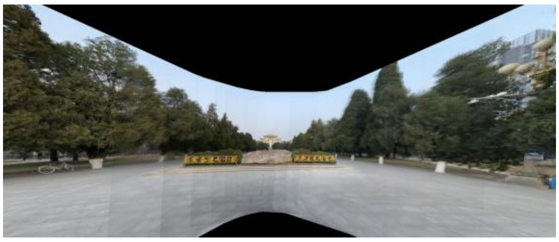


Figure 11. Results from using two-band blending.

Figure 12 shows the processing results of image materials taken under strong light intensity conditions. It can be seen that the multi-image stitching algorithm in this article has a relatively good processing effect for images with the same exposure (regardless of the intensity of light).

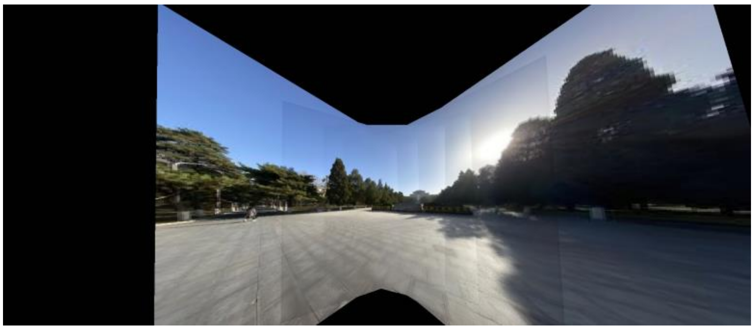


Figure 12. Splicing Results of Sunshine Datasets.

However, in Figure 13, we can see that there is a shading phenomenon between each material image. Although this is a small problem in image fusion, it also proves the correctness of the previous assumption: the camera used to take material images must have a consistent exposure. Overall, this issue has little impact on the multi-image mosaic algorithm in this article, and the overall effect is still acceptable.

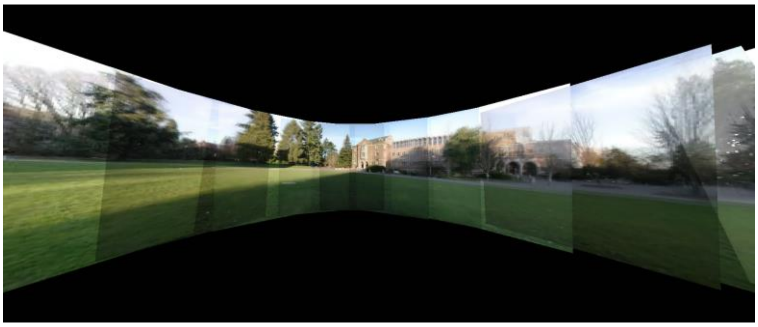


Figure 13. Splicing Results of Different Brightness Datasets.

In addition, we can see from the above panoramic mosaic results that there are still a small amount of ghosting and blurring between the various images, which is due to the slight displacement of the world coordinates of the camera used to collect the images. Images are related by a linear projective transformation known as a homography when the scene has a flat surface or when they were captured from the same angle [10]. Therefore, there are some ghosts in the panoramic image.

4. Conclusion

In this paper, we take the panoramic stitching of multiple images as the research goal, compare the differences between the effectiveness and efficiency of different algorithms in the feature extraction and matching, image fusion steps, and select the algorithm more suitable for doing panoramic stitching of multiple images. First, the SIFT algorithm is used to extract feature points between images and match them to generate feature point pairs. Then the RANSAC algorithm is used to reduce the estimation error of Homography. One image is selected as the central plane, and the other images are mapped to the plane of that central image. Finally, the two-band blending method is used to fuse the overlapping areas of these images. The results show that although fusion algorithms separate and fuse high-frequency and low-frequency information separately, shadow and ghosting effects may occur when there are objective interference factors such as significant changes in the brightness of the image, different camera exposures, and displacement of the device that captures the image. The relevant research results have certain reference value.

References

[1] Ha, S. J., Koo, H. I. , Sang, H. L. , Cho, N. I. , & Kim, S. K. . Panorama mosaic optimization for mobile camera systems.2007 *IEEE Transactions on Consumer Electronics,* **53(4)**, 1217-1225.

[2] Abdel-Aziz, Y. I., Karara, H. M. Direct linear transformation from comparator coordinates into object space in close-range photogrammetry. 1971 *American Society of Photogrammetry,* 1-10.

[3] Fischler, M. A. , Bolles, R. C.. Random sample consensus. 1981 *Communications of the ACM.*

[4] Zaragoza, J., Chin, T. , Tran, Q. , Brown, M. S. Suter, D.. As-projective-as-possible image stitching with moving dlt.2014 *IEEE Trans Pattern Anal Mach Intell*, **36(7),** 1285-1298.

[5] Brown, M. , & Lowe, D. G.. Automatic panoramic image stitching using invariant features. 2007, *International Journal of Computer Vision*, **74(1)**, 59-73.

[6] Qi Fengshan & Jiang Tingyao. (2016). Improvement of QR code image angle point detection method based on Harris. 2016 *Software Guide* **(05)**, 199-201.

[7] Liu, X. , Yu, H. T. , & Chen, B. M. Adaptive Weight Multi-Band Blending Based Fast Aerial Image Stitching and Mapping. 2018 *15th International Conference on Control, Automation, Robotics and Vision.* 1-12.

[8] Pan, X. , & Lyu, S. Region duplication detection using image feature matching. 2010 *IEEE Transactions on Information Forensics & Security,* **5(4)**, 857-867.

[9] Liu Guixi,&Yang Wanhai Image fusion method and performance evaluation based on multi scale contrast tower, 2001 *Journal of Optics*, **21 (11)**, 7

[10] Chiba, N. , Kano, H. , Higashihara, M. , Yasuda, M. , & Osumi, M. Feature-based image mosaicking, 1998 *Journal of Optics,* **19** (**15).**

Research on digital currency based on encryption technology

Yubo Zhang

School of Computer Science, Xi 'an Polytechnic University, Xi 'a Shaanxi,710000, China

42009040113@stu.xpu.edu.cn

Abstract. Digital currencies have become an increasingly popular topic of discussion in recent years. Digital currencies are virtual forms of currency that operate outside the traditional banking system. They are based on cryptographic technologies and are often decentralized, meaning they are not controlled by a central authority. The most well-known digital currency is Bitcoin, but there are many other types of digital currencies in existence. Digital currencies can be used to purchase goods and services online or transferred between users directly without intermediaries like banks. They have gained popularity due to their potential for increased security, transparency, and efficiency in financial transactions. In today's digital currency, a variety of digital currencies emerge in an endless stream, and crypto technology is also constantly developing to improve the security of digital currency payments. In section 2, this paper briefly introduces several common digital currencies and encryption algorithms, and in section 3, this paper introduces these typical digital currencies in detail through the analysis of representative literature. Bitcoin is mainly encrypted based on blockchain technology, and its encryption principle is mainly divided into three parts: public key encryption, hash function, and proof of work. Ethereum is a distributed blockchain platform with encryption principles similar to Bitcoin, including public key encryption and hashing algorithms. Ripple is a distributed cryptocurrency. Its encryption principle mainly adopts the public-private key encryption system. In terms of encryption technology, blockchain technology, the Hash algorithm and symmetric and asymmetric encryption are also popular encryption algorithms in digital currencies.

Keywords: bitcoin, Ethereum, ripple, blockchain, hash.

1. Introduction

Digital currency, additionally recognized as Cryptocurrency, is a cryptocurrency asset primarily based on cryptography. It makes use of cryptography to impenetrable transactions and manipulates the advent of new units, making it greater impervious and obvious than standard economic systems. The origins of digital currency can be traced back to the late 1990s when computer scientist Nick Szabo coined the term "bit gold". Bit gold was a precursor to Bitcoin and other digital currencies, relying on complex algorithms to verify transactions and prevent double-spending. However, it wasn't until 2009 that the first digital currency, Bitcoin, was introduced to the world. Bitcoin was created by an unknown person or group using the pseudonym Satoshi Nakamoto, and it quickly gained popularity among tech enthusiasts and libertarians who were drawn to its decentralized nature and potential to disrupt traditional financial systems [1].

In section 2 this paper briefly introduced the types of digital currencies and the classification of cryptographic technologies they use. This paper gave a brief background on Bitcoin, Ethereum, and Ripple and the cryptography they use. In terms of encryption technology, this paper also introduced three main encryption technologies: Blockchain, Hash, and Symmetric and asymmetric encryption. Blockchain encryption technology is a technology that uses cryptography to protect the data in the blockchain network. On the blockchain, every transaction needs to be encrypted, including the content of the transaction, the time of the transaction, the parties to the transaction, and so on. A blockchain is created by organizing this data into blocks, each of which contains an encrypted summary that is connected to the summary of the block before it uses a hash function. An algorithm known as hash encryption converts arbitrary-length inputs (messages) to outputs with set lengths. For message integrity checks, digital signatures, and password storage, hash encryption is utilized. The same key is used for both encryption and decryption in symmetric encryption, also referred to as private key encryption. Asymmetric encryption, commonly referred to as public key encryption, is a method that encrypts and decrypts data using two keys: the public key and the private key.

In section 3, this paper gives a detailed description of these currencies by analyzing their representative works. Bitcoin is a digital cryptocurrency that is issued and traded using blockchain technology. Bitcoin's features include decentralization, anonymity, and fixed circulation. It does not rely on banks or government agencies, can transact across borders, and can make fast money transfers in a short time. Ethereum is a distributed computing platform based on blockchain technology with smart contract capabilities that can support the development and operation of a variety of decentralized applications. At the heart of Ethereum are Ether, the cryptocurrency within the platform and the "fuel" charge for the execution of smart contracts on the platform. Ethereum realizes more flexible smart contract functions based on blockchain technology. Compared with other blockchain projects such as Bitcoin, Ethereum has more kinds of applications and more powerful scalability. Ripple is a digital cryptocurrency, as well as a decentralized payment protocol and open-source global payment network. Ripple uses blockchain technology and cryptocurrencies to help users quickly and easily transfer assets without the need for complex transfer processes and high fees. Ripple also offers an "XRP" digital currency that can be used to transfer money and make payments across borders. Compared to other digital currencies, Ripple focuses more on speeding up transactions and reducing transaction costs, as well as making payments more secure.

2. Preliminary

2.1. The introduction of digital currency

Bitcoin. The digital currency has been successfully implemented in recent years and has gradually swept the world, but the idea of digital currency has been proposed as early as the third technological revolution [1]. In 2008, bitcoin was proposed by the anonymous Satoshi Nakamoto, which aims to make money transactions free and safe by removing the control of money circulation by traditional financial institutions. Bitcoin used a generator of the computational proof which contains a system that utilizes a P2P distributed timestamp server as the chronological order of transactions [2]. Bitcoin's encryption technology employs two types of keys, namely a public key and a private key. The public key is used to determine the next owner of a Bitcoin during a transaction. The prior transaction's digitally signed hash is also included in the definition of the transaction [2]. The same encryption and decryption techniques are used by both private and public keys, but only certain communications can be decrypted by each key individually [3]. The public key is used for verification in a Bitcoin transaction, whereas the private key is used for signing. Figure 1 [1] shows the structure of a Bitcoin transaction on a blockchain.

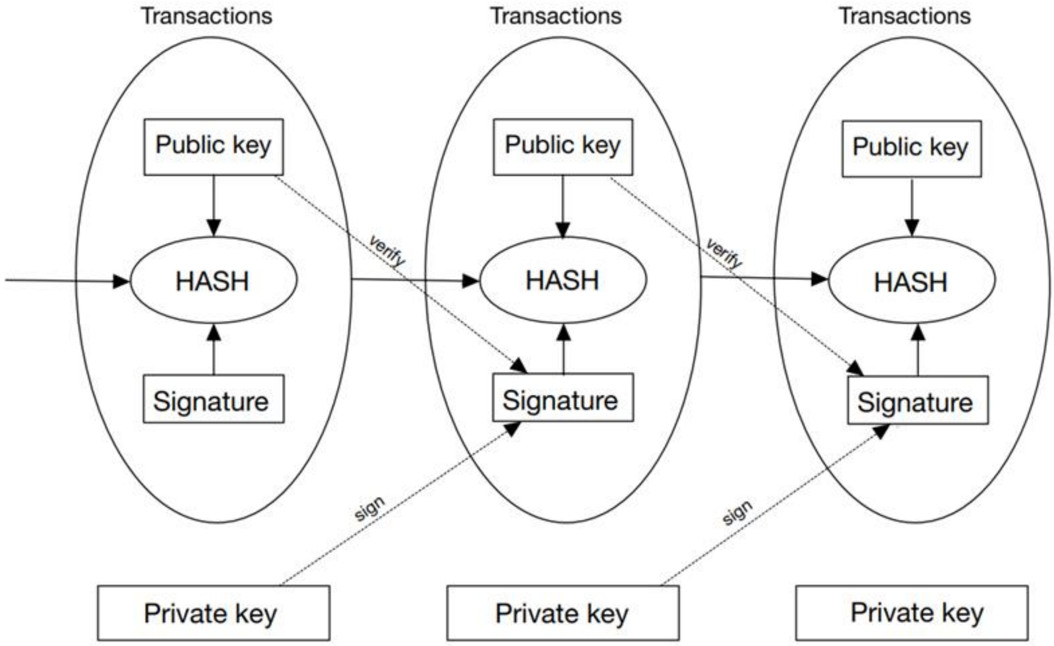


Figure 1. An Ethereum block with hashed transactions into a Merkle tree [1].

In addition, the Bitcoin network also adopts a distributed accounting system, that is, blockchain technology. Each node has a complete ledger that records the balance and transaction history of each user [4]. When a user initiates a new transaction, the node will verify it in its ledger. If the transaction is legal, the transaction will be packaged into a new block and broadcast to other nodes in the network. Other nodes will also validate this new block and add it to their ledger.

Ethereum. Ethereum is an open-source, distributed blockchain platform. It can not only support cryptocurrency transactions but also the execution of smart contracts [5]. Ethereum uses its virtual machine (EVM) to execute smart contract code and uses ether (ETH) as a token for cryptocurrency. Ethereum's blockchain technology is very similar to Bitcoin, and it is a ledger system maintained by distributed nodes. Each block contains transaction information, timestamps, and hash values of the previous block [5], [6]. The following is the specific process of Ethereum encryption: First, the user needs to generate a pair of public and private keys. The public key can be used to receive ether or smart contracts, and the private key is used for signed transactions or smart contract execution. When a user initiates a transaction or executes a smart contract, he needs to sign it with the private key to prove this is the operation that the user has sent. At this time, the private key is only held by the user, ensuring the security of the transaction or contract. Use the hash function to convert the signed transaction or contract into a string of numbers [1]. The string of numbers is called a transaction hash or a contract hash value. The transaction or contract hash value is broadcast to the entire network for validation and recording by other nodes [7]. Other nodes can decrypt the signature with the use of the public key and confirm the legality of the transaction or contract. If the verification is successful, the transaction or contract will be introduced to the block. Each block in the blockchain incorporates the hash fee of the preceding block, and this interconnected hash fee chain constitutes an immutable ledger system [8]. If someone tries to change any records in the block, the entire blockchain system will fail and need to be rebuilt [9]. An Ethereum block is shown in Figure 2 [1].

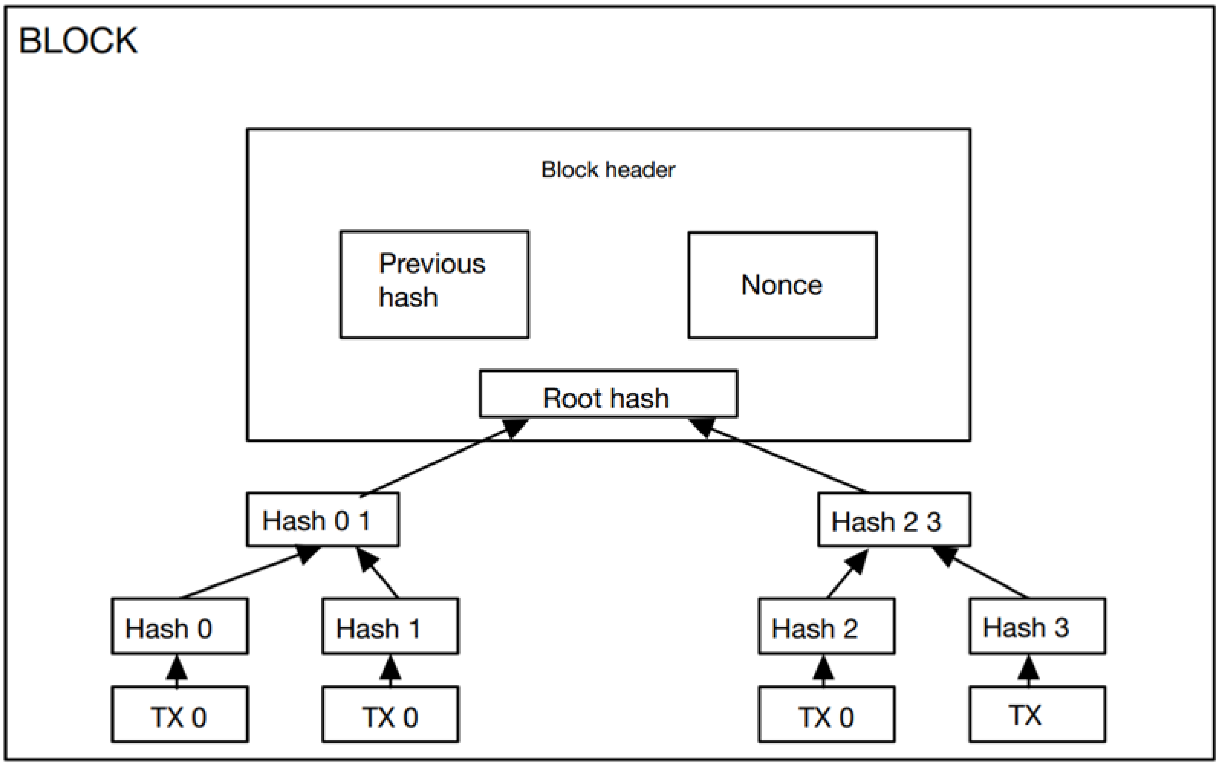


Figure 2. An Ethereum block with hashed transactions into a Merkle tree [1].

Ripple. Ripple [10] is a digital currency based on distributed ledger technology, also known as blockchain [11]. Its encryption principle mainly involves public key encryption and hash functions. Specifically, the encryption process of Ripple is as follows: the sender uses its private key to digitally sign the transaction information and broadcast it to the entire network [12]. The nodes in the network verify this information, including verifying the validity and availability of the digital signature, and whether the sender's Ripple balance is sufficient. If the verification is passed, the node adds the transaction information to the distributed ledger and performs hashing, that is, the transaction information is combined with a random number to generate a fixed-length hash value composed of numbers and letters [12]. The node then broadcasts the hash value to other nodes in the network so that other nodes can also verify the validity and availability of the transaction information.

2.2. The introduction of encryption techniques

The main technologies used in various digital currencies are blockchain technology, and the two core technical points of blockchain are consensus mechanism and cryptography. Next, this paper will introduce two types of cryptographic algorithms mainly applied in the blockchain, one hashing algorithm, and the other is symmetric encryption and asymmetric encryption algorithm.

Hash: Hash encryption is an encryption algorithm that compresses messages of any length to a certain length of the message digest. The simple precept of Hash encryption is to enter the plaintext information into the hash function, and then use a unique mathematical characteristic to convert messages of any size of the entry into a fixed-length output [13]. This output price is generally known as a hash cost or a summary. The simple facts shape of the hash on the blockchain is proven in Figure 3 [13].

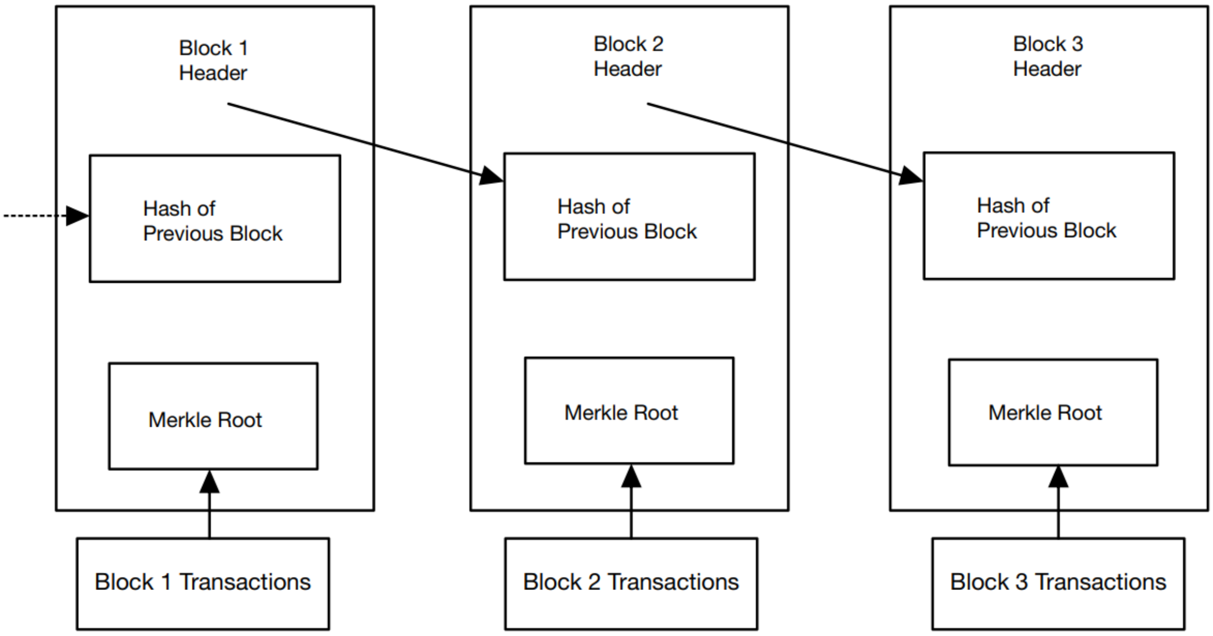


Figure 3. The basic data structure of the hash on the blockchain [13].

An important feature of the hash function is uniqueness, that is, the same input always produces the same output, and even if the input is only a small change, it will lead to different hash values. This hash value can be used to verify the integrity and consistency of the message, as any modification to the original data will result in different hash values. Another important feature of Hash encryption is irreversibility, that is, the original data cannot be reversed through hash values. This feature makes Hash encryption very useful when protecting sensitive information. Once the data is encrypted by hash, this cannot restore the data unless it has the same inverse function as the hash function.

Symmetric Encryption and Asymmetric Encryption Algorithm: Symmetric encryption refers to the encryption technique of encryption and decryption the use of the equal key, whether or not it is encryption or decryption, the equal key is used [14]. Common symmetric encryption algorithms consist of DES (Data Encryption Standard), 3DES (Triple Data Encryption Algorithm), AES (Advanced Encryption Standard), etc. The benefit of a symmetric encryption algorithm is that the encryption and decryption pace is quick and the encryption effectivity is high, however, the drawback is that the safety of the key is without problems threatened. Asymmetric encryption refers to the encryption approach of encryption and decryption the usage of distinctive keys, additionally acknowledged as public key encryption. In uneven encryption, the public key used for encryption can be made public, and the non-public key used for decryption needs to be stored secret [15]. Common uneven encryption algorithms consist of RSA (Ron Rivest, Adi Shamir, Leonard Adleman), ECC (Error Correcting Code), etc. The benefit of the uneven encryption algorithm is that the key protection is high, however, the drawback is that the encryption and decryption velocity is sluggish and the encryption effectivity is low.

Usually, symmetric encryption and asymmetric encryption are used together to give full play to their respective advantages, to achieve more efficient and secure encrypted communication. The comparison of common encryption algorithms of symmetric encryption and asymmetric encryption is shown in Table 1 [14], [15].

Table 1. The comparison of symmetric encryption and asymmetric encryption [14], [15].

Factors	Symmetric Encryption Algorithms			Asymmetric Encryption Algorithms
	DES	AES	3DES	RSA
Block Size	64 bit	128 bit	64 bit	Variable
Key Size	56 bit	128,192,256 bit	168 bit(k1,k2 and k3) 112 bit(k1 and k2)	Depends on the number of bits in the modulus n where n=p*q
Created By	IBM in 1975	Joan Daeman in 1998	IBM in 1978	Ron Rivest, Adi Shamir, and Leonard Adleman In 1978
Speed	Slow	Fast	Very Slow	Slowest
Rounds	16	9,11,13	48	1
Attack	Brute Force Attack	Side Channel Attacks	Brute Force Attack	Wiener’s Attack

3. Literature review

This article summarizes the representative literature on digital currency in recent years, as shown in Table 2 below.

Table 2. Literature Review.

Type and Representative work	Main idea	Advantage	Disadvantage
Bitcoin [1][2][3][6]	Encryption and transactions are conducted via blockchain	High level of security, quick transactions, Low transaction cost	Limited supply, high market price volatility, not being covered by insurance, and anonymity lead to illegal behavior
Ethereum [4][5][7][8][9]	The blockchain technology platform that supports smart contracts	High scalability, smart contract function, decentralization, high security, openness, and interoperability	Throughput limitations, user requirements for technical knowledge, high energy consumption, and systems prone to congestion and delays during peak trading periods
Ripple [10][11][12]	Use blockchain and consensus algorithms, through the internal ledger to achieve currency conversion of different values	Decentralized, fast money transfer, low transaction costs, multiple ways to transact, very flexible technology architecture	Transactions are public and can be tracked and traced; Subject to institutional review and regulation; they have Low security and credibility; High price

3.1. Bitcoin

The main focus of "Blockchain Technology, Bitcoin, and Ethereum: A Short Overview" is the concept of Bitcoin transactions. As defined, the Bitcoin ledger represents a state transition system that records the ownership status of every Bitcoin ever created through transactions and a state transition function [1]. Both Hashcash and Bitcoin employ proof-of-work hashing algorithms, but Bitcoin's algorithm is based on SHA-256.To achieve proof-of-work in Bitcoin, a nonce is added to the block until the resulting value meets the required number of zero digits at the beginning of the block hash. Once completed, this cannot be undone without double counting, and any subsequent blocks will have incorrect hashes if manipulated by a malicious attacker [1]. Therefore, the longest chain in the network with a majority consensus rule is followed, which means that an attacker would require significant processing power to override the votes of the most trustworthy nodes and participate in the competition problem if they want to modify a block. In a Merkle tree, transactions within a block are hashed. A Merkle tree is a binary tree structure with numerous leaf nodes and a root hash of all of its offspring nodes. Since any discrepancies in the tree will be mirrored somewhere along the blockchain, merge trees are essential for long-term maintainability. As a result, nodes' blockchain storage systems can use less space. The network only keeps the root hash found in the block header after all transactions in a block have been gathered together and the block has been validated.

In "Bitcoin: A Peer-to-Peer Electronic Cash System," the problem of a receiver not being able to confirm whether a sender has not repeatedly copied the same money is addressed. The payee must demonstrate that most nodes always receive transaction data for the first time whenever a transaction takes place to resolve this issue [5]. The solution that is being suggested comprises a timestamp server that creates a hash of the object block that needs to be timestamped and extensively disseminates it. The existence of the data at the moment the hash value was input must be demonstrated. Each timestamp contains the previous timestamp in its hash, forming a chain that makes subsequent timestamps stronger [5]. This paper aims to establish an allocated timestamp server on a peer-to-peer basis with a proof-of-work system like Adam Back's Hashcash [2]. Proof-of-work involves starting with a nonce to a block and scanning for a value that provides the required zero bits when hashed using SHA-256. The amount of work required is exponentially confirmed by the number of zeros necessary [2].In a timestamp network, proof-of-work is achieved by adding a nonce to a block until a value that satisfies the proof-of-work requirement is found. Once the CPU workload meets the proof of work, the block can't be modified if it doesn't work again since subsequent blocks are chained after it. Any attempt to alter the block will also require redoing all blocks after it [2]. Figure 4 illustrates the changes made to the blockchain before and after implementing the timestamp server.

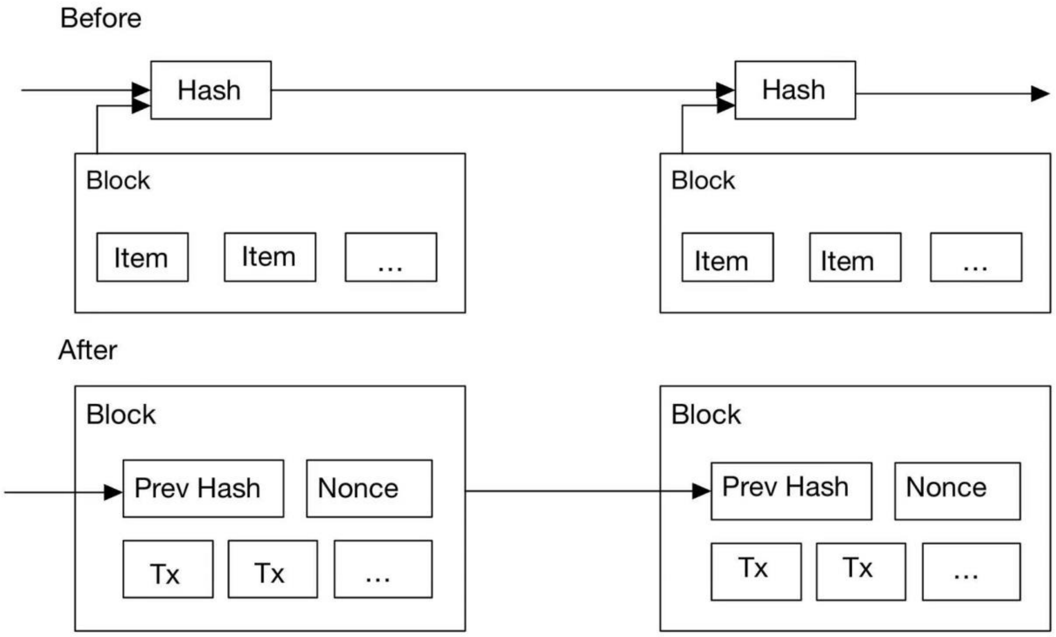


Figure 4. Timestamp the change in the relationship between the stack of the blockchain before and after the server is used [2].

The page additionally addresses Bitcoin transactions, which can be verified without a complete neighborhood node. Until he is confident that he has the longest chain, the character must keep a copy of the block header from the longest proof-of-work chain that was obtained by querying the local node and obtaining the Merkle branch that connects the transaction to the block where its timestamp is located. He cannot independently validate the transaction, but by connecting it elsewhere in the chain, he can see that a community node has recognized it, and the blocks that follow it confirm that the community has extensively disseminated it, as illustrated in Figure 5 [2].

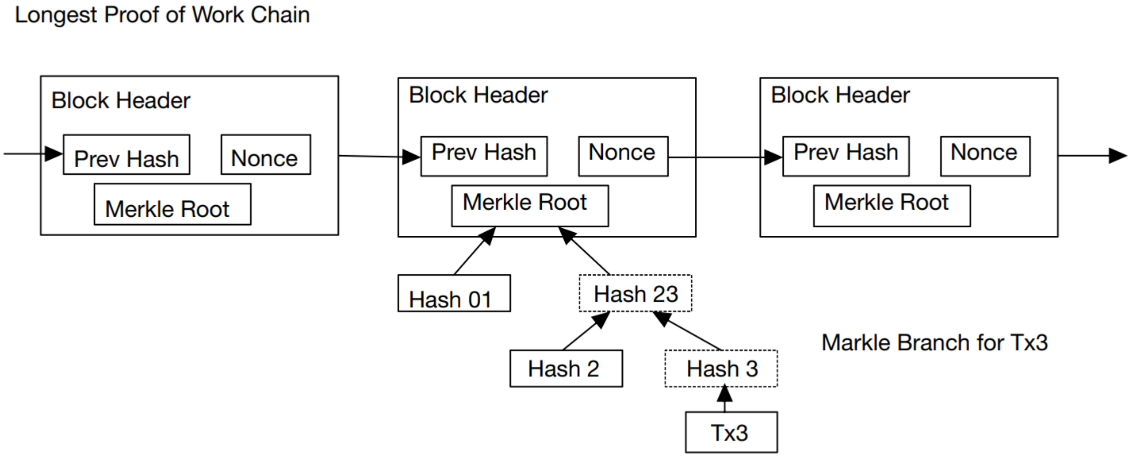


Figure 5. Simplified Payment Verification [2].

Similar content is described in other representative works. To sum up, Bitcoin is a decentralized digital currency and is one of the applications of blockchain technology. Bitcoin uses a simplified payment verification (SPV) and state transition system in transactions to ensure the visibility and security of each transaction. And the method that the payee cannot verify that one of the owners has not spent coins repeatedly is proposed and solved, the payee needs to prove that most nodes received the information for the first time when each transaction occurred. In terms of cryptography, Bitcoin cryptography relies mainly on two basic techniques in cryptography: public-key cryptography for storage and consumption, and cryptographic verification of transactions.

3.2. Ethereum

Vitalik Buterin gives a succinct overview of Ethereum and its transactional design in his well-known essay. With Ethereum, developers will be able to build consensus-based apps with arbitrary functionality as well as incorporate and improve the notions of scripts, cryptocurrencies, and on-chain meta-protocols. These applications combine the benefits that these distinct paradigms' scalability, standardization, feature completeness, ease of development, and interoperability provide. Ethereum is a utility that makes use of the most abstract layer of the blockchain, which may be used in a variety of ways. It can function as a blockchain with a built-in Turing-complete programming language, enabling anybody to develop decentralized apps and smart contracts with their own unique or specific rules for ownership, transaction formats, and state transition functions [4].

In terms of transactions, Ethereum and Bitcoin are very similar, however, there are also significant variations in the following three areas: First, unlike Bitcoin transactions, which can only be made externally, Ethereum messages can be created by external entities or contracts. Second, an explicit option is provided for the inclusion of Ethereum message data. The idea of functions is also included in Ethereum messages, and if the recipient is a contract account, they can decide whether to respond.

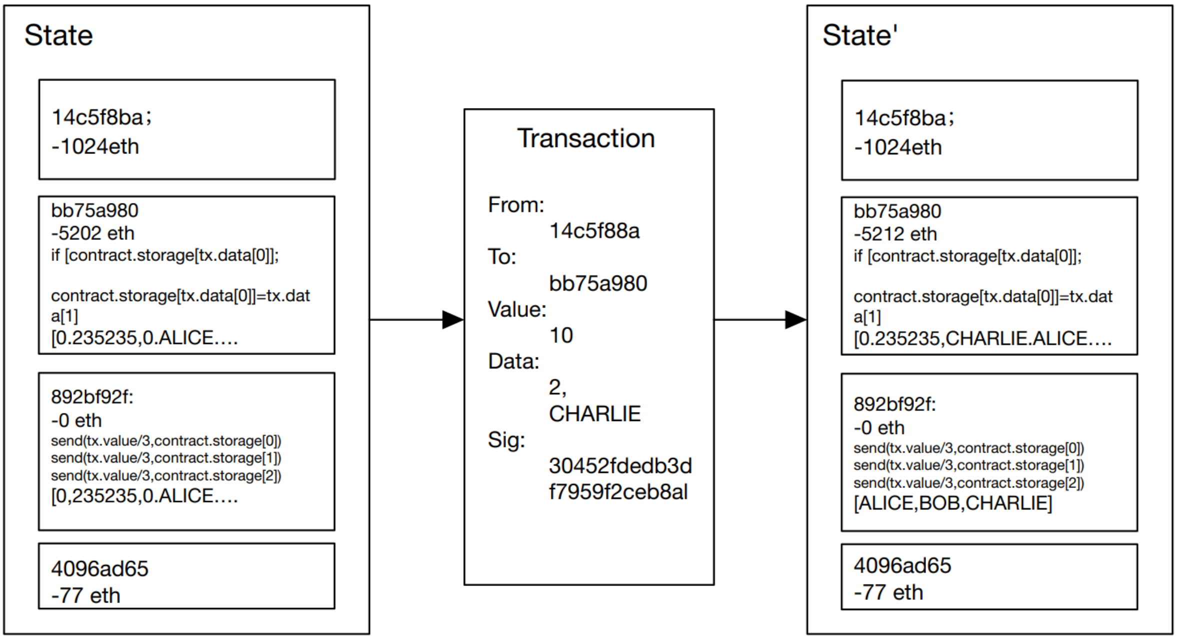

State
14c5f8ba;
-1024eth
bb75a980
-5202 eth
if [contract.storage[tx.data[0]];
contract.storage[tx.data[0]]=tx.dat
a[1]
[0.235235,0.ALICE....
892bf92f:
-0 eth
send(tx.value/3,contract.storage[0])
send(tx.value/3,contract.storage[1])
send(tx.value/3,contract.storage[2])
[0,235235,0.ALICE....
4096ad65
-77 eth
Transaction
From:
14c5f88a
To:
bb75a980
Value:
10
Data:
2,
CHARLIE
Sig:
30452fdedb3d
f7959f2ceb8al
State'
14c5f8ba;
-1024eth
bb75a980
-5212 eth
if [contract.storage[tx.data[0]];
contract.storage[tx.data[0]]=tx.dat
a[1]
[0.235235,CHARLIE.ALICE....
892bf92f:
-0 eth
send(tx.value/3,contract.storage[0])
send(tx.value/3,contract.storage[1])
send(tx.value/3,contract.storage[2])
[ALICE,BOB,CHARLIE]
4096ad65
-77 eth

Figure 6. Ethereum State Transition Function [4].

The time duration transaction in Ethereum is used to factor out a signed packet that retail outlets a message despatched from an externally owned account, the transaction consists of the receiver of the message, the signature that identifies the sender, the volume of Ethernet, and the statistics to be sent, and two values named Startgas and Gasprice [4]. For every transaction, to forestall exponential bloat and countless loops in code, this paper wants to set a restriction on the computation steps that the code can perform, together with preliminary messages and any greater messages derived in the course of execution.

In the piece entitled "Defining the Ethereum Virtual Machine for Interactive Theorem Provers," the Ethereum Virtual Machine (EVM) is described. An external account can start a transaction in the EVM by contacting an existing account or initiating a contract. The whole transition between states of the EVM is known once the transaction is started. The state of contracts produced by external accounts after formation is publicly inspectable, even though this is not covered in full. Contracts and external accounts can both call one another. A balance, gas, and data outflow occur when a third-party account calls an account. When an external account is called, a straightforward balance transfer takes place. The balance transaction is done to the called account, and then the called contract's code is executed, if the called account is a contract. The execution of code has the power to change how executed contracts are stored, read every balance in the account and codes, and do much more. The Ethereum implementation contract is displayed in Figure 7 [7].

In the Ethereum Virtual Machine (EVM), transactions are grouped into blocks which serve as units of protocol between nodes on the Ethereum network. The EVM has specific rules for examining the block number of a transaction and the hash value of previous blocks. While blocks in the network generally form a tree structure, there is only one large branch in terms of the state of the EVM, so this paper assumes that the EVM operates sequentially like a computer.

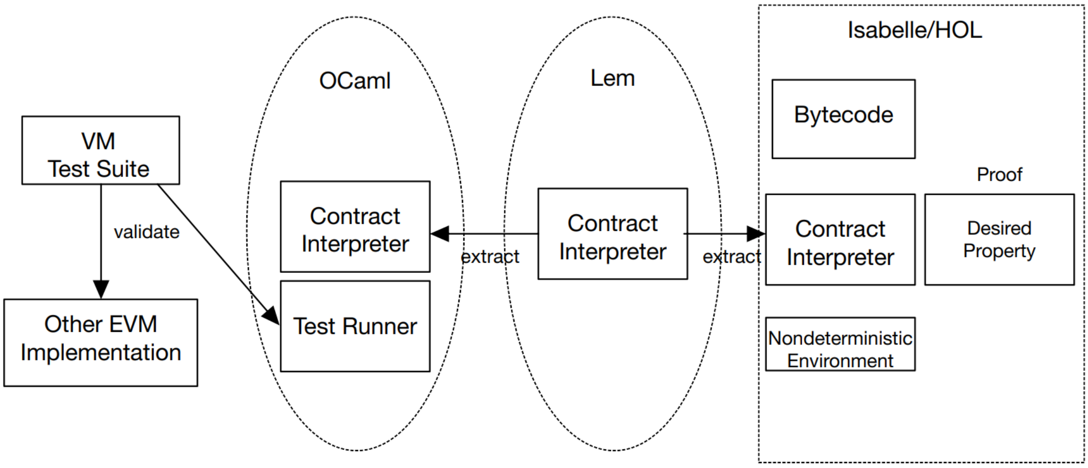
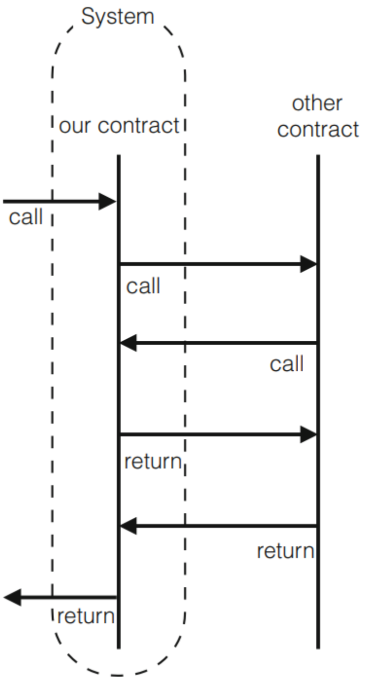


Figure 7. Execution of contract [7].

The environment here refers to anything outside the boundaries of the EVM in addition to every transaction on the EVM other than the verified contract because this article also views a system as a contract. Even tighter than a single account, the entire system symbolizes a single contract call. The contract can invoke the environment, and it also can respond to the calling account. Additionally, the environment has the right to base contract calls on credits outside of its control. Reentrant calls are taken into account when the network refers to our agreement in Figure 8. Reentrant calls are seen as a component of the environment in Figure 9, where the system depicts just one invocation of our contract. Despite the validity of both techniques, this work chooses Figure 9 since it corresponds to the program syntax, which states that CALL commands are followed by further actions in the identical message call rather than the subsequent actions within the reentrant contact [7].



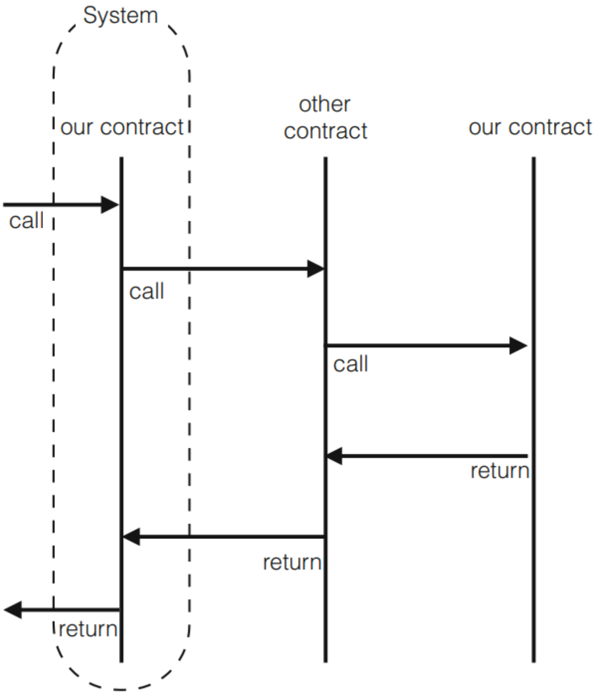


Figure 8. The system is the contract [7]. **Figure 9.** The system is a single invocation [7].

Similar content is described in other representative works. To sum up, Ethereum is an open-source, distributed computing platform based on blockchain technology that can run smart contracts as well as decentralized applications. Ethereum features include programmability, decentralization, openness, security, and scalability. It has its digital currency, Ether, and is the basis for many decentralized applications. At the heart of Ethereum is the Ethereum Virtual Machine (EVM), a stack-based virtual machine that can execute smart contracts. The goal of Ethereum is to build a global decentralized computer to provide developers and users with a wider range of application scenarios and a better service experience.

3.3. Ripple

The Ripple Protocol Consensus Algorithm uses the way that all nodes communicate once every few seconds to reach consensus, at which point the modern ledger is deemed "closed" and becomes the closing closed [10]. This ensures that the network remains correct and consistent. Assuming that the consensus algorithm no longer forks in the neighborhood and is successful, the closing closed ledger maintained via the capacity of all nodes in the neighborhood will be the same. It wants to be tested that all trouble-free nodes agree on the same set of transactions, regardless of their UNLs (Unique Node List), to meet the requirements of the protocol. Because proof of correctness through the way of itself does no longer guarantee protocol consistency, UNLs may additionally moreover be different for each server. For example, a fork may also be applied with no restrictions on the participants of UNL and the measurement of UNL is no longer larger than 0.2^{+}total the place the complete is the number of nodes in the complete network. This can be illustrated with the aid of an easy instance: think about that there are two clusters in the UNL diagram, every higher than 0.2^{+}in total. All nodes are aggregated to structure a cluster, the place the UNL of every node is the identical set of nodes. Because the two factions do now not share any members, every faction can violate the settlement by way of attaining the right consensus independently of every other. Disagreements between factions forestall consensus on the required 80% consensus threshold, so if the connectivity of the two factions exceeds 0.2⇤total, then there is no longer an opportunity for a fork [10].

What the utility can show is its convergence, even although many of the elements are subjective: the consensus procedure will cease in a finite time. The limiting component for algorithm termination is conversation extend between nodes due to the fact the consensus algorithm itself is deterministic and has a preset quantity of rounds t earlier than consensus termination, i.e. The modern set of transactions is declared permitted or disapproved (even if no transaction has extra than 80% of the required settlement at this point, and the consensus is the solely trivial consensus) [10]. The response time of the monitoring node appreciably limits the response time of the node and gets rid of all UNL nodes with a latency increased than the preset bounded b, which ensures that the consensus will terminate at the top sure of tb. But the bounds of correctness and consistency described above should be blissful via the last UNL, deleted after all nodes have been satisfied. If the preliminary UNLs of all nodes meet these conditions, however, some subsequent nodes are eliminated from the community due to latency, and the correctness and consistency ensured will no longer be routinely maintained however need to be comfortable by using a new set of URLs.

Ripple: Overview and Outlook are different from the above, this article not only discusses the protocol consistency algorithm of Ripple but also analysis Ripple under the hood. As previously mentioned, Ripple's consensus protocol is a round-based asynchronous protocol executed by the network's validation server. At the end of each round, all relevant servers publish a newly formed ledger that has been closed after validation by the network.

Transactions are broadcast throughout the network during the collection phase and are subsequently received by the authentication server. The verification server next validates the associated signature and checks the transaction sender's public key from the ledger to confirm the sender's legitimacy. In the candidate set (CS), valid transactions are momentarily saved for later verification. The verification server next examines the relevant XRP transaction history to ensure that the issuing account has enough credit before concluding that the accuracy of transactions stored in CS is accurate. The verification server also determines whether there is a trust path between the sender and receiver for IOU payments. Each validation server gathers verified transactions into a verified proposal, which is then sent out over the network. Table 3 summarizes the common fields present in all Ripple transaction types [12].

Table 3. Common fields contained in all Ripple transaction types [12].

Field	Internal Type	Description
Account	Account	The individual account address that started the transaction.
AccountTxnID	Hash256	(Optional) The hash value identifies another transaction. For the present transaction to be valid, the transaction that preceded it (by Sequencing Numbers) additionally has to be valid and equal the hash. The combination of two transactions simultaneously is made easier by this field.
Fee	Amount	(Required) The number of drops represents the amount of XRP that will be lost as a penalty for distributing this transaction around the network.
Flags	UInt32	(Optional) Bit-flags for this transaction, as a set.
LastLedgerSeq	UInt32	(Optional) A perfect ledger sequence quantity that a transaction can show up in.
Memos	Array	(Optional) Additional statistics were used to perceive this transaction.
Sequence	UInt32	(Required) A transaction must have an identification number that must be precisely one higher than the most recent transaction from the same account that was authenticated for it to be regarded as valid.
SigningPubKey	PubKey	(Required) The public key that corresponds with the private key utilized to sign this transaction is shown in ASCII.
SourceTag	UInt32	(Optional) A random integer is utilized to specify the purpose of this payment.
TransactionType	UInt16	The type of transaction.
TxnSignature	VariableLength	(Required) Transaction signature.

This is done in Ripple by first creating a hash tree of every single transaction that passes verification, then signing the tree's root. When verifying server v receives another proposal from the entire network, it determines if the proposal's issuer is the server listed in its UNL and confirms the accuracy of the transactions it contains [12].

Similar content is described in other representative works. In conclusion, Ripple is a digital currency based on distributed ledger technology. It is positioned as a fast and reliable global payment system. Ripple's blockchain system uses a unique consensus algorithm - the Ripple Protocol consensus algorithm. The algorithm is a trust-based algorithm that reaches consensus by integrating the opinions of individual nodes, rather than solving consensus problems through computation. As a result, Ripple's transaction speed is very fast and can be completed in a matter of seconds. In addition, Ripple's transaction fees are relatively low because it employs a trust-based consensus algorithm rather than a computationally intensive proof-of-work algorithm. Ripple's cryptocurrency, XRP, is a digital asset used to pay transaction fees. When a user transacts using the Ripple network, the transaction fee will be paid in XRP. Ripple also offers a network called RippleNet, a global payment network designed to connect financial institutions around the world. RippleNet enables financial institutions to conduct fast, reliable, low-cost transactions on a global scale.

4. Conclusion

Bitcoin, Ethereum, and Ripple are among the most well-known and valuable digital currencies in the world today. This paper mainly introduced several digital currencies and their main encryption technologies and then carried out a detailed analysis and introduction based on the representative works of these digital currencies. Both Bitcoin and Ethereum are based on blockchain technology, with Bitcoin enabling secure and transparent transactions and storage of value through decentralized, peer-to-peer networking and blockchain technology, while Ethereum is an open-source platform based on blockchain technology that aims to give developers the tools to build decentralized applications. XRP uses a technology called a "consensus ledger," a distributed ledger system that tracks and records every transaction, through both symmetric and asymmetric encryption. Also in the aspect of encryption technology, the main application is blockchain technology, Hash algorithm, and symmetric encryption algorithm.

The future direction of digital currencies is becoming increasingly significant in the global economy. Moving beyond being just speculative assets, digital currencies have evolved into a more stable store of value and an efficient payment system. As such, their appeal continues to grow among investors, businesses, and consumers alike. One potential direction for digital currencies is their widespread adoption as a means of payment and commerce. Major companies like PayPal, Square, and Visa are already integrating digital currencies into their payment systems, and more are expected to follow suit. This opens up new opportunities for digital currencies to become mainstream payment methods, with the added benefit of being faster, cheaper, and more secure than traditional payment systems. However, the future of digital currencies isn't without challenges. Regulatory issues, security concerns, and privacy risks remain major hurdles to overcome. Additionally, the volatility that has plagued many digital currencies in the past still needs to be addressed. Nonetheless, the potential benefits of digital currencies are too great to ignore, and as long as these obstacles are addressed, their growth and adoption will continue to trend upwards.

References

[1] Dejan Vujičić, Dijana Jagodić and Siniša Ranđić 2018 International Symposium INFOTEH-JAHORINA Blockchain Technology, Bitcoin, and Ethereum: A Brief Overview pp 21-23

[2] Satoshi Nakamoto 2008 Bitcoin: Decentralized business review A peer-to-peer electronic cash system

[3] Rainer Böhme, Nicolas Christin, Benjamin Edelman and Tyler Moore 2015 Journal of Economic Perspectives Bitcoin: Economics, Technology, and Governance vol 29 pp 213–238

[4] Vitalik Buterin 2014 white paper A next-generation smart contract and decentralized application platform

[5] Gavin Wood 2014 Ethereum project yellow paper Ethereum: A secure decentralized generalised transaction ledger vol 151 pp 1-32

[6] Sompolinsky, Yonatan,and Aviv Zohar 2015 Financial Cryptography and Data Security: 19th International Conference Secure high-rate transaction processing in bitcoin

[7] Yoichi Hirai 2017 Financial Cryptography and Data Security Defining the Ethereum Virtual Machine for Interactive Theorem Provers pp 520–535

[8] Cachin Christian 2004 Advances in Cryptology-EUROCRYPT 2004: International Conference on the Theory and Applications of Cryptographic Techniques Springer Science & Business Media

[9] Nicola Atzei, Massimo Bartoletti and Tiziana Cimoli 2017 Principles of Security and Trust: 6th International Conference, Held as Part of the European Joint Conferences on Theory and Practice of Software Principles of Security and Trust A Survey of Attacks on Ethereum Smart Contracts pp 164–186

[10] Brad Chase, Ethan MacBrough 2018 arXiv preprint Ripple Research Analysis of the XRP Ledger Consensus Protocol vol 1802.07242

[11] Xiaotie Deng and Fan Chung Graham 2007 Third International Workshop Internet and Network Economics vol 4858

[12] Frederik Armknecht,Ghassan Karame,Avikarsha Mandal,Franck Youssef and Erik Zenner 2015 Trust and Trustworthy Computing: 8th International Conference Ripple: Overview and Outlook pp 163–180

[13] Dylan Yaga, Peter Mell, Nik Roby, and Karen Scarfone 2019 arXiv preprint Cryptography and Security Blockchain Technology Overview

[14] Monika Agrawal and Pradeep Mishra 2012 International Journal on Computer Science and Engineering (IJCSE) A Comparative Survey on Symmetric Key Encryption Techniques

[15] Gurpreet Singh 2013 International Journal of Computer Applications A Study of Encryption Algorithms (RSA, DES, 3DES, and AES) for Information Security vol 67

A content-based collaborative filtering algorithm for movies and TVS recommendation

Ziqi Wang

School of Management and Economics, Beijing Institute of Technology, Beijing, 102488, China

1120201047@bit.edu.cn

Abstract. With the rapid development of multimedia technology and the constant upgrading of film and television libraries, users' demand for movies and television is increasing. How to accurately and timely find favorite movies from massive movie and television resources according to user's preferences and needs has become a great challenge. In recent years, the recommendation of movies and TVs has attracted a lot of research interest from academia and industry. The existing recommendation algorithms mainly include content based and collaborative filtering. The former recommends projects through collaborative learning of others' interests, while the content-based method examines the rich context of the project. In this paper, to further improve the performance of recommendations, a content based collaborative filtering method is proposed to provide recommendations for movies and television. Specifically, we extract and vectorize feature and category information from movies based on TF-IDF and apply truncated SVD to reduce the dimensions of the rating and TF-IDF matrix to retain the most representative information. We calculate the cosine similarity between the vectors from these two matrices. The final recommendation is to list 10 movies based on the average similarity of content and ratings. Extensive experiments on Amazon review data have proven the effectiveness of this method.

Keywords: movie recommendation, content, collaborative filtering

1. Introduction

In recent years, Internet information and film websites have exploded, and film and television resources are unusually rich. However, various movies and television cannot be effectively integrated, which leads to so much information and makes it hard for people to quickly find the movies they like. To this end, how to accurately recommend the desired movie from the massive film and television resources has become a challenge, attracting a large amount of research interest from academia and industry. Accurate movie recommendation not only brings convenience to users, but also brings more profits and traffic to movie websites.

In the current digital era, recommendation plays an import role in our daily life, which aims at predicting the user choices and produce results according to user preference. According to the difference of algorithm designing, the existing recommendation systems are usually divided into recommender based on collaborative filtering, content, and Hybrid methods [1,2,3]. Content-based recommendation manage to list items similar to what users favored in the history as a result [4]. The text of items, like description and category, is then transformed into an unordered bag of words and the examples represented as a vector of words [5]. Then, items will be recommended based on the similarity of contexts and attributes. In common cases, this kind of systems are used when there is abundant attribute information [6]. Therefore, other users play little role in this way. Unlike Content-based approach, Collaborative Filtering relies on the $m \times n$ user-item matrix, which contains m users and n items, to leverages the ratings of other users and calculate the similarities between items. The basic idea behinds it is that similar items receive similar ratings. Some famous systems, like Ringo/Firefly [7] and Recommender [8], are using this technique.

If used in isolation, these two approaches have their own shortcomings. For collaborative filtering, it may encounter problems like the sparsity and cold start. As for the content-based method, it just recommends items similar to what users have rated, leading to less novelty. For years, researchers have been exploring the hybrid technique to eliminate many of the weakness of each approach. Fab designs a partial hybridization approach. In this way, content-based methods are used to classify the peer group, while the ratings are leveraged in the recommendation process [6]. In recent years, more advanced techniques have been developed. For example, a hybrid recommendation approach for articles, introduced by Wang et al. [9], manages to incorporate social tag and friend information in scientific social network. A hybrid scholarly recommendation method, proposed by Sakib,N. et al.[10] integrates metadata in scientific papers. Other methods include hybrid collaborative filtering model integrating deep presentation learning and matrix factorization [11], recommendation algorithm combining user trust network with probability matrix factorization [12], and so on.

In this paper, we try to combine the content-based and collaborative filtering methods to recommend movies and TVs. The remaining section of this paper is organized as follows. The overall architecture of our design is given in Section II. Section III introduces the whole procedure of our approach in details. Finally, evaluation and future works are presented in Section IV.

2. Method

2.1. *General system architecture*

In this study, we try to use content-based together with collaborative filtering by averaging the similarity scores calculated with these two approaches. We manage to make full use of the advantages of content-based filters and reduces the effects of their shortcomings.

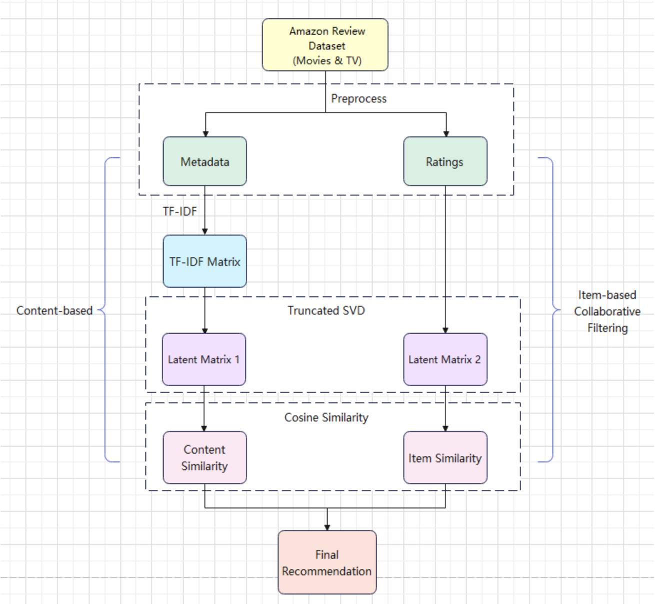


Figure 1. Overall Architecture of System.

Figure 1 depicts the procedure of our proposed method. For the metadata, which is about the detailed information about the movies and TVs, we use text mining approach to transform it into vectors in high dimensional space. For the ratings, we apply Truncated SVD method to it, just like in the content-based phrase, to reduce the dimensionality for efficiency when reserving as much important information as possible. Then, using the two matrices generated respectively, we calculate the similarities between movies. For final recommendation, we combine the similarity results by computing and ranking the average of them and return a list of top-10 similar movies as recommendations.

2.2. Original datasets

In the project, the recommendation system is designed based on the metadata and over 8 million ratings of about 20 thousand Movies and TVs, which are subsets of the complete Amazon review dataset (2018) [1]. The product metadata include 19 attributes like descriptions, category information, price, brand and so on. And ratings are recorded with user ID, product ID and time. Parts of these two datasets are shown in Table 1 and 2.

Table 1. Rating dataset.

	user	item	rating	timestamp
0	A3478QRKQDOPQ2	0001527665	5.0	1362960000
1	A2VHSG6TZHU1OB	0001527665	5.0	1361145600
2	A23EJWOW1TLENE	0001527665	5.0	1358380800

Table 2. Metadata dataset of movies and TVs.

	category	description	title	brand
0	[Movies & TV, Movies]	[Disc 1: Flour Power (Scones;Shortcakes;…)]	My Fair Pastry (Good Eats Vol.9)	Alton Brown
1	[Movies & TV, Movies]	[Barefoot Contessa Volume 2: On these three…]	Barefoot Contessa (with Ina Garten),…	Ina Garten
2	[Movies & TV, Movies]	[Rise and Swine (Good Eats Vol.7) includes…]	Rise and Swine (Good Eats Vol.7)	Alton Brown

2.3. Data preprocessing

Though there are quite a few records in the dataset, many of them are redundant with same information. Therefore, the first step we carried out was to drop the duplicated records. Then, since we are trying to apply the content-based method to the metadata of movies and TVs, we need to clean the columns which contain useful context, but the original structure is inappropriate. In our practice, columns of "category" and "description" are chosen to generate the "Bag of words" for each product. All punctuations are removed, and all terms are transformed into lowercase letters. After that, for convenience, we just reserve the cleaned columns. In our practice, except for the ID, title and bag of words, other columns have been dropped. The final cleaned metadata of movies and TVs are in Table 3.

Table 3. Cleaned metadata of movies and TVs.

item	title	Bag_of_words
0000695009	Understanding Seizures and Epilepsy	movies
0000143529	My Fair Pastry (Good Eats Vol.9)	disc1 flour power scones shortcakes…
0000143592	Rise and Swine (Good Eats Vol.7)	rise and swine good eats vol7 includes…

Similar operations are also carried out on the rating dataset, which contains lots of repeated records. Besides, for time-sequential effects are not taken into consideration in our simple model, we drop the column of “timestamp” as well. Using the cleaned datasets, a larger table can be generated, containing all the information needed in our analysis in later phase. Note that because of the large size of the complete dataset, we just select 20000 records for training. Table 5 shows part of the final table.

Table 4. Final table after merging.

	item	title	Bag_of_words
3911304	B002DLB1IO	Anvil: The Story of Anvil	at 14 toronto school friends steve lips…
1035588	6305476098	The Confession	hired to defend a client who killed to…
3463376	B001AQR3LC	The Tudors: Season 3	henry tudor must overcome his despair…

From the merged table, we can get some rough information about the users and products. Surprisingly, most of the users have ratings that are less than five, while the most active one has commented on over 4000 movies. The distribution of users’ ratings is shown in Table 5.

Table 5. Distribution of users’ ratings.

Quantile	25%	50%	75%	Max
Ratings	1	1	2	4254

Besides, we also manage to find out 20 most active users, and the results are depicted in Figure 2. From the bar chart, we can see that the number of ratings of the top 1 user is almost twice as many as that of the second one. And for the users followed, the figures just decrease steadily.

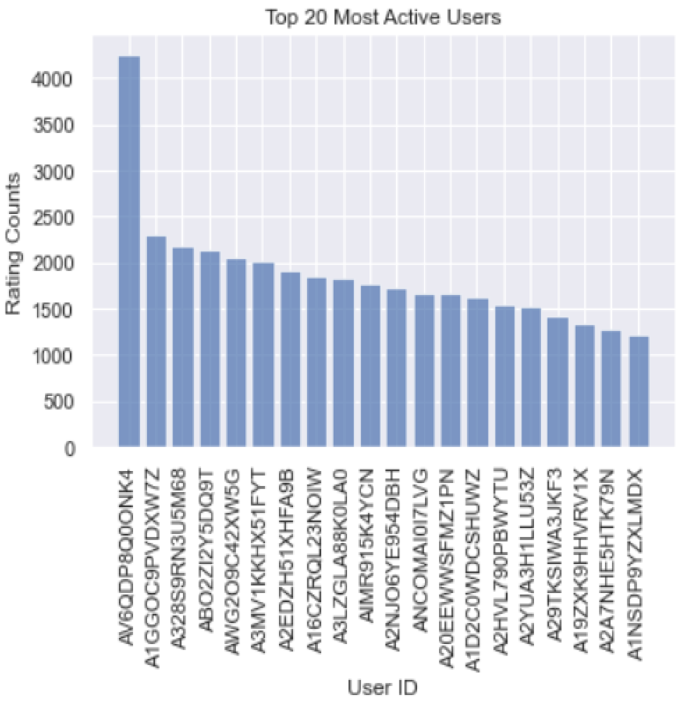


Figure 2. Top 20 Users.

Similarly, we explore the movies data. The results are shown in Table 6 and Figure 3. It seems that though over half of the movies have few ratings, the works Band of Brothers is really popular among the users, with about 50,000 ratings in total.

Table 6. Distribution of movies' ratings.

Quantile	25%	50%	75%	Max
Ratings	2	4	17	24543

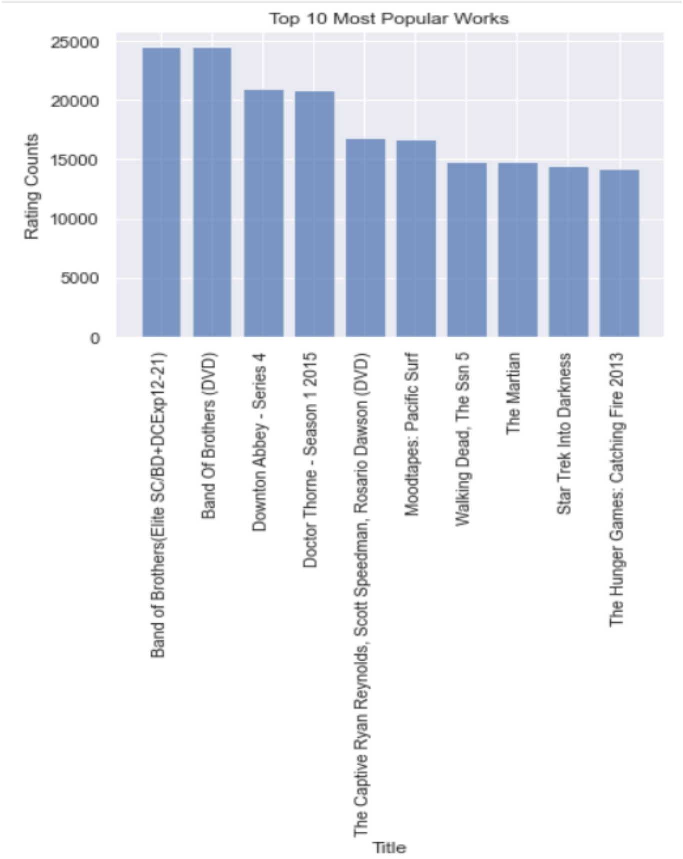


Figure 3. Top 10 Movies.

2.4. Content-based analysis

In our project, metadata is used in this phase. The categories and descriptions are the sources for generating the "bags of words". By removing useless information like punctuation and stop words in data preprocessing, we can get cleaned data prepared for analysis.

2.4.1. TF-IDF vectorization. Nowadays, TF-IDF (Term frequency-inverse document frequency) is one of the most famous term weighting schemes in the field of text mining. It has been used to measure word relatedness [13]. If one specific term appears really frequently in the document set, it will be assumed as a more common term which is less helpful to distinguish one document from the others. However, if it just appears in one document frequently, it is more likely to be regarded as the keyword of the document, and it should be more weighted.

In this approach, for a set of documents which contains m terms in total, a document D is transformed to an m-dimensional vector, and each dimension represents a term. Using TF-IDF, the term weight is calculated as:

$$w_i = tf_i \times \log(\frac{n}{df_i}) \tag{1}$$

Where n documents are in the set, $\mathrm{tf_i}$ represents the times of appearance of term $\mathrm{t_i}$ in document D and $\mathrm{df_i}$ is the number of documents in which term $\mathrm{t_i}$ occurs [14].

Using Tf-idf Vectorizer from Python, we can get a 10610×61703 matrix with rows of movies and columns of terms. Table 7 shows parts of the result. Since the complete matrix is too large to operate calculation on it, for the next step we use the truncated SVD to reduce its dimension.

Table 7. TF-IDF matrix for movies.

	0	1	2	3	4
3911304	0.0	0.040343	0.0	0.0	0.0
1035588	0.0	0.000000	0.0	0.0	0.0
3463376	0.0	0.000000	0.0	0.0	0.0

*2.4.2. Truncated SVD. S*VD (Singular Value Decomposition) is a popular method of dimensionality reduction [4]. It shrinks the space dimension from N to K $(K < N)$. For the $n \times d$ matrix M, SVD decomposes it into three other matrices:

$$M = U\Sigma V^T \tag{2}$$

Where Σ is an $k \times k$ diagonal matrix with nonnegative elements, U and V are $n \times k$ and $d \times k$ matrix respectively, and both of them consist of orthonormal columns. In this way, the value k is the rank of M[15].

2.5. Item-based collaborative filtering

For the ratings data, since the unnecessary column has been removed in the data preprocessing, now we just consider the problem of its large size. After transforming it into pivot table, it is in the format of user-item rating matrix, which is shown in Table 8. As we can see in the table, usually the rating matrix is sparse, containing lots of zeros.

Table 8. User-item Rating Matrix.

user item	**A0019420MGJRFO7TA5QC**	**A0283642BURXFWRSJIJT**
0005019281	0.0	0.0
0005119367	0.0	0.0
0307142493	0.0	0.0

2.6. Final recommendation

In item-based recommendation approaches, cosine similarity is usually used to measure how related two items are. Transformed into a high dimensional space, these two vectors' similarity is calculated based on the angle between them. In the fields of information retrieval and text mining, text documents, which are represented as vectors of terms, are also compared in this way [1]. The formula to calculate cosine similarity is:

$$sim(\vec{a}, \vec{b}) = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}| \times |\vec{b}|} \tag{3}$$

In our process, we calculate the cosine similarity of movies in the two matrices respectively. The content-based matrix shows how movies are similar in context, while collaborative filtering matrix sees them from a different perspective based on ratings they get. And in the final phrase, the similarities computed in two ways will be combined to give the recommendations. As a really primitive model, we just use the average of them for ranking the items.

3. Experiments and performance analysis

3.1. Evaluation metrics

Instead of predicting rating values of the users, our model ranks$t$ items for users and recommend top-k items. The length of the recommended list becomes rather important. If the list is short, the user may miss relevant items and we will lose potential customers. In this case, it is called false-negative. However, if the list is really long, the user may be bored with so many repeated and irrelevant recommendations(false-positive).

Therefore, to evaluate the accuracy of this model, we use the indicators like precision, recall and F1-score. To calculate the indicators mentioned above, first let us think of a recommendation list with t items and denote the set of the recommended items as $S(t)$, and the true set of relevant items as G. Then, the precision will be calculated as follows [6]:

$$Precision(t) = \frac{|S(t) \cap G|}{|S(t)|} \tag{4}$$

And the recall is defined as:

$$Recall(t) = \frac{|S(t) \cap G|}{|G|} \tag{5}$$

To make a trade-off between them, $F_1 - score$ is calculated as:

$$F_1 = \frac{2 \times Precision(t) \times Recall(t)}{Precision(t) + Recall(t)} \tag{6}$$

In our project, $S(t)$ is the list of recommended movies, and G refers to all movies seen by audience of the given input. We randomly select 500 movies to evaluate the performance of top-10 recommendation list, and the evaluation function will return the average F_1 score of them. In our test, we get the result as around 0.94.

3.2. Effectiveness of Truncated SVD

To evaluate the performance of Truncated SVD, we conduct several experiments to see the performance of Truncated SVD on TF-IDF Matrix. Truncated SVD produces the closest rank-k approximation of a given input matrix [16]. Unlike the regular SVD, it can generate a factorization where the number K of columns can be specified (usually $K < rank(M)$). In practice, we use this method to extract the most representative features. Setting the parameter K as 3000, its performance is shown in Figure 4. It can be observed that majority of the original information can be reserved.

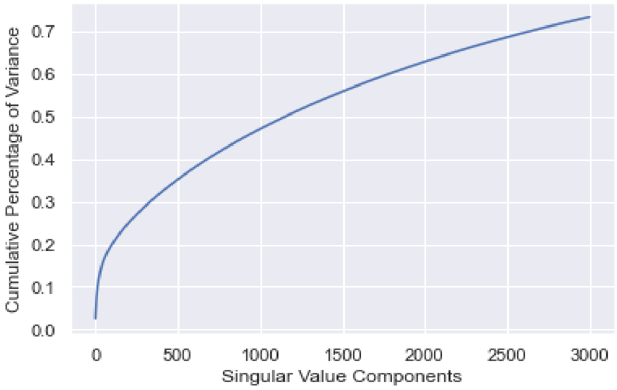


Figure 4. Performance of Truncated SVD on TF-IDF Matrix.

Similarly, Truncated SVD is also used in collaborative filtering analysis, with the parameter set as 3000. Its performance is illustrated in the Figure 5. All the results demonstrate the necessary of introducing the Truncated SVD into our method.

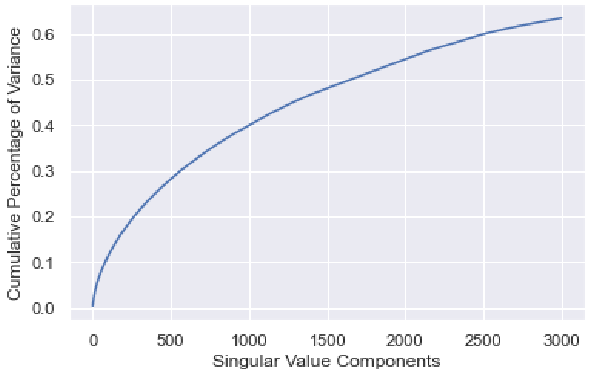


Figure 5. Performance of Truncated SVD on User-Item Matrix.

3.3. Performance analysis

For test, we input the ballet drama "The Flames of Paris" to this model, hoping to find its related works. And the recommendation list provided is in Table 9. Obviously, they are all related with ballet.

Table 9. The Recommended Movies.

Movie	Content Based	Collaborative	Final
Ballet 422	0.513553	3.422993e-07	0.256777
Ballet 201, Beyond the Basics-VHS	0.492126	-1.441504e-04	0.245991
Tchaikovsky-The Nutcracker/Maximova, Vasiliev, Boishoi VHS	0.450763	-1.353671e-04	0.225314
Prima Princessa Presents Swan Lake	0.445547	-1.333689e-03	0.222107
The Red Shoes	0.440672	-3.417913e-04	0.220165
New York City Ballet Workout VHS	0.437733	-1.021616e-03	0.218356
Balanchine Library-Balanchine Essays-Arabesque VHS	0.429910	-8.964244e-04	0.214507
Felia Doubrovska Remembered-From Diaghilev's Ballets Russes to Balanchine's School of American Ballet	0.404366	-1.908196e08	0.202183
The Merry Widow: Martins, McBride, New York City Ballet VHS	0.396656	-3.749068e04	0.198140
Beginner Ballet Barre	0.357443	-9.983903e04	0.178222

4. Conclusion and future work

Recommendation systems are widely used model and we have built movies and TVs recommendation system using content-based and item-based collaborative filtering approaches. As for evaluation, F_1 score is used to exam its accuracy. Though it seems it performs well, we should keep in mind that we just use a very small proportion of the original dataset due to the restrictions of the hardware. Apart from that, the time and space complexity of the program are still problems, especially when it runs on a large-scale dataset. What's more, what we have done is just a primitive experiment, for all the methods are used separately, but not encapsulated in a so-called system. For future scope, the problems mentioned above need solving, and more advanced technique as well as models should be taken into consideration.

References

[1] Jannach D, Zanker M, Felfernig A and Friedrich, G. 2010. Recommender Systems: An Introduction (Cambridge: Cambridge University Press)

[2] Zhang S, Yao L, Sun A and Tay Y. 2019. J. Deep learning-based recommender system: A survey and new perspectives. ACM computing surveys (CSUR), 52(1), 1-38.

[3] Adomavicius G and Tuzhilin A. 2005. J. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE transactions on knowledge and data engineering, 17(6), 734-749.

[4] Lu Z, Dou Z, Lian J, Xie X, and Yang Q. 2015. Proc. Int. Conf. on Artificial Intelligence vol 29 no 1. Content-based collaborative filtering for news topic recommendation.

[5] Mooney R J and Roy L. 2000. Proc. Int. Conf. on 5th ACM Conf. on Digital libraries. Content-based book recommending using learning for text categorization. pp 195-204.

[6] Aggarwal C C. 2016. M. Recommender systems (Vol. 1) (Cham: Springer International Publishing)

[7] Shardanand U and Maes P. 1995. Proc. Int. Conf. On SIGCHI Conf. on Human factors in computing systems. Social information filtering: Algorithms for automating "word of mouth". pp 210-217.

[8] Hill W, Stead L, Rosenstein M and Furnas G. 1995. Proc. Int. Conf. On SIGCHI Conf. on Human factors in computing systems. Recommending and evaluating choices in a virtual community of use. pp 194-201.

[9] Wang G, He X and Ishuga C I. 2018. J. HAR-SI: A novel hybrid article recommendation approach integrating with social information in scientific social network. Knowledge-Based Systems, 148, 85-99.

[10] Sakib N, Ahmad R B, Ahsan M, Based M A, Haruna K, Haider J and Gurusamy S. 2021. J. A hybrid personalized scientific paper recommendation approach integrating public contextual metadata. IEEE Access, 9, 83080-83091.

[11] Dong X, Yu L, Wu Z, Sun Y, Yuan L and Zhang F. 2017. Proc. AAAI Conf. on artificial intelligence vol 31 no 1. A hybrid collaborative filtering model with deep structure for recommender systems.

[12] Yang F R, Zheng Y J and Zhang C. 2018. J. Hybrid recommendation algorithm combined with probability matrix Factorization. Computer Application, vol 38 no 3 pp 644–649.

[13] Yih W T and Qazvinian V. 2012. Proc. Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Measuring word relatedness using heterogeneous vector space models. pp 616-620

[14] Van Meteren R and Van Someren M. 2000. Using content-based filtering for recommendation. Proc. of the machine learning in the new information age: MLnet/ECML2000 workshop vol 30 pp 47-56

[15] Aggarwal C C, Aggarwal L F and Lagerstrom-Fife. 2020. Linear algebra and optimization for machine learning (Vol. 156) (Cham: Springer International Publishing)

[16] Frank M and Buhmann J M. 2011. Proc. IEEE Int. symposium on information theory. Selecting the rank of truncated SVD by Maximum Approximation Capacity. pp 1036-1040

DOI: 10.54254/2755-2721/15/20230813

Research on accuracy analysis and improvement of recommender system based on amazon review

Xuyang Wang

School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou, 730070, China

20213208134@stu.lzjtu.edu.cn

Abstract. Recommender system is a system that uses artificial intelligence and data mining technology to recommend items or services that meet users' preferences based on their historical behaviors and interests. In modern society, people are faced with more and more choices, and the emergence of recommender system can help users filter out valuable content from complex information and improve user satisfaction and experience. In this paper, collaborative filtering is used to implement a recommender system based on Amazon review data set. Meanwhile, Singular Value Decomposition (SVD) and Principal Component Analysis (PCA) are used to conduct dimensionality reduction and other operations on the data. Root Mean Squared Error (RMSE) A value expressed as a recommendation for accuracy. After the establishment of the recommender system and three precision analysis experiments, it achieves this by applying a selected filtering algorithm to the input supplied, which is frequently in the form of user reviews of products.

Keywords: recommender systems, collaborative filtering, singular value decomposition (SVD), principal component analysis (PCA).

1. Introduction

The recommender system is designed to simulate sales staff to inform clients about products and make product recommendations in order to assist users in making purchasing decisions. In today's Internet age, e-commerce, social media, music, movies and other fields are developing rapidly, and a wide variety of various products have emerged. How to recommend information and products that users are interested in from massive data according to the user's interest characteristics and purchase behavior contains huge commercial value. In recent years, the study of recommender systems is particularly interesting to both academics and business.

In recommender systems, there are two basic entities: users and items. Users of recommender systems are required to comment on earlier items. A recommender system's objective is to produce suggestions for brand-new products for these particular users. It accomplishes this by using a chosen filtering algorithm on the input given, typically in the form of user ratings on items [1]. The problem of recommending items from a database has received a great deal of attention, and two main paradigms have emerged. The three primary methods for making suggestions are content-based filtering, collaborative filtering, and hybrid techniques. A movie's genre, director, and star are just a few examples of discrete aspects of what a content-based filtering algorithm can use to generate recommendations.

Collaborative filtering suggestions aim to provide a list of desirable items for engaged users based on the preferences of their like-minded group [2-4]. In content-based recommendation, although in collaborative suggestion, people with similar likes to a given user are found and products they enjoy are recommended, items comparable to items that a specific user has previously liked are recommended. [5]. In order to create hybrid recommendations systems, these two approaches are often used in combination [2-4].

The most popular method of suggestion is collaborative filtering [6-7]. There are several methods for content-based analysis, including spectrum analysis, latent semantic models, matrix factorization, and social recommendation [8–13]. All of these strategies advise products that users who are comparable to the target user for whom the suggestions are being computed have enjoyed (similarly, item-based approach builds on evaluating the similarity of items). Because to their significant role in the winning solution for the Netflix reward competition, the last listed class of algorithms has lately acquired notoriety. [14-16]. Based on the requirements of the above data files and recommender system, collaborative filtering recommendation meets the requirements of the above tasks.The recommender system uses the collaborative filtering algorithm to generate suggestions, which may locate other users who have a high degree of similarity to users based on previous data and recommend their preferred things.

By analyzing users' historical data and interests, the recommender system can improve users' shopping experience, reduce their selection difficulties, and increase their adherence to the platform. The recommender system can simultaneously help retailers boost sales, lower inventory, improve pricing, etc. On the basis of the data files of Amazon sports outdoor reviews from 2014 to 2018, a recommender system will be built and recommend appropriate products to users based on the goods users have previously purchased, rated, and the purchase scores of users who are similar to them. Experimental analysis is carried out to further understand the recommender system.

In this paper, focusing on the above-mentioned aspects, the following section 2 Recommender System basic information will describe the basic information about the recommender system, including data set examples, basic concepts related to the recommender system, the model and algorithm used by the recommender system, as well as specific details of the design and operation of the recommender system. The section 3 Experiment and analysis of accuracy of recommender system is about three experiments to test the recommendation accuracy of the system, with RMSE as the evaluation index. Discuss problems and summarize options ask how the recommender system should improve the accuracy, i.e., reduce the value of RMSE, and list the corresponding methods to reduce the value of RMSE. The section 5 Conclusions is the output of the whole paper and the experimental results.

2. Recommender system basic information

2.1. datasets

The 2014-released Amazon review dataset has been updated using this dataset. Links, reviews (ratings, text, helpfulness votes), product metadata (descriptions, category information, price, brand, and picture properties), and also viewed/also bought graphs are all included. The details of original data set are illustrated in following Table 1 and Table 2. In the data set, the meaning of different attributes are as follows in Table 3.

2.2. Basic principles theoretical concepts

A number of fundamental issues plague recommender systems, reducing the accuracy of forecasts made. Examples of these problems include synonymy, sparsity, and scalability. To deal with this, several alternatives have been put up. It is particularly interested in mathematical procedures that successfully address the aforementioned problems by identifying efficient methodsto make the starting data less dimensional. Singular Value Decomposition (SVD) and Principal Component Analysis (PCA) are two examples of such methods [1].

Table 1. Samples of the review data of the original data set.

overall	verified	reviewTime	reviewerID	asin	reviewerName
5	TRUE	06 3, 2015	A180LQZBUWVOLF	32034	Michelle A
1	TRUE	04 1, 2015	ATMFGKU5SVEYY	32034	Crystal R
5	TRUE	01 13, 2015	A1QE70QBJ8U6ZG	32034	darla Landreth
5	TRUE	12 23, 2014	A22CP6Z73MZTYU	32034	L. Huynh
4	TRUE	12 15, 2014	A22L28G8NRNLLN	32034	McKenna

Table 2. Samples of the reviewText of the original data set.

reviewText	summary	unixReviewTime	style	vote	image
What a spectacular tutu!	Five Stars	1433289600	NaN	NaN	NaN
What the heck? Is this ...	Is this a tutu for nuns?	1427846400	NaN	NaN	NaN
Exactly what we ...	Five Stars	1421107200	NaN	NaN	NaN
I used this skirt for ...	I liked that the elastic ...	1419292800	NaN	NaN	NaN
This is thick ...	This is thick enough ..	1418601600	NaN	NaN	NaN

Table 3. The meaning of different attributes.

Title	Meaning
overall	rating of the product
reviewTime	time of the review (raw)
reviewerID	ID of the reviewer, e.g. A2SUAM1J3GNN3B
asin	ID of the product, e.g., 0000013714
reviewerName	name of the reviewer
reviewText	text of the review
summary	summary of the review
unixReviewTime	time of the review (unix time)
style	a disctionary of the product metadata, e.g., "Format" is "Hardcover"
vote	helpful votes of the review
image	images that users post after they have received the product

2.2.1. Principal component analysis (PCA). Principal component analysis (PCA), a statistical technique, is used to reduce the dimensionality of datasets while maintaining the integrity of the important data. It is a multivariate mathematical method that takes a collection of variables that might be connected and turns them into a new set of uncorrelated variables. The starting variables are combined linearly to form its principal components, which are its constituent parts. Usually, the variables are organized in decreasing order of degree of variation, with the first principal component including the variables with the highest degree of variation and each subsequent principal component reflecting the next highest degree of variation [17]. Data compression, visualization, and feature extraction are just a few of the many uses for PCA in data analysis and machine learning. By reducing the number of variables in a dataset, PCA can speed up processing and improve the efficacy of machine learning algorithms, especially when working with high-dimensional datasets.

2.2.2. Singular value decomposition (SVD). Singular Value Decomposition (SVD) is a common linear algebra technique, which is mainly used to decompose and reduce dimensionality of high-dimensional matrix [18]. It breaks down a matrix into the sum of three other matrices. In some cases, the true meaning behind a data matrix cannot be found, but the essential information in the data matrix is acquired by singular value decomposition. SVD are often used in the fields of data dimension reduction, data compression, matrix approximation, etc. The following are the benefits of using SVD in a collaborative filtering recommendation algorithm: Because SVD can uncover the underlying characteristics behind the data matrix, they enable users to establish a closer relationship with the project, which significantly improves the precision of the recommended outcomes. In this recommender system, SVD are used to decompose the user-item scoring matrix, so as to find the potential user and item characteristics in the low-dimensional space, so as to realize the recommendation.

2.3. *Details of model construction*

2.3.1. Data processing. The input data is pre-processed to calculate the average rating for each unique combination by grouping users, items, and time. After then, only users who had rated at least five things in the datasets and had done so in 2018 or later were allowed to access the data. The processed data is then transformed into Surprise datasets objects, as shown in Table 4.

Table 4. Samples of processed data.

asin	reviewerName	reviewTime	AVGoverall
0000032034	Crystal R	04 1, 2015	3.571429
0000032034	JmeEd	02 7, 2016	3.571429
0000032034	L. Huynh	12 23, 2014	3.571429
0000032034	McKenna	12 15, 2014	3.571429
0000032034	Michelle A	06 3, 2015	3.571429
...	...	...	...
B01HJHHBHG	medinaroger	04 27, 2017	5.000000
B01HJHHBHG	PJT	10 28, 2017	5.000000
B01HJHHBHG	Steve	02 13, 2017	5.000000
B01HJHHBHG	goosedowner	06 11, 2017	5.000000
B01HJHHBHG	old hunter	03 17, 2018	5.000000

2.3.2. Model training. A training set and a test set are created from the data set, and an SVD algorithm with 100 potential factors is trained using the training set. Then, PCA was used to reduce. To speed up the computation of suggestions, the user and item dimensions are multiplied by 50.

2.3.3. Recommendation. Define a "recommend" function that takes the user's name as input and returns the user's top 5 recommended items. It first converts the username into the internal ID used by Surprise library, obtains the user's dimensionality reduction feature vector from the trained SVD algorithm, calculates the similarity score between the user and all items in the dimensionality reduction feature space using dot product, and chooses the best five recommendations based on their score. Finally, use the Surprise library's utility function to convert the final recommended item back to its original item ID.

2.3.4. User input and output. Prompts the user for his or her username and checks to see if the user is in the training set. If the user is in the training set, call the "recommend" function to generate a personalized recommendation and print it out. Otherwise, it prints out an error message. As shown in Table 5, we give some examples of the product recommended.

Table 5. Examples of the product recommended.

asin	userName	
	Andrew M. Silverman	Blake Zimmerman
asin1	7245456313	BO00051ZHS
asin2	B0004TBLW	B0004TBLW
asin3	B000051ZHS	B00004U31L
asin4	B00004NKIQ	B00004T11T
asin5	B00002N6T4	7245456313

3. Experiment and accuracy analysis

3.1. Evaluation metric

The root mean squared error calculates the discrepancy between predicted and actual numbers (RMSE). In the context of machine learning, it is frequently employed, particularly when evaluating neural networks. By calculating the square root of the mean squared errors, RMSE is obtained. The squared error is the difference between the actual value and the anticipated value. For expected values, RMSE is a useful metric of accuracy. The test set's root mean square error (RMSE) is calculated by using the "accuracy" module of Surprise library to assess the trained algorithm's performance. Overall, the recommender system uses the Surprise library to pre-process, train, and evaluate the recommender system, as well as how to generate personalized recommendations based on past user ratings.

3.2. Loss curve in model training

In the experiment, to study the loss of the model in training, The training set's and test set's loss curves are examined., which is shown in Figure 1. This loss curve reveals that there is an overfitting issue with the model because the RMSE values of the training set and test set show no discernible changes. Overfitting is the term for a model's performance when it is good on the training set but poor on the test set. When a model is overfit, it learns the specifics and noise of the training data while neglecting its general characteristics and capacity for generalization. Therefore, on the test set, the model may not properly generalize to the new data.

Specifically, when the loss curves of the training set and the test set are parallel but greatly different, it often denotes an overfitting of the model to the training set. Alternatively, the model overmatches the data in the training set, resulting in a very tiny error on the training set, but when generalized to the test set, the error on the test set is quite significant, meaning that it is considerably different from the error on the training set.

To solve this problem, regularization techniques can be used to limit the model's complexity and avoid overfit. In addition, more data can be added, or data enhancement techniques can be used to broaden the range of the data and enhance the model's generalizability.

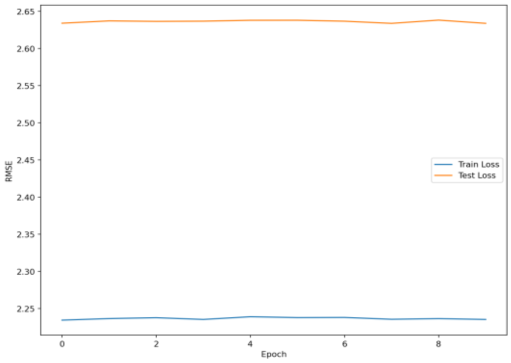


Figure 1. Loss curve of the training and test set.

3.3. Parameters analysis

In this experiment, different combinations of SVD and PCA parameters were selected to run the effectiveness of the recommender system and the recommender system was calculated. RMSE is used as an evaluation index to represent the root mean square error of the recommender system.Table 6 are the results of RMSE changes that change PCA parameters, SVD parameters, etc. In this table, it can be found that for this model, The recommender system's accuracy of recommendations is unaffected by a change in PCA settings. The suggestion accuracy varies clearly when the SVD value is modified. The system's suggestion accuracy can be increased by the decrease of SVD parameters.

Table 6. Change of RMSE with different parameters settings.

SVD	PCA									
	10	20	30	40	50	60	70	80	90	100
10	2.4763	null	null	null	null	null	null	null	null	null
20	2.5121	2.5059	null	null	null	null	null	null	null	null
30	2.5324	2.5337	2.5316	null	null	null	null	null	null	null
40	2.5599	2.553	2.5606	2.5485	null	null	null	null	null	null
50	2.5688	2.5774	2.5759	2.5801	2.577	null	null	null	null	null
60	2.5854	2.5856	2.5902	2.5977	2.5915	2.5953	null	null	null	null
70	2.5989	2.6059	2.5979	2.6069	2.6076	2.6087	2.609	null	null	null
80	2.6163	2.621	2.6106	2.6121	2.6113	2.6127	2.6176	2.6182	null	null
90	2.6206	2.6285	2.6296	2.6149	2.6372	2.6294	2.6284	2.6269	2.6295	null
100	2.6342	2.6409	2.644	2.6415	2.6363	2.6391	2.6455	2.6401	2.6499	2.6451

3.4. Performance for different numbers of recommended products

It is a difficult problem to determine how the quantity of items in the recommender system affects accuracy, because it depends on many factors, such as data set, recommendation algorithm, user characteristics and so on. Generally speaking, the more items are recommended in the recommender system, Users may find it simpler to discover the products they want, but there are certain drawbacks, such as: (1) Too long a recommendation list will increase the cost of user selection, and users need to spend more time and energy to browse the recommendation list. (2) Because it might be challenging for users to locate the products they are truly interested in, extensive lists of recommendations can reduce the effectiveness of recommendations. (3) The accuracy of suggestions may be compromised by excluding certain things that the user may find interesting in a list that is too short. As a result, in real-world applications, it is necessary to consider various factors and choose the appropriate quantity of recommended products for the recommender system.

Table 7. The RMSE of model when the recommended products increasing from 1 to 95.

1	5	10	15	20	25	30	35	40	45
2.6457	2.6456	2.6410	2.6260	2.6458	2.6377	2.6439	2.6317	2.6396	2.6367
50	55	60	65	70	75	80	85	90	95
2.6429	2.6341	2.6384	2.6421	2.6416	2.6441	2.6460	2.6308	2.6403	2.6394

The RMSE value in this recommender system is unaffected whether the experimental findings provided in Table 7 above indicate that the recommended number of elements is between 5, 10 to 95. If the number of suggested products is altered, the RMSE value is not affected, possibly for the following reasons: (1) Sparsity of the data set. If the data set used is very sparse, increasing the number of recommended items may not have a significant impact on the RMSE. This is because even if more recommended items are added, the user is likely to have no rating record, causing the RMSE value to remain the same. (2) The algorithm's performance in making recommendations. Increasing the number of recommended items likely has no appreciable impact on the RMSE because the recommendation system performs well enough. In this case, it may be necessary to use other metrics or evaluation methods to more fully evaluate the performance of the recommendation algorithm. (3) Problems with experimental design. There may be problems with experimental design, such as insufficient changes in the number of recommended items to significantly affect RMSE, or problems with experimental data. The specific cause of the problem can be determined by a more detailed analysis of the experimental design.

Through the analysis of the performance and data set of the recommender system, it can be concluded that the reason why RMSE is not affected when the number of recommended items is changed is Article 2 above: The performance of the recommendation algorithm is probably good enough that increasing the number of recommended items has no significant effect on the RMSE.

4. Discuss problems and summarize options

There are still some problems in this recommender system. For example, the value of RMSE is too large, and the prediction error of the recommender system is relatively large. No cross-validation is used to more precisely validate the recommender system. Here are some ways to lower RMSE:

(1) Use additional data. The model's ability to accurately represent the distribution of scores is enhanced as the size of the data set grows.

(2) Addition of features. The model's accuracy and error may be increased by adding new features, such as user history score, commodity category, time, etc.

(3) Make changes to the model's parameters. To improve the model's performance in the collaborative filtering process, for instance, the hidden vector dimension, regularization parameters, learning rate, and other super parameters can be changed.

(4) Use an integrated approach. Combining multiple models can reduce prediction errors, such as using random forests, gradient lifting trees, etc.

(5) Cross-validation. You may assess your model's performance and choose the ideal set of model parameters and features by using cross-validation approaches.

5. Conclusions

This study first provides a thorough description of a recommender system based on Amazon review data. Three experiments are used to examine and verify the recommender system's accuracy in order to gain a greater knowledge of it: loss curve, changing key parameters and the number of recommended items. This work is an integral part of the design and implementation process of the recommender system. The experimental findings indicate that SVD parameters have an effect on the accuracy of the recommender system, and its operation is satisfactory.

References

[1] Vozalis M.G., Margaritis K.G. “A Recommender System using Principal Component Analysis”, published in 11th panhellenic conference in informatics, 2007, pp. 271-283.

[2] Bobadilla J., et al., “Recommender systems survey”, Knowl. -based Syst 46, 2013, pp. 109-132.

[3] Dietmar Jannach,et al., “Recommender Systems: An Introduction”, Cambridge University Press, USA, 2010.

[4] Feng Zhang, et al., “Fast algorithms to evaluate collaborative filtering recommender systems”, Knowl. -based Syst 96, 2016, pp. 96-103.

[5] Balabanović M. and Shoham Y. “Fab: content-based, collaborative recommendation”, Commun. ACM, vol 40, no 3, pp. 66-72, 1997.

[6] Goldberg, D, et al., “Using collaborative filtering to weave an information tapestry”, Commun. ACM, vol 35, pp. 61-70, 1992.

[7] Schafe J.B., et al., “Collaborative filtering recommender systems”, The Adaptive Web, Springer, 2007, pp. 291-324.

[8] M.J. Pazzani and D. Billsus. “Content-based recommender systems”, The Adaptive Web, Springer, 2007, pp. 325-341.

[9] K. Goldberg, et al., “Eigentaste: A Constant Time Collaborative Filtering Algorithm”, Inf.

Retr., vol 4, no 2, pp. 133-151, 2001.

[10] T. Hofmann. “Latent semantic models for collaborative filtering”, ACM Trans. Inf. Syst., vol 22, pp. 89-115, 2004.

[11] Y. Koren, R. Bell and C. Volinsky. “Matrix factorization techniques for recommender systems”, Computer, vol 42, pp. 30-37, 2009.

[12] U. Shardanand and P. Maes. “Social information filtering: algorithms for automating ‘word of mouth’”, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM Press/Addison-Wesley Publishing Co. pp. 210-217, 1995.

[13] H. Ma, et al., “Recommender systems with social regularization”, Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, ACM, pp. 287-296, 2011.

[14] Y. Koren. “Collaborative filtering with temporal dynamics”, Commun. ACM, vol 53, pp. 89-97, 2010.

[15] J. Bennett and S. Lanning The netflix prize, Proceedings of KDD Cup and Workshop, vol 9, p. 35, 2007.

[16] Yu F., et al., “Network-based recommendation algorithms: A review”, Physica A, vol 452, pp. 192-208, 2016.

[17] Jolliffe T.I. Principal Component Analysis, Springer, New York, 2002.

[18] Ying Zhao, “Collaborative Filtering Algorithm Based on SVD”, Communications World, vol 293, no 10, p. 255, 2016.

A research on quantum digital signatures

Haoxuan Duan

School of Engineering, Computer and Mathematical Sciences, Auckland University of Technology, Auckland, 1010, New Zealand

xtd8436@autuni.ac.nz

Abstract. This paper provides an overview of the basic principles, types, recent works, and applications of quantum digital signatures. The security of traditional digital signature schemes is compromised by the rise of quantum computing, leading to a need for post-quantum cryptography. Quantum digital signatures, which rely on the principles of quantum mechanics, offer a potential solution to this problem. This paper aims to provide a comprehensive overview of quantum digital signatures and post-quantum digital signatures. The paper introduces the basic principles of quantum mechanics, then explains key distribution in quantum digital signatures. The paper then provides a detailed description of both quantum and post-quantum digital signatures, including their differences and applications. Finally, the paper summarizes the main findings in the field, highlights potential future directions, and discusses challenges that humans must address. In addition, the paper examines the widespread applications of quantum digital signatures and post-quantum digital signatures in various fields such as Bitcoin, smart city blockchain, and finance. Finally, the paper summarizes the key findings in the field, highlighting potential future directions and discussing challenges that humans must address. Overall, this paper aims to provide readers with a comprehensive understanding of quantum digital signatures and post-quantum digital signatures and their applications in various domains.

Keywords: digital signatures, post-digital signatures, bitcoin, blockchain.

1. Introduction

In recent years, quantum computing has emerged as a promising technology that could revolutionize the field of cryptography. While quantum computers offer many potential benefits, they also pose a significant threat to traditional cryptographic schemes, which rely on challenging mathematical problems that classical computers can effectively solve. The solution to the mathematical problem led to the development of post-quantum cryptography schemes to resist attacks by classical and quantum computers. This survey aims to understand quantum digital signatures and their security potential comprehensively.

A highly promising approach in post-quantum cryptography is the utilization of quantum digital signatures. Quantum digital signatures use the tenets of quantum mechanics to derive secure digital signatures resilient to the attacks of classical and quantum computers. These signatures are all based on properties of quantum states, such as superposition and entanglement, and provide a way to verify the authenticity and integrity of digital information.

This paper's structure is as follows. The first section outlines the basic principles of quantum mechanics and cryptography, including required distribution, message signing, and verification. Section

2 discusses quantum digital signature schemes, including lattice-based, code-based, multivariate-based, LPN-based, and hash-based schemes. In the third part, this paper discusses the applications of quantum digital signatures in various fields, such as finance, blockchain, and innovative city systems. Finally, this paper summarizes the paper's main findings and discusses future research directions in quantum digital signatures.

2. Basic principles of quantum digital signature

2.1. Basic principles of quantum mechanics

2.1.1. Quantum bits (Qubits). Quantum bit is an abbreviation for a quantum bit, the basic unit of quantum information, which plays a crucial role in quantum computing. They are a quantum analogy of classical bits, but unlike classical bits, they can exist as a superposition of two possible states $|0\rangle$ and$|1\rangle$ [1].

For simplicity, the general rule is$|0\rangle$=$\begin{bmatrix}1\\0\end{bmatrix}$,$|1\rangle$=$\begin{bmatrix}0\\1\end{bmatrix}$.

These states form a standard set of orthogonal bases in quantum computing, commonly called computational bases. An arbitrary quantum state $|\varphi\rangle$ can express as a linear combination of $|0\rangle$ and $|1\rangle$: $|\varphi\rangle$=$\alpha|0\rangle$+$\beta|1\rangle$=$\begin{bmatrix}\alpha\\\beta\end{bmatrix}$…… ①, with α and β being complex coefficients.

A quantum state $|\varphi\rangle$ shaped like a ① is called a quantum bit or qubit. Eq. ① demonstrates the superposition nature of the quantum state, i.e., $|\varphi\rangle$ is at $|0\rangle$ and$|1\rangle$ any superposition state, while a classical bit can only be 0 or 1.

Because $|\alpha|^2 + |\beta|^2 = 1$, The superposition state of qubits can write as $|\varphi\rangle = e^{i\gamma}\left(cos\frac{\theta}{2}\middle|0\rangle + e^{i\varphi}sin\frac{\theta}{2}|1\rangle\right)$. Here, θ, φ, and γ are real numbers. Since $e^{i\gamma}$ has no pronounced effect, this paper can ignore it. So, it can abbreviate as $|\varphi\rangle = e^{i\gamma}\left(cos\frac{\theta}{2}\middle|0\rangle + e^{i\varphi}sin\frac{\theta}{2}|1\rangle\right.$. θ and φ define a point on the unit's three-dimensional sphere. This ball calls a Bloch ball, as shown in Figure 1.

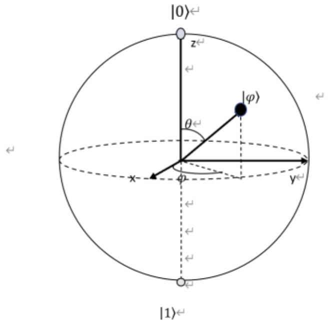


Figure 1. Bloch sphere representation of a Qubit.

2.1.2. Quantum superposition. Quantum superposition is a foundational concept of quantum mechanics, describing the ability of a quantum system to survive in multiple states simultaneously. In classic physics, an object may only be in one place simultaneously, but in the quantum world, particles can be simultaneously in multiple locations.

The concept of quantum superposition has been confirmed experimentally through numerous experiments, such as the double-slit experiment, where a beam of particles is sent through two slits and forms an interference pattern on a screen [2]. The pattern can only be explained by the particles existing in a superposition of states, with a probability distribution determined by the wave function.

In a quantum computer, quantum bits, or qubits, can survive in numerous states simultaneously, allowing for massively parallel processing. However, superposition is a fragile state easily disrupted by external factors, such as noise and environmental interactions. Thus, understanding and controlling superposition is a crucial challenge in developing practical quantum technologies.

2.1.3. Quantum entanglement. Quantum entanglement describes a phenomenon where the states of two or more quantum systems become correlated, even when separated by large distances. This phenomenon has been observed experimentally and now recognize as a fundamental property of quantum mechanics.

Einstein, Podolsky, and Rosen first introduced the concept of quantum entanglement in a 1935 paper. They argued that entanglement violates the principle of local realism, which states that physical processes should be explainable without the need for non-local influences [3]. However, many experiments have experimentally confirmed entanglement, including the famous Bell test, which proved that local hidden variables could not explain entanglement [4].

In quantum cryptography, entanglement is used to distribute cryptographic keys that are secure against eavesdropping. Attempting to intercept the keys will change the entangled state, and the parties will immediately detect the interference. In quantum computing, entanglement performs operations on multiple qubits simultaneously, enabling massively parallel processing that is impossible with classical computers [5].

Despite its potential applications, quantum entanglement remains challenging to understand and control. Its fragility makes it difficult to maintain entangled states in real-world conditions, and decoherence can quickly destroy entanglement. However, ongoing research continues to explore the possibilities and limitations of entanglement in various fields, from quantum computing to quantum metrology. It expects to play a significant role in developing future technologies.

2.2. Quantum digital signatures

2.2.1. Quantum cryptography. (1) Key distribution. Quantum key distribution (QKD) is an encryption protocol that empowers two sides to share a secret key over an unsecured transmission channel in a provably secure manner [6]. QKD, based on the laws of quantum mechanics, provides a means to detect any eavesdropper attempting to intercept or manipulate the transmitted information [6]. The securities of QKD are based on several fundamental tenants of quantum mechanics, for instance, the no-cloning theorem, the Indeterminacy Principle, and the entanglement of quantum states [6].

While practical implementations of QKD can be subject to various attacks and vulnerabilities, research has proposed techniques to mitigate them, such as decoy-state protocols, measurement-device-independent QKD, and trusted-device architectures [6]. QKD has attracted growing interest as a potential technology for secure communication in various applications, such as financial transactions, military communications, and data privacy [7].

QKD's development milestones include the first demonstration by Bennett and Brassard in 1984 [8], the introduction of the BB84 protocol by Bennett and Brassard in 1984 [9], and the first long-distance demonstration over a fiber-optic network by Hughes et al. in 2002 [10]. Numerous experimental demonstrations of QKD have been reported in the scientific literature [11].

Overall, QKD provides a strong foundation for developing future cryptographic technologies to enhance security in various applications.

(2) Using Quantum Mechanics for Message Signing and Verification. a) Eve, as the legitimate party, intercepts and manipulates the quantum gateway between the corresponding parties to perform a man-in-the-middle attack. See Figure 2.

b) During quantum key distribution (QKD), the sender transmits a quantum signal to the receiver through a quantum channel, while classical processing occurs over a classical channel using the shared information. To validate the information, Alice and Bob generate a summary using a hash function. Bob encrypts his summary using a pre-shared or private key and sends the encrypted label to Alice. Alice decrypts the label using the same key and compares it to her summary. If they coincide, authentication is successful. A two-way verification is conducted, with Bob also authenticating Alice's identity, as illustrated in Figure 2.

c) Alice and Bob exchange credentials and nonce, employing the certificate authority's public key to verify each other's public keys. They sign the message digest and non-nonce using their respective private keys to create a signature. They then use each other's confirmed public keys to verify the signatures and confirm that the messages are legally signed. Bit strings are concatenated using ｜｜｜ notation [12].

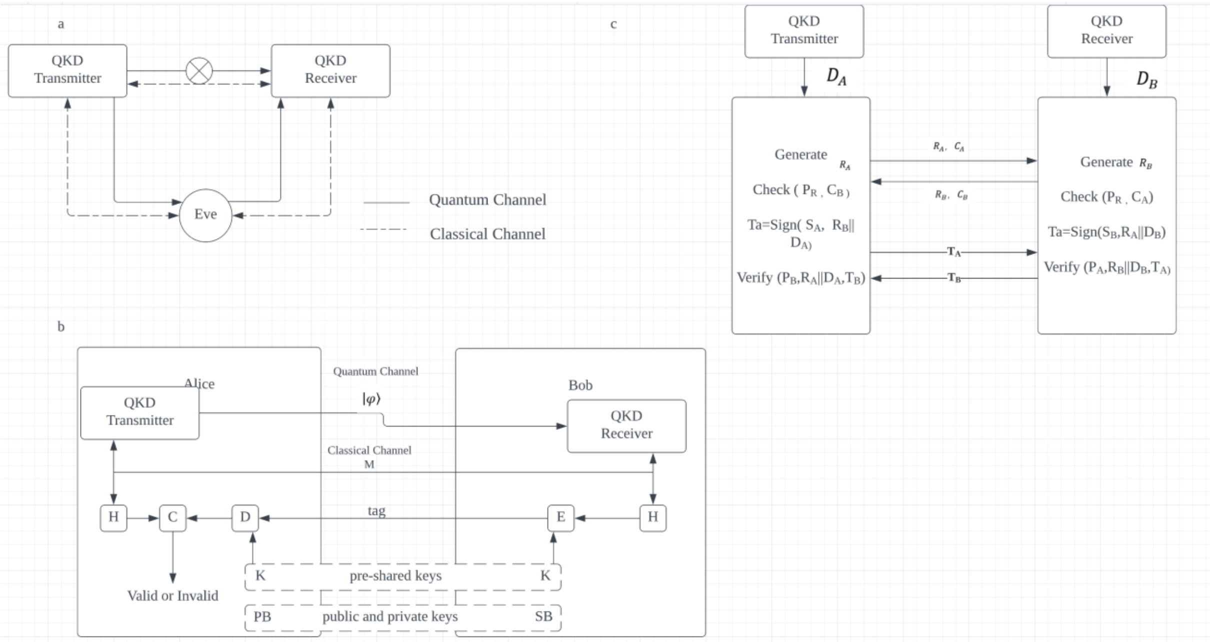


Figure 2. Schematic of a man-in-the-middle attack and flow diagram of post-quantum cryptography authentication. (From Experimental authentication of quantum key distribution with post-quantum cryptography).

2.2.2. Quantum digital signatures. Quantum digital signatures (QDS) offer a secure method for validating the authenticity and integrity of digital data by leveraging quantum mechanics principles, including quantum superposition, quantum entanglement, and the no-cloning theorem [8]. Quantum superposition allows quantum states to exist simultaneously in multiple states, while quantum entanglement links the properties of separate quantum particles, even when they are far apart. It is impossible to make an exact duplicate of an unidentified quantum state, according to the no-cloning theorem, which is a fundamental premise of quantum mechanics. This feature prevents an attacker from forging a quantum signature.

The working principle of QDS involves encoding a message into a quantum state, performing a series of quantum operations to generate a signature, and transmitting the signature along with the message over a classical communication channel. The message recipient can subsequently employ a quantum measurement to confirm the message's authenticity and integrity [7].

One advantage of using quantum signatures is their inherent security, as they rely on the laws of quantum mechanics to prevent tampering or eavesdropping by an attacker. Additionally, quantum signatures can provide a secure method for key distribution, allowing two parties to establish a shared secret key for use in encryption and other cryptographic protocols.

Challenges and limitations of quantum digital signature schemes include maintaining and transmitting quantum states and the need to develop practical and scalable quantum communication technologies. As quantum states are susceptible to their environment, preserving their coherence and preventing decoherence during transmission is a significant challenge.

2.2.3. The post-quantum digital signature. Post-quantum digital signatures are a crucial cryptography component that provides secure communication channels resistant to quantum computer attacks. With the increasing power of quantum computers, traditional signature schemes become vulnerable to attacks, and therefore, post-quantum digital signatures offer a solution for maintaining the security of digital communications. In recent years, significant research has been conducted on post-quantum digital signature schemes, including lattice-based, code-based, multivariable-based, hash-based, and LPN-based schemes. For example, in a paper published in 2021, Tang et al. proposed a new post-quantum digital signature scheme based on binary Goppa codes [13]. The proposed scheme has a minor key size and faster signing and verification than other code-based signature schemes, making it a promising candidate for resource-constrained environments. Such research on post-quantum digital signature schemes is essential for ensuring the security of digital communications in the face of the growing threat of quantum computers.

3. Types of post-quantum digital signature schemes

Table 1. Basic types of quantum digital signatures.

Method	Description	Advantage	Disadvantage
Lattice-based [14-16]	Uses private critical operations on a message to create a digital signature	Minor key size resists classical & quantum attacks, widely used	Vulnerable to channel attacks (e.g., power analysis, timing)
Code-based [17-18]	Encodes message into linear code, adds extra info using the private key	Post-quantum security, relatively simple, well-researched	Limited adoption, expensive, large key, low efficiency
Multivariable-based [19-20]	Applies mathematical operations using a private key	Fast computation speed	Sizeable key size, prone to channel attacks
Hash function–based [21-22]	Hashes message generates a signature using hash and private key	Highly simple and efficient	Slight risk of information leakage
LPN-based [23-24]	Based on the hardness of the LPN problem	Smaller key size compared to other schemes	Vulnerable to quantum attacks

Table 1 compares five post-quantum digital signature schemes: lattice-based, code-based, multivariable-based, hash-based, and LPN-based. Lattice-based schemes are small in size but vulnerable to channel attacks [14][15][16]. Code-based schemes are resistant to quantum attacks but expensive and inefficient [17][18]. Multivariable-based schemes are fast and efficient but have a large key size and are prone to channel attacks [19][20]. Hash-based schemes are simple and efficient but have little information leakage risk due to hash functions [21][22]. LPN-based schemes have a minor key size but are vulnerable to quantum attacks [23][24]. Choosing a post-quantum digital signature scheme depends on specific application requirements, including security level, available resources, and acceptable trade-offs between efficiency and security.

3.1. Lattice-based

In recent years, lattice-based cryptography has gained attention as a potential solution for post-quantum digital signature schemes. A review of lattice-based cryptography and its potential for quantum digital signatures is provided in [14]. This article discusses the advantages and limitations of lattice-based schemes, including their ability to resist classical and quantum attacks but vulnerability to certain types of side-channel attacks. Another article [15] focuses on applying lattice-based cryptography in digital signature schemes, highlighting the potential advantages of smaller key sizes, faster signature generation times, and post-quantum security. However, the authors also acknowledge the limitations and vulnerabilities of these schemes. Finally, article [16] assesses the practical security of lattice-based post-quantum cryptographic schemes against side-channel and fault-injection attacks. This article emphasizes the need for careful implementation and testing to ensure the security of these schemes. Despite the challenges and limitations, lattice-based digital signatures offer a promising approach to achieving secure and efficient digital signature schemes resistant to quantum attacks.

3.2. *Code-based*

The article [17] presents a quantum-resistant digital signature scheme based on the Lyubashevsky framework, which incorporates principles of quantum mechanics to ensure security against quantum computer attacks. The motivation for this work is to develop post-quantum digital signatures resistant to quantum computer attacks, considering the potential threat of quantum computers to traditional digital signature schemes. The proposed scheme uses a code-based construction resistant to classical and quantum attacks, leveraging the underlying principles of quantum mechanics to achieve this level of security.

The survey of code-based digital signatures in [18] provides a comprehensive overview of these post-quantum encryption schemes that resist attacks from classical and quantum computers. While the article does not explicitly emphasize quantum digital signatures, it presents code-based digital signatures as a potential solution to the threat posed by quantum computers. The survey acknowledges some limitations of these schemes, such as their relatively large vital sizes and slow computation times compared to traditional schemes. It also highlights the most recent research and developments in the field of quantum digital signatures, providing readers with the most up-to-date information on state of the art in this area. Overall, the proposed code-based signature scheme from the Lyubashevsky framework in [17] addresses the need for quantum-resistant digital signatures and presents a scheme resistant to classical and quantum attacks. The scheme has several advantages, including small signature and public key sizes, fast signing and verification, and resistance to classical and quantum attacks. However, it also has some limitations, such as the need for a trusted setup and the potential for some side-channel attacks.

3.3. Multivariable-based

The article [19] proposes a new digital signature scheme based on multivariate polynomial cryptography resistant to classical and quantum attacks. The scheme aims to provide post-quantum security in digital signatures, as traditional signature schemes are vulnerable to quantum computers. The article analyzes the proposed scheme's security and performance compared to other post-quantum signature schemes, highlighting its advantages, such as small vital sizes, fast signing, and verification. However, the scheme may be vulnerable to side-channel attacks. The article emphasizes the importance of developing quantum-safe digital signature schemes to prepare for the potential threat of quantum computers to traditional signature schemes.

The article [20] investigates the security of two post-quantum signature schemes, UOV and Rainbow, against fault attacks. The authors present a detailed analysis of the vulnerabilities of these schemes to fault attacks and propose countermeasures to mitigate these attacks. The article discusses the advantages and disadvantages of these schemes compared to other post-quantum signature schemes. However, the article does not explicitly focus on quantum digital signatures but rather on the vulnerability of post-quantum signature schemes to fault attacks. Nonetheless, the article provides important insights into the security of post-quantum signature schemes. It highlights the need for further research to ensure the robustness of these schemes against both classical and quantum attackers.

3.4. Hash function-based

The article [21] reports on an experimental implementation of a secure quantum network that uses digital signatures and encryption. The proposed scheme employs quantum key distribution for encryption and a hash-based digital signature scheme for authentication. The motivation for this work is to provide secure communication channels resistant to quantum computer attacks. The article describes the implementation of the network and presents a detailed analysis of its security and performance. The proposed scheme has several advantages, including resistance to classical and quantum attacks and providing secure communication channels over long distances. The plan does have some drawbacks, though, such as the requirement for specialist tools and the potential for some side-channel attacks. The paper highlights the potential of secure quantum networks in achieving secure communication channels resistant to attacks from quantum computers.

In their paper [22], Li et al. propose a one-time universal hashing quantum digital signature scheme that does not rely on perfect keys. The authors describe their method for constructing one-time universal hash functions using the Gottesman-Chuang stabilizer formalism and show how these can be used to sign messages in a quantum digital signature scheme. The motivation for this work is to develop more secure digital signatures that can resist attacks from quantum computers. The advantages of this approach include its simplicity, efficiency, and security against quantum attacks. However, the scheme has some limitations, such as a higher rate of false positives than traditional digital signatures and the need to manage secret keys carefully. Further research could focus on improving the scheme's performance and addressing these limitations to make it more suitable for practical applications.

The second article [22] proposes a one-time universal hashing quantum digital signature scheme resistant to attacks from quantum computers but also has some limitations, such as a higher rate. In contrast, the first article [21] presents an experimental implementation of a secure quantum network using digital signatures and encryption based on hash functions, which may have some limitations in terms of security and potential side-channel attacks. Therefore, further research and development are needed to improve the security and performance of quantum digital signature schemes based on hash functions.

3.5. LPN-based

Article [23] provides an overview of post-quantum cryptography (PQC), discussing its challenges and prospects for strong and secure hardware design. The authors emphasize the learning parity-with-noise (LPN) problem, which underpins numerous PQC schemes, including digital signature schemes. This paper introduces the LPN problem and its security properties and describes several LPN-based digital signature schemes, including the recent GeMSS scheme. The authors discuss the advantages and limitations of LPN-based digital signature schemes, such as their resilience to quantum attacks and relatively low computational cost. They also point out the need for careful parameter selection to ensure security. This paper provides valuable insights into the challenges and opportunities of PQC- and LPN-based digital signature schemes for robust and secure hardware designs.

The paper [24] proposes a machine-learning framework that tolerates physical noise or errors in hardware. The authors describe their approach to modeling physical noise or errors in hardware and show how it can be incorporated into the training process to improve the accuracy of machine learning models. The advantages of this approach include its ability to improve the robustness of machine learning models in the presence of physical noise or errors and its potential to reduce the need for expensive and time-consuming hardware testing. However, the scheme also has some limitations, such as the need to carefully calibrate the noise or error model and the potential for increased computational complexity. Although the paper by Kamel et al. [24] proposes a machine-learning framework that can tolerate physical noise or hardware errors, it is not directly related to LPN-based quantum digital signatures.

When comparing quantum digital signature schemes, the survey considers key size, efficiency, and security factors. To provide a more detailed comparison, specific security levels or performance benchmarks should be included as criteria, allowing for a clearer understanding of the strengths and weaknesses of each scheme. Additionally, discussing practical implementations and real-world applications of quantum digital signature schemes would offer insight into the potential use cases and practical implications of these schemes in various industries and contexts.

Future research directions in quantum digital signatures include addressing open problems and challenges related to efficiency, key size, and resistance to side-channel attacks. Furthermore, the research could focus on developing new cryptographic primitives based on quantum mechanics principles to enhance the security of digital signature schemer further.

4. Application of quantum digital signature

Table 2. Applications of quantum digital signatures.

Application	Description
Bitcoin [25-26]	Using quantum digital signatures to secure Bitcoin.
Smart city blockchain [27]	Everyone can protect the blockchain, and Everyone can still mine it.
Finance [28]	In finance, only security analysis and proposals must be widely used after people agree.

Table 2 summarizes several potential applications of quantum digital signatures. In the case of Bitcoin, researchers have compared classical and post-quantum digital signature algorithms to protect Bitcoin transactions. In the context of intelligent city blockchains, quantum digital signatures can help protect the blockchain while still allowing mining. Finally, quantum digital signatures can be used in the financial sector for security analysis and proposals after people have consented. These applications demonstrate the potential versatility of quantum digital signatures in various fields and highlight their importance for protecting sensitive information.

4.1. Bitcoin

Noel et al. conducted a comparison between classical and post-quantum digital signature algorithms employed in Bitcoin transactions [25]. The authors evaluate the performance of several post-quantum algorithms based on the hash function, including the XMSS, SPHINCS+, and WOTS+ schemes, and compare them to the widely used ECDSA algorithm. They show that post-quantum algorithms provide better security against quantum attacks but with increased computational complexity and larger signature sizes. León-Chávez et al. propose a hash-based digital signature scheme resistant to quantum attacks and can be implemented on current Bitcoin hardware [26]. The authors assess their scheme's performance by examining the signature size and verification time, and they compare these results with other post-quantum digital signature schemes. The proposed scheme is compatible with existing Bitcoin infrastructure and has a low computational cost. However, the scheme also has some limitations, such as its larger signature size compared to some classical schemes and the need for careful parameter selection. Both papers highlight the need for post-quantum algorithms to ensure the long-term security of the Bitcoin blockchain. These studies provide valuable insights into applying post-quantum digital signatures based on hash functions in protecting Bitcoin transactions.

4.2. Smart city blockchain

The article by Chen et al. presents a post-quantum blockchain construction for innovative city applications using quantum digital signatures [27]. The motivation behind this work is to address the security challenges of existing blockchain systems in the era of quantum computing. The authors propose a post-quantum blockchain framework that employs quantum digital signatures based on hash functions to ensure the security and privacy of innovative city applications. They evaluate the performance of their proposed framework using various innovative city scenarios and show that it outperforms existing blockchain solutions in terms of security and efficiency. The advantages of using quantum digital signatures include their resistance to quantum attacks, enhanced security and privacy, and the ability to support new cryptographic primitives. However, the authors also acknowledge some limitations, such as the need for specialized hardware and software to implement quantum digital signatures and the potential impact of future developments in quantum computing. Overall, this article provides valuable insights into the potential applications of quantum digital signatures in blockchain systems for innovative city applications.

4.3. Finance

The essay by J. Hayes [28] explains how quantum computing might be used in the financial sector. This work is motivated by the need for faster and more secure financial transactions and the limitations of classical computing in meeting these challenges. The authors emphasize the potential of quantum computing in fields like portfolio optimization, risk assessment, fraud detection, and cryptography. They discuss quantum computing methods and algorithms, such as Shor's algorithm, Grover's algorithm, and quantum annealing, which can be applied in the financial sector. The advantages of quantum computing in finance include faster and more efficient computing, better risk management, and improved security through quantum digital signatures. However, there are challenges and limitations to adopting quantum computing in finance, such as the need for dedicated hardware, the high cost of quantum computing, and the potential security risks associated with quantum cryptography. Overall, this research offers insightful information about the possible uses and restrictions of quantum computing in the financial services industry.

5. Conclusion

This investigative paper provides an overview of quantum digital signatures, a promising approach to secure and real-world digital communication in the post-quantum era. This article first describes the importance of digital signatures in modern communications and the threat of quantum computers. This article then explores the fundamental principles of quantum mechanics and quantum cryptography and how they can be used for message signing and verification. This paper then discusses the types of quantum digital signature schemes, including lattice-based, code-based, multivarious-based, and hash-based LPN-based. The advantages and limitations of each scenario are discussed. The challenges and opportunities of future research in quantum digital signatures are summarized. Future research directions in quantum digital signatures should address the open problems and challenges related to efficiency, key size, and resistance to side-channel attacks. Developing new cryptographic primitives based on quantum mechanics principles could further enhance the security of digital signature schemes. Additionally, exploring practical implementations and real-world applications of quantum digital signature schemes will offer valuable insights into their potential use cases and implications in various industries and contexts. Researchers should also investigate the integration of quantum digital signatures with other emerging technologies, such as blockchain, to leverage their potential for secure and efficient communication and transactions in the quantum era.

References

[1] Mullamuri B 2021 *ProQuest Dissertations* Publishing Enabling Quantum Cryptography Using Quantum Computer Programming p 28864879.

[2] Bouwmeester D and Zeilinger A 2000 *The Physics of Quantum Information* The Physics of Quantum Information: Basic Concepts Berlin Heidelberg.

[3] A. EinsteinPodolsky and N. RosenB 1935 Can Quantum-Mechanical Description of Physical Reality Be Complete?

[4] Bell JS 1964 Physics On the Einstein-Podolsky-Rosen paradox vol 1 pp 195-200.

[5] Chuang DG 2001 Quantum digital signatures.

[6] J. Mullins 2001 *IEEE Spectrum* The topsy turvy world of quantum computing.

[7] 2018 *Springer Science and Business Media LLC* Applied Cryptography and Network Security

[8] D. Gottesman and I. Chuang, 2001 *arXiv preprint* Quantum digital signatures.

[9] Pramode K. Verma, Mayssaa El Rifai and Kam Wai Clifford Chan 2019 *Springer Science and Business Media LLC* Multi-photon Quantum Secure Communication.

[10] Delpech De Saint Guilhem and Cyprien P. R. 2021 *University of Bristol (United Kingdom) ProQuest Dissertations* Publishing On the Theory and Design of Post-Quantum Authenticated Key-Exchange, Encryption, and Signatures.

[11] Marius Nagy and Selim G. Akl 2006 *International Journal of Parallel Emergent and Distributed Systems* Quantum computation and quantum information.

[12] Liu-Jun WangZhang, Jia-Yong Wang, Jie Cheng, Yong-Hua Yang, Shi-Biao Tang, Di Yan, Yan-Lin Tang, Zhen Liu, Yu Yu, Qiang Zhang and Jian-Wei PanKai-Yi 2021 *npj Quantum Information* Experimental authentication of quantum key distribution with post-quantum cryptography vol 7.

[13] TangLi X, Hu X, Wang R and Zeng XY 2021 *IEEE Access* A New Post-Quantum Digital Signature Scheme Based on Binary Goppa Codes pp 164530-164543.

[14] Yu Y 2021 *National Science* Review Preface to special topic on lattice-based cryptography vol 8.

[15] Lyubashevsky V 2021 *National Science Review* Lattice-based digital signatures vol 8.

[16] Ravi, Chattopadhyay, A, D'Anvers, J. P and Baksi A 2022 Side-channel and Fault-injection attacks over Lattice-based Post-quantum Schemes (Kyber, Dilithium): Survey and New Results.

[17] Song Y, Huang X, Mu Y, Wu W and Wang H 2020 *Theoretical Computer Science* A code-based signature scheme from the Lyubashevsky framework vol 835 p 15-30.

[18] SONG Y. 2021 *Chinese Journal of Network and Information Security* Survey of code-based digital signatures vol 7 pp 1-17.

[19] Kuang R, Perepechaenko M and Barbeau M 2022 *Scientific Reports* A new quantum-safe multivariate polynomial public key digital signature algorithm.

[20] Krämer J and Loiero M 2019 *Lecture Notes in Computer Science book series (LNSC)* Fault attacks on UOV and Rainbow vol 11421 pp 193-214.

[21] Yin H, Fu Y, Li C, Weng C, Li B, Gu J, Lu Y, Huang S and Chen Z 2022 *National Science Review* Experimental quantum secure network with digital signatures and encryption.

[22] Li B, Xie Y, Cao X, Li C, Fu Y, Yin H and Chen Z 2023 *Quantum Physics (quant-ph) Cryptography and Security* One-Time Universal Hashing Quantum Digital Signatures without Perfect Keys.

[23] Bellizia D, El Mrabet N, Fournaris A. P, Pontié S, Regazzoni F, & Standaert F. X, Tasso É and Valea E 2021 *IEEE International Symposium on Hardware* Oriented Security and Trust Challenges and Opportunities for Robust and Secure HW Design.

[24] Kamel D, Standaert F, Duc A, Flandre D and Berti F 2020 *IEEE Transactions on Dependable and Secure Computing* Learning with physical noise or error vol 17 pp 957-971.

[25] Noel MD, Waziri OV, Abdulhamid MS, Ojeniyi AJ and Okoro MU 2020 *IEEE* Comparative Analysis of Classical and Post-quantum Digital Signature Algorithms used in Bitcoin Transactions.

[26] León-Chávez M Á, Perin LP and Rodríguez-Henríquez F 2022 *Springer* Post-Quantum Digital Signatures for Bitcoin Principles and Practice of Blockchains pp 251-270.

[27] Chen J, Gan W, Hu M and Chen C M 2021 *Journal of Information Security and Applications* On constructing a post-quantum blockchain for a smart city vol 102780.

[28] Hayes J 2019 *Engineering & Technology Quantum* on the money: Quantum computing in financial services sector vol 14 pp 34–37.

Modeling and numerical simulation optimization of gain spectrum of thulium-doped broadband fiber amplifier based on cat swarm algorithm

Rui Guo[1,†], **Zhuoer Liu**[2,4,†] **and Yuchen Quan**[3,†]

[1] School of Electronic Engineering, Jiangsu Ocean University, Lianyungang, Jiangsu, 222000, China
[2] School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing, 100876, China
[3] International College, Zhengzhou University, Zhengzhou, Henan, 450001, China

[4]lze0517@bupt.edu.cn
[†]These authors contributed equally

Abstract. With the development of light technology, especially the maturation of WDM/DWDM technology, the demand for optical amplification of the S-band and S+ band (1450nm~1520nm) is increasing day by day, and the energy level structure of Tm^{3+} has energy level transitions to meet the requirements of S-band and S+ band amplification. Although thulium ion has a very complex energy level structure, TDFA is one of the most promising optical fiber amplifiers for S and S+ bands. At the same time, with the continuous development of computer technology and mathematical theory, the optimization algorithm has been rapidly developed and widely used in recent decades, based on genetic algorithm, simulated annealing algorithm, and other traditional optimization algorithms that have been proved to get good convergence speed and optimization results. In this paper, the thulium-doped fiber amplifier's gain is optimized by using a cat swarm intelligent optimization algorithm to obtain the maximum fiber length and doping concentration.

Keywords: Fiber Optic Communication, Thulium-doped Fiber Amplifier, Cat Colony Optimization Algorithm, Amplification Gain.

1. Introduction

With the continuous development of information and communication technology, people's requirements for the network are getting higher and higher, which to a certain extent also promotes the development of fiber optic communication technology. At the same time, this also puts forward many new requirements for the optical amplification technology in the communication window band, for example, the effective optical amplification in the S-band (1450nm~1520nm) is one of the important optimization targets [1]. However, the erbium-doped fiber amplifier (EDFA), which has been used extensively in fiber optic communication systems, cannot effectively amplify the S-band signal. On the contrary, there are jumps in the energy level structure of thulium ion to meet the S-band amplification, which can realize the effective amplification of the S-band optical signal to meet the communication demand. Therefore, the thulium-doped fiber amplifier has become a new research hotspot for fiber optic communication devices and is the most promising laser amplifier device in S-band. Most of the studies on the gain of thulium-doped fiber amplifier (TDFA) start from the steady-state conditions of thulium ion particle rate equation and make reasonable approximations and modeling to derive the analytical expression of TDFA, to derive the effect of three parameters on the gain by the traditional calculation method.

With the continuous development of computer technology and mathematical theory, optimization algorithms have been rapidly developed and widely used in recent decades, providing powerful tools and methods for solving practical problems. In 2006, Shu-Chuan Chu et al. proposed the cat colony algorithm [2], a heuristic optimization algorithm based on the behavior of natural cat colonies, which finds the optimal solution by simulating the population intelligence of a cat colony. The algorithm can solve the function problem with a set number of iterations with a fast convergence rate to the maximum value. Therefore, it is feasible and innovative to use the cat swarm optimization algorithm as a tool to solve the problem of gain optimization of a thulium-doped fiber amplifier. This paper combines the optimization algorithm with the thulium-doped fiber amplifier's gain equation from the principle of the cat swarm intelligent optimization algorithm to derive the optimization results of fiber length and doping concentration at maximum gain.

2. Gain simulation and optimization

2.1. Research background

With the development of light technology, especially the maturation of WDM/DWDM technology, the demand for optical amplification of the S-band and S+ band (1450nm~1520nm) is increasing day by day, and the energy level structure of Tm^{3+} has energy level transitions to meet the requirements of S-band and S+ band amplification [3]. Although thulium ion has a very complex energy level structure, TDFA is one of the most promising optical fiber amplifiers for S and S+ bands [4]. At the same time, with the continuous development of computer technology and mathematical theory, the optimization algorithm has been rapidly developed and widely used in recent decades, based on genetic algorithm, simulated annealing algorithm, and other traditional optimization algorithms that have been proved to get good convergence speed and optimization results. At present, there are still many problems in Thulium-doped fiber amplifiers that need further study. Therefore, it is necessary to combine the optimization algorithm to model and optimize the thulium-doped fiber amplifier's gain.

2.2. Method

2.2.1. Energy-level modeling. This topic is to optimize the thulium-doped fiber amplifier's gain from 1450 to 1520nm. By consulting the relevant literature, the $^3H_4 \rightarrow {}^3F_4$ emission band of Tm^{3+}-doped ZBLAN is obtained as shown in figure 1 below, in which the pump wavelength is 790nm, so the three-level system model is abstracted according to the ion-doped level transition diagram (figure 2), where $E_3 \rightarrow E_2$ is the radiation transition (emitting photon) and $E_2 \rightarrow E_1$ is the radiation-free transition (heat generation) (as shown in figure 3).

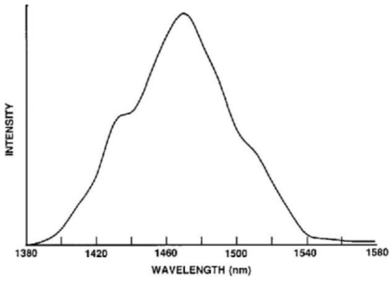


Figure 1. $^3H_4 \rightarrow {}^3F_4$ emission band for Tm^{3+}-doped ZBLAN [5].

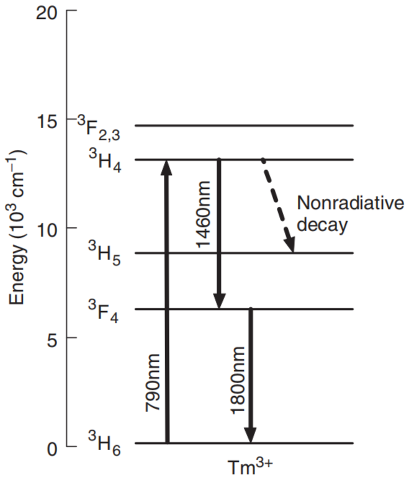


Figure 2. Energy level diagram of Tm^{3+} ion [6].

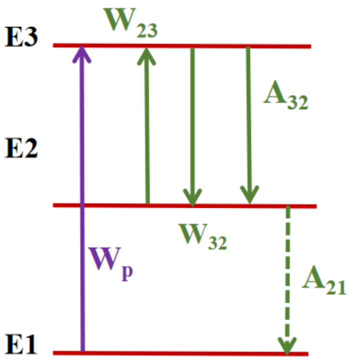


Figure 3. Energy level model.

According to the characteristics of the three-level system, the corresponding motion rate equation system(in equations (1)(2)(3)) and the power equation(in equations (4)(5)(6)) are constructed.

$$\frac{\partial N_1(z)}{\partial t} = -W_p(z)N_1(z) + A_{21}N_2(z) \tag{1}$$

$$\frac{\partial N_2(z)}{\partial t} = -[W_{23}(z) + A_{21}]N_2(z) + [W_{32}(z) + A_{32}]N_3(z) \tag{2}$$

$$\frac{\partial N_3(z)}{\partial t} = W_p(z)N_1(z) + W_{23}(z)N_2(z) - [W_{32}(z) + A_{32}]N_3(z) \tag{3}$$

Where $W_p(z) = \frac{\sigma_{13}P_p(z)}{hv_{13}A_{eff}}, W_{23}(z) = \frac{\sigma_{23}P_s(z)}{hv_{23}A_{eff}}$, $W_{32}(z) = \frac{\sigma_{23}P_s(z)}{hv_{23}A_{eff}}$ are the upward transition rate after absorption of the pump photon, the transition rate after spontaneous forward transmission and the stimulated radiation rate respectively. $A_{eff} = \pi r^2$is the fiber core cross-sectional area.

$$\frac{dP_p(z)}{dz} = \Gamma_p(-\sigma_p N_1(z) - \alpha_a)P_p(z) \tag{4}$$

$$\frac{dP_s(z)}{dz} = \Gamma_s[\sigma_{32}N_3(z) - \sigma_{23}N_2(z) - \alpha_s]P_s(z) \tag{5}$$

$$\frac{dP_{ase}(z)}{dz} = \Gamma_{ase}[\sigma_{32}N_3(z) - \sigma_{23}N_2(z) - \alpha_s]P_s(z) + \sigma_{21}N_2(z)hv\Delta v \tag{6}$$

2.2.2. Curve fitting. To find the maximum value of the emission interface in the studied band, fitting the emission interface as a function of wavelength is a key step. Figure 4 shows the fluorescence spectrum of thulium-doped ion glass. To find the emission spectrum as a function of wavelength, one of the curves was selected for preprocessing. The MATLAB curve fitter tool was used to fit the functions in the waveband range from 1450 to 1520nm. A multinomial Fourier expansion was chosen to obtain a functional relationship with high fitting accuracy. $y1 = a_0 + a_1\cos\omega x + b_1\sin\omega x + \ldots + a_6\cos 6\omega x + b_6\sin 6\omega x$ with constant parameters a_0, b_0, a_1, b_1, a_2, b_2, a_3, b_3, a_4, b_4, a_5, b_5, a_6, b_6, is chosen as the fitting function for subsequent amplification gain model simulation.

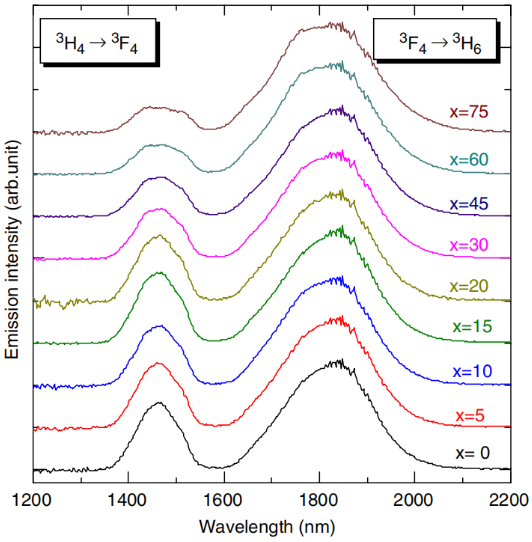


Figure 4. Fluorescence spectra of glasses [6].

Similarly, to obtain the absorption cross-section maximum, it is also necessary to fit the absorption cross-section versus the signal wavelength curve as a function. Figure 5 shows the absorption spectra of thulium-doped glasses. Between the ground state level of $^3H_6$ and the levels of $^1G_4$, $^3F_{2,3}$, $^5H_4$, $^3H_5$, and $^3F_4$, there are five bands of absorption. The fit function is still selected for the studied waveband (1450~1520nm) which uses a similar curve fitting method as the emission interface.

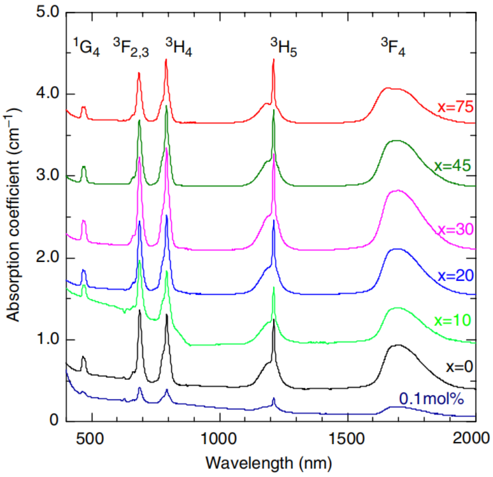


Figure 5. Absorption spectra of glasses [6].

2.2.3. Fiber amplifier gain simulation. The rate and power equations derived from the above modeling of the system energy levels of thulium-doped fiber amplifiers in the 1450~1520nm signal band can be used to simulate the gain curves of fiber amplifiers under the influence of different variables (wavelength, doping concentration, and pumping power).

$$\mathrm{G}(\mathrm{P_p}(z),\mathrm{P_s}(z),\mathrm{N_2},\mathrm{N_1},z) = 10\log_{10}(\frac{\mathrm{P_s}(z)}{\mathrm{P_s}(0)})(\mathrm{dB}) \tag{7}$$

The gain equation is obtained from the three-energy level power propagation equation as in equation (7). Write the code using MATLAB as the simulation environment. After parameter localization,

$$\mathrm{f}(1) = \mathrm{P}(1)\mathrm{T_s}(\sigma_{\mathrm{se}}\mathrm{N_3} - \sigma_{\mathrm{sa}}\mathrm{N_2}) - \alpha\mathrm{P}(1) \tag{8}$$

$$\mathrm{f}(2) = \mathrm{P}(2)\mathrm{T_p}(\sigma_{\mathrm{13pe}}\mathrm{N_3} - \sigma_{\mathrm{13pa}}\mathrm{N_1}) - \alpha\mathrm{P}(2) \tag{9}$$

$$\mathrm{f}(3) = \mathrm{P}(3)\mathrm{T_{ase}}(\sigma_{\mathrm{se}}\mathrm{N_3} - \sigma_{\mathrm{sa}}\mathrm{N_2}) + 2\sigma_{\mathrm{se}}\mathrm{N_3}\mathrm{T_s}\mathrm{hf_s}\Delta - \alpha\mathrm{P}(3) \tag{10}$$

Equation (8)(9)(10) is the power propagation equation, which is also the ordinary differential equation to be solved. The column vector P=[P_s; P_p; P_{ase}], where P_s is the signal power, P_p is the pump power, and Pase is the ASE power. After the parameters are determined, the above absorption cross section and emission cross section in the waveband is taken out of the maximum value of the amplitude to give σ_{sa} and σ_{se}. The wavelength range is defined as 1450~1520nm, and the number of wavelengths is measured as a function to determine the gain matrix dimension.

The signal light wavelength, σ_{21}, and σ_{21} are used as a function of wavelength, and the number of circular light wavelengths is solved by invoking the ordinary differential equation using ode45. The gain curves are plotted for different fiber lengths and doping concentrations, and the simulation functions are used as interface functions for the subsequent optimization process.

Table 1 below displays the TDFA's parameters. When uniform spreading is taken into account, the excited emission cross-sectional area of the 1460 nm band thulium-doped fiber amplifier is roughly equal to the exciting absorption cross-sectional area, which means σ_{se}=σ_{sa}=σ_s. In contrast, the spontaneous emission rate from energy level $^3H_4$ to $^3F_4$ is extremely tiny [7].

Table 1. Correlation coefficient of thulium-doped fiber amplifier.

Parameter	Value
Planck's Constant $\boldsymbol{h}$	6.626×10^{-34} J·s
Background Loss $\boldsymbol{\alpha}$	0.1 dB/m
the Velocity of Lightwave $\boldsymbol{c}$	3×10^{8} m/s
Spontaneous emission rate $\boldsymbol{A_{21}}$	108.6/s[8]
Wavelength of Main Pump $\boldsymbol{\lambda_p}$	790nm[5]
The wavelength of Signal $\boldsymbol{\lambda_s}$	1460nm
$\boldsymbol{\Gamma_p}$	0.45[9]
$\boldsymbol{\Gamma_s}$	0.45[9]
Fiber doping concentration N	$1.6\times1025/\mathrm{m}^3$

2.2.4. Cat group algorithm. The cat swarm optimization algorithm was selected for the optimization of the fiber amplifier gain curve, which can find the optimal value (extreme value) with a fast convergence rate.

2.2.4.1. Algorithm background. CSO is a heuristic optimization algorithm based on how a cat swarm behaves in the wild. The algorithm simulates the behavior of cats in foraging, hunting, and escaping, and simulates the group intelligence to find the optimal solution.

The cat group algorithm is mainly divided into two stages: the search stage and the aggregation stage. During the search phase, each cat moves randomly with a certain probability to search the new solution space and find the optimal solution through both local and global search. During the aggregation phase, each cat in the herd tries to move toward the optimal solution and gradually converges in the vicinity of the optimal solution. Compared with other heuristic algorithms, the cat group algorithm has the following characteristics [10]: (i) algorithm has strong global search ability and convergence speed. (ii) The algorithm is more robust to the selection of parameters such as initial population number, step size, and search space. (iii) The implementation of the algorithm is simple and easy to parallelize. (iv) The cat group approach has been utilized effectively to optimize a wide range of issues, including those in wireless sensor networks, power systems, image processing, and other areas.

2.2.4.2. Search mode. The principal flow of the search mode algorithm is described as follows [2]:

I. Copy *j* parts of the cat *n*, namely *j* = *SMP*, if *SPC* = TRUE, make *j* = *SMP*-1, and keep the current position as one of the candidates solutions.

II. Replace the previous value for each copy by adding or removing the *SRD* at random from the *CDC* 's current value:

$$\mathrm{X}_{cn} = (1 \pm SRD \times \mathrm{R}) \times \mathrm{X}_{c} \tag{11}$$

where X_c is the current location, X_{cn} is a new position, and R is an arbitrary value within the [0,1].

III. Find the fitness value *FS* for each potential option.

IV. The selection probability of each potential solution is determined from II if *FS* is not uniform; otherwise, the selection probability of each potential solution is set to.

$$|FS_i\text{-}FS_b| \tag{12}$$

$$P_i = FS_{max} - FS_{min}, \text{ where } 0 < i < j \tag{13}$$

where P_i is the selection probability of the current solution, FS_i is the fitness value of the cat, FS_{max} and FS_{min} are the maximum and minimum values of the fitness, respectively. For the maximization problem, $FS_b=FS_{min}$, for the minimization problem $FS_b=FS_{max}$.

V. From the candidate solution of the memory pool, replace the current cat *n*'s position by adhering to the selection probability.

2.2.4.3. Tracking mode. When the cat enters the tracking mode, it moves in accordance with the speed in each dimension, simulating the cat tracking the target [2].

I. Each cat *n* updates the speed of its current iteration following the following equation:

$$\mathrm{v}_{n,d}(t) = \mathrm{v}_{n,d}(t-1) + \mathrm{r}_1\mathrm{c}_1\left[\mathrm{x}_{\mathrm{B},d}(t-1) - \mathrm{x}_{n,d}(t-1)\right] \ \ , d = 1,2,3\ldots, \mathrm{M} \tag{14}$$

Where $x_{B,d}(t\text{-}1)$ indicates the location with the highest fitness value from the most recent iteration, and $x_{n,d}(t\text{-}1)$ is the position of the last iteration *n*. c_1 is a constant and r_1 is the random number between[0,1].

II. Check the speed to see if it falls within the maximum speed range, if it does, take the boundary value.

III. Update the location of *n* by the following equation:

$$x_{n,d}(t) = x_{n,d}(t-1) + v_{n,d}(t) \tag{15}$$

2.3. Result and discussion

2.3.1. Result. The simulation curve obtained by running the simulation code is shown in the figure6,7,8 below.

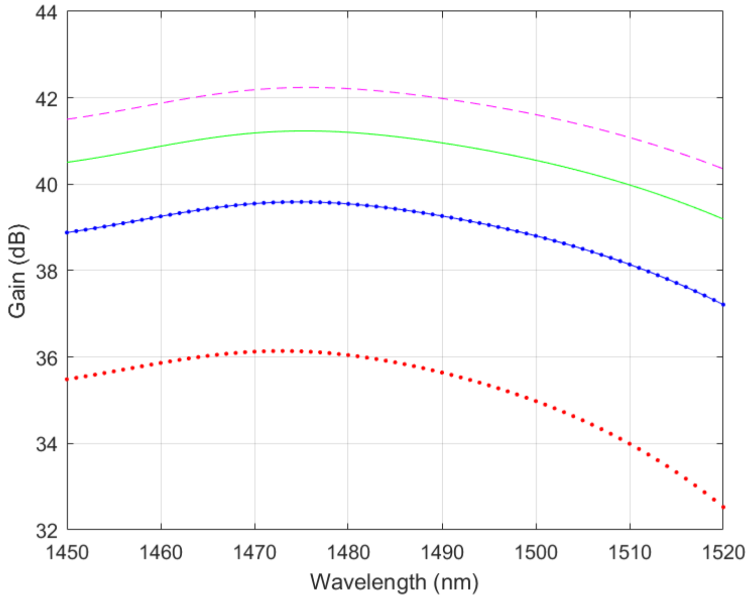


Figure 6. Simulation curve of amplifier gain regarding wavelength for various fiber lengths.

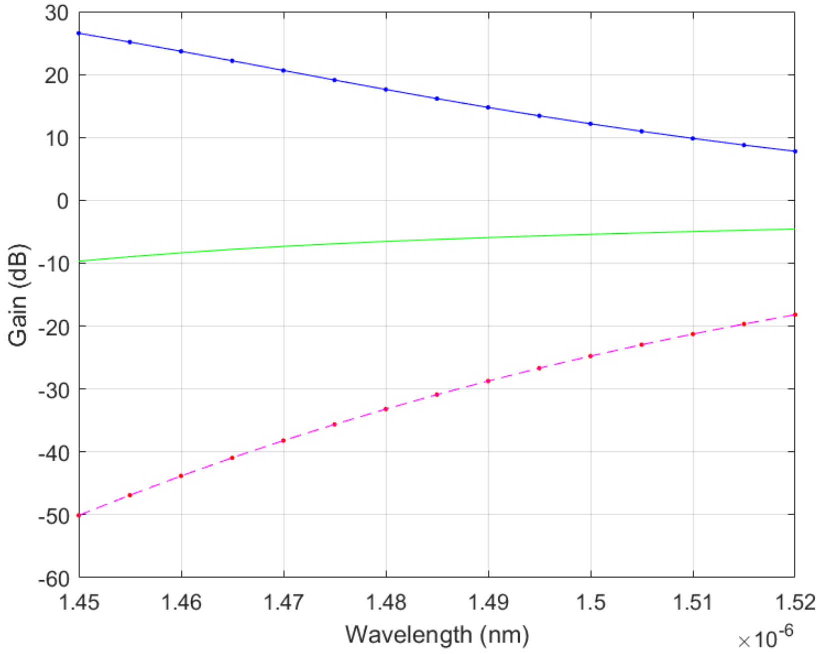


Figure 7. Simulation curve of amplifier gain with wavelength for various doping concentrations.

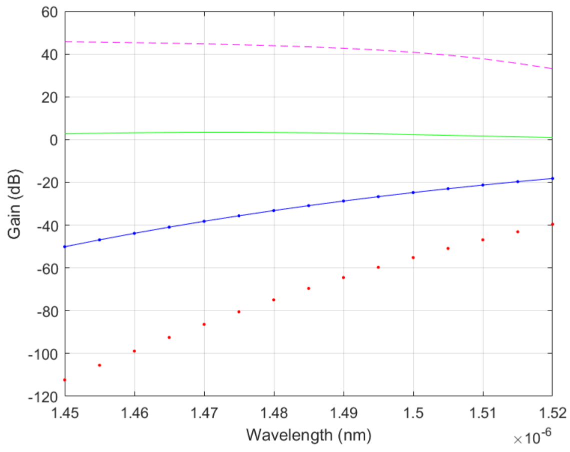
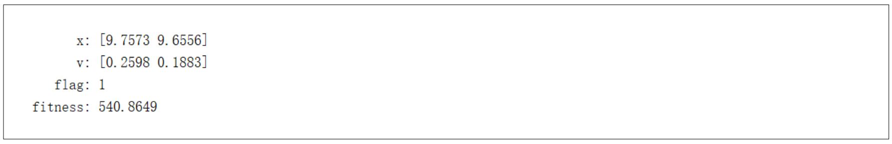


Figure 8. Simulation curve of amplifier gain with wavelength for various pump optical power.

The optimization results obtained by running the interface function by the cat colony algorithm are shown in figure 9 below.



```
      x: [9.7573 9.6556]
      v: [0.2598 0.1883]
   flag: 1
fitness: 540.8649
```

Figure 9. Optimization results.

Where x is a two-dimensional matrix, x(1) is the first optimization variable, namely the fiber length, and x(2) is the second optimization variable, namely the doping concentration. The optimization results show that when the gain is the maximum, the fiber length is 9.7m and the doping concentration is 9.6×1025mol/m³.

2.3.2. Discussion. After simulation and optimization, it is found that the cat colony optimization algorithm is feasible and effective in the gain optimization of the fiber amplifier. Based on the efficient function convergence rate and good optimization results of the cat colony optimization algorithm, the maximum value of the gain curve of the fiber amplifier can be obtained, which can be used in the design and application of a thulium-doped fiber amplifier.

3. Conclusion

After simulation and optimization, it is found that the cat colony optimization algorithm is feasible and effective in the gain optimization of the fiber amplifier. Based on the efficient function convergence rate and good optimization results of the cat colony optimization algorithm, the maximum value of the gain curve of the fiber amplifier can be obtained, which can be used in the design and application of the Tm^{3+}-doped fiber amplifier.

The simulation results show that the fiber length, doping concentration, and pump power have direct effects on the output signal gain characteristics of the Tm^{3+}-doped fiber amplifier.

The output signal gain of the fiber amplifier increases with the length of the fiber. The longer the fiber length, the more pump power that can be absorbed by the doped fiber, the higher the output signal power value, and the greater the signal amplification. The gain of the amplifier, however, will not increase indefinitely and will stop increasing once it reaches a particular level. There is an optimum length of fiber for the best performance of the amplifier's signal gain.

The doping concentration directly affects the absorption coefficient of pump light. The output light power increases with the doping concentration and the signal gain also increases correspondingly.

When the signal gain and pump power are constant, the larger the ion doping concentration is, the shorter the fiber length is required, which is conducive to the development of miniaturization and modularization of the system. However, the doping concentration cannot be increased continuously, and the performance of Tm^{3+}-doped fiber amplifiers will be seriously affected when the doping concentration exceeds a certain range. In other words, there is an optimal doping concentration in optical fiber for the signal amplification of the amplifier to obtain the best performance.

Pump power is also an important influence parameter. When the signal gain and doping concentration are fixed, the larger the pump power, the shorter the fiber length required, which is also conducive to the development of miniaturization and modularization of the system. However, the pump power cannot be continuously increased. On the one hand, the high pump power may exceed the bearing capacity of the doped fiber and burn the fiber; on the other hand, it may cause the waste of pump power and reduce the power conversion efficiency of the system. In other words, there is an optimal pumping power for the signal gain of the amplifier in terms of signal gain.

Therefore, in a Tm^{3+}-doped fiber amplifier, there is a matching relationship among fiber length, doping concentration, and pump power, which can make the amplifier obtain the optimal signal gain characteristics under certain conditions.

References

[1] Naftaly M, Shen S X and Jha A 2000 Tm^{3+}-doped tellurite glass for a broadband amplifier at 1.47 m. *Applied Optics* **39(27)** pp 4979-4984

[2] Chu S C, Tsai P W and Pan J S 2006 Cat Swarm Optimization *Int. Conf. Mach. Learn. Cybern.*

[3] Ren J J, He Z X, Yu T and Ye X S 2022 Progress of nanosecond thulium-doped fiber lasers in the 2 μm band

[4] Li Z, Heidt A M, Daniel J M O, Jung Y, Alam S U and Richardson D J 2013 Thulium-doped Fiber Amplifier for Optical Communications at 2m. *Optic. Exp.* pp 9289-9297

[5] Miniscalco W J 2001 Optical and Electronic Properties of Rare Earth Ions in Glasses

[6] Setsuhisa T 2001 Properties of Tm^{3+}-doped tellurite glasses for 1.4-um amplifier *Proc. SPIE - Int. Soc. Opt. Eng.* **4282** pp 85-92

[7] He C J, Chen X B, Sun Y G, Chen L and Meng C 2000 Spectroscopic properties of Tm^{3+} in fluorine oxide glasses *J. Beijing Norm. Univ. Nat. Sci. Edi.* **36(3)** pp 5

[8] Komukai T 1995 Upconversion pumped thulium-doped fluoride fiber amplifier and laser operating at 1.47 μm. *IEEE J. Quantum Electron.* **31(11)** pp 1880-1889

[9] Unknow 2003 Theoretical study of gain characteristics of 1064nm pumped thulium-doped fiber amplifier *China Laser* **30(9)** pp 5.

[10] Ahmedaram M, Rashidtarik A and Saeedsoran a M 2020 Cat Swarm Optimization Algorithm *Comput. Intell. Neuro.*

A new interpolation algorithm based on Hibbard-Laroche algorithm and its superiority

Yuxuan Huang[1, †],**Yiming Ren**[2, 3, †]
[1]Shaoguan University, Shaoguan, Guangdong Province, China, 512005
[2]University of Shanghai for Science and Technology, Shanghai, China, 200093

[3]Corresponding author: 1912140119@st.usst.edu.cn
[†]These authors contributed equally.

Abstract. In order to optimize the possible problems and improvements in the existing color image restoration interpolation algorithms, we conduct research based on the existing bilinear interpolation method, cok algorithm and Hibbard-Laroche algorithm. Our method is to use our own comparison method to compare different types of images through three algorithms to find the advantages and disadvantages and to some extent combine the advantages of bilinear interpolation and Hibbard-Laroche algorithm to try to innovate a new algorithm to compare with the existing three algorithms. The results show that the existing three algorithms have their own advantages in different scenarios, and the new algorithm is superior to the existing algorithms in terms of clarity and color restoration accuracy in most scenarios. However, due to the large computational complexity, the operation speed is slow.

Keywords: algorithm comparison, Hibbard-Laroche algorithm, a new algorithm, algorithm improvement.

1. Introduction

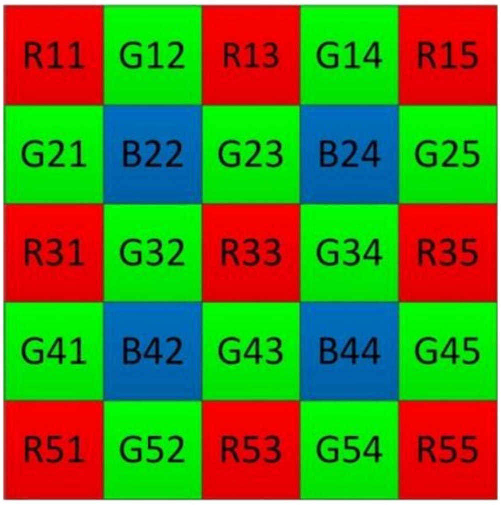
In daily life, it is not difficult to find that when our mobile phone is facing the computer screen, obvious stripes will appear. The photos we take are enlarged to see some details are distorted. These are all problems in color image restoration [1].

In order to solve the problem of color image restoration, scientists have proposed several famous interpolation algorithms, including bilinear interpolation method, color ratio constant method, and gradient edge interpolation method [2]. These algorithms cleverly use the average value, ratio, difference and so on of adjacent pixels in bayercfa for interpolation calculation.

However, there are some color interpolation distortion phenomena such as zipper effect and moire fringe effect [3]. When the image jumps from low frequency to high frequency, the interpolation is not along the boundary but through the boundary, and the boundary part will produce blur and color overflow. After interpolation, there are some regular interval distribution of pixels in the horizontal or vertical direction, which is called zipper effect. When the optical image frequency is close to the CCD pixel frequency, the frequency aliasing is generated, and the Moiré fringe is generated. In the processing of black and white high-frequency edge images, we found that the images restored by bilinear interpolation method and color ratio constant method have obvious edge jagged stripe grids, and the laroche method can be perfectly restored. This is because this algorithm considers the edge situation in the horizontal and vertical directions, but in fact, the laroche method does not perfectly eliminate the distortion phenomenon. When the image has uneven black and white stripes, it also has a certain degree of distortion. This is because the mechanism of judging the boundary is still flawed [4].

Based on the discovery that the Hibbard-Laroche algorithm is good for this edge, we began to consider that different algorithms may have different advantages for different images [5]. The selection of interpolation algorithms directly affects the final effect of the image.For different application fields, it is necessary to comprehensively consider the complexity and recovery effect of the interpolation algorithm and select the appropriate algorithm [6]. In order to deeply understand the principle of color image interpolation algorithm, this paper firstly introduces several commonly used image interpolation algorithms for digital image sensors based on Bayer format color filter array, then we improves the Hibbard interpolation algorithm and forms a new algorithm [7]. The simulation results of different styles of images are compared with the other three difference algorithms. Finally, the corresponding conclusions are given.

2. Related Works



R11	G12	R13	G14	R15
G21	B22	G23	B24	G25
R31	G32	R33	G34	R35
G41	B42	G43	B44	G45
R51	G52	R53	G54	R55

Figure 1. RGB filter.

2.1. *Bilinear algorithm*

As Figure 1 shows, in such a Bayer CFA, we take the average of the component values around each pixel as the value of the channel[5]. On the r_{53} pixel, take the picture as an example, find b_{53}, g_{53}.

$$b_{53} = \frac{(b_{42} + b_{44} + b_{62} + b_{64})}{4} \tag{1}$$

$$g_{53} = \frac{(g_{43} + g_{52} + g_{54} + g_{63})}{4} \tag{2}$$

Similarly on the B pixel. On the G pixel point, r_{54} and b_{54} are calculated as g_{54}.

$$r_{54} = \frac{(r_{53} + r_{55})}{2} \tag{3}$$

$$b_{54} = \frac{(b_{44} + b_{64})}{2} \tag{4}$$

By analogy, this is a bilinear interpolation method.

2.2. Cok algorithm

Because from the perspective of physical optics, in a small smooth neighborhood of the same object,the proportion of light intensity of the three color channels will not mutate, so there is the following law :

$$\frac{R_{ij}}{G_{ij}} = \frac{R_{mn}}{G_{mn}} \tag{5}$$

$$\frac{B_{ij}}{G_{ij}} = \frac{B_{mn}}{G_{mn}} \tag{6}$$

Therefore, we first use the bilinear interpolation method to calculate all the G components in the picture, and use the adjacent pixel R and B components to calculate the pixel R and B components to be solved [5].

$$R_{mn} = G_{mn}\left(\frac{R_{ij}}{G_{ij}}\right) \tag{7}$$

Multiple adjacent points were averaged as r_{53} cases:

$$b_{53} = g_{53} \times \frac{\frac{B_{42}}{G_{42}} + \frac{B_{44}}{G_{44}} + \frac{B_{62}}{G_{62}} + \frac{B_{64}}{G_{64}}}{4} \tag{8}$$

2.3. Hibbard-Laroche algorithm

2.3.1. According to the gradient to determine the possibility of the existence of the horizontal direction. Hibbard method : take the green component as an example, if $\alpha = |G_{43} - G_{63}|$, $\beta = |G_{52} - G_{54}|$, $\alpha<\beta$, The possibility of boundary in the vertical direction is small, and the interpolation is carried out along the vertical direction[5].

$$g_{53} = \frac{(g_{43} + g_{63})}{2} \tag{9}$$

Contrarily,

$$g_{53} = \frac{(g_{52} + g_{54})}{2} \tag{10}$$

Laroche method: taking red component as an example, if $\alpha = |2 \times R_{53} - R_{51} - R_{55}|$, $\beta = |2 \times R_{53} - R_{33} - R_{73}|$, $\alpha<\beta$, The possibility of boundary in the vertical direction is small, and the interpolation is carried out along the vertical direction, and the formula is the same as above.

2.3.2. Using the idea of constant color difference to restore the red and blue channel. Constant chromatic aberration is considered constant in a small smooth region of an image [5].

$$R_{ij} - G_{ij} = R_{mn} - G_{mn} \tag{11}$$

$$B_{ij} - G_{ij} = B_{mn} - G_{mn} \tag{12}$$

Therefore,

$$R_{i-1,j} = G_{i-1,j} + \frac{R_{i-2,j} - G_{i-2,j} + R_{i,j} - G_{i,j}}{2} \tag{13}$$

3. Simulation of three algorithms

3.1. Bilinear algorithm

Bilinear algorithm is also known as bilinear interpolation method. Firstly, select the material photo and read its rows and columns and the value of the channel and then define a zero matrix named bayer, only rows and columns, traverse the matrix, and assign the channel value of the selected material photo to bayer[8]. If the odd row even sequence is assigned to blue, the even row odd sequence is assigned to red, and the other is assigned to green. The bayer is expanded to form a new matrix, and then a zero matrix with rows and columns and channels is defined. Four loops and judgment statements are used to traverse each pixel point and interpolate it with bilinear interpolation formula. After transforming the data type, the new image is output[9].

3.2. Cok algorithm

Cok algorithm is also known as color ratio constant method. Firstly, the original bayer image is extracted and expanded as the bilinear interpolation method. The for loop and the judgment statement are used to traverse each pixel point after the bayer expansion and the bilinear interpolation method is used to calculate all the green components. The for loop and the judgment statement traverse all the pixels once by judging the odd and even rows and columns, and use the color ratio constant method formula to interpolate the red and blue, and output the new image after transforming the data type[10].

3.3. Hibbard-Laroche algorithm

Hibbard-Laroche algorithm is also known as gradient edge interpolation method. Get the image information and list the three channels RGB respectively, forming three matrices with only rows and columns. The three matrices are filled in the corresponding components in the original image, and then the green component is recovered first. The values of the horizontal gradient a and the vertical gradient b are calculated and compared. At the B and R components, the G component is obtained by comparing the a and b values. Then, the R and B components are calculated at the G component by using the formula of color difference constant method. At the B component, find the R component; at the R component, find the B component. After transforming the data type, a new image is output[4].

3.4. A new algorithm based on Hibbard-Laroche algorithm

3.4.1. Firstly, the image information is obtained, and then the values of the R, G and B components of the three channels on the bayer template are color restored according to the image information. By using the method of line-by-line traversal, the value of the red channel component in the original image is assigned to the odd column of the odd row, the value of the green channel component in the original image is assigned to the even column of the odd row and the even column of the even row, and the value of the blue channel component is assigned to the even column of the even row. Then write a simple function to compare the size of three numbers and get the maximum value.

The bayer template obtained above is divided into nine grids, and the average of each part of R, G and B components is calculated to obtain R1, G1 and B1 respectively.

```
for i=2:m/3
    for j=int32((n-1)*(2/3)):(n-1)
        if
            sum1=sum1+G(i,j);
            a1=a1+1;
        elseif
            sum2=sum2+R(i,j);
            a2=a2+1;
        else
          sum3=sum3+B(i,j);
            a3=a3+1;
        end
  end
end
G1=sum1/a1;
R1=sum2/a2;
B1=sum3/a3;
```

3.4.2. Next, according to the average value of G1, R1 and B1, we compare the size of the three numbers and get the maximum value function to judge which color channel in each region has the most components and calculate all the component values of the channel color in the region. The G value is calculated by the gradient calculation method mentioned in the Hibbard-Laroche algorithm, and the R and B values are calculated by bilinear interpolation method.

3.4.3. Finally, the idea of constant color difference is used to restore the color components of other channels in each pixel, such as when G is the background color in a block: $R_{ij} - G_{ij} = R_{mn} - G_{mn}$, $B_{ij} - G_{ij} = B_{mn} - G_{mn}$,When R is the background color: $G_{ij} - R_{ij} = G_{mn} - R_{mn}$, $B_{ij} - R_{ij} = B_{mn} - R_{mn}$, When B is the background color: $G_{ij} - B_{ij} = G_{mn} - B_{mn}$, $R_{ij} - B_{ij} = R_{mn} - B_{mn}$, Convert data types and output new images.

4. Experimental evaluation

In this chapter, we use three methods to quantitatively analyze the image processing results of the new algorithm and the other three old algorithms. First, we select five different types of images for algorithm simulation and scoring. Secondly, we compare the time required for the operation of the four algorithms and rank them, and add and subtract points according to the different rankings. Third, we adapted the image clarity scoring software to score the processing results of the four algorithms. Finally, we add the total scores of the three parts to intuitively reflect the advantages of the new algorithm.

4.1. Flow chart

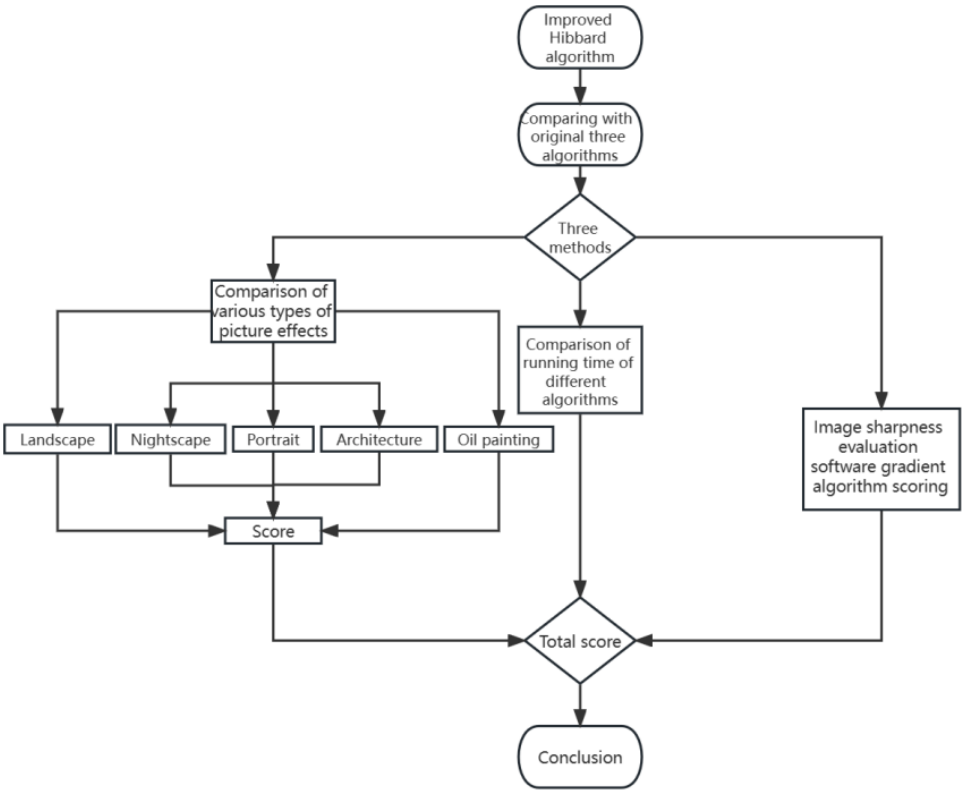

Improved Hibbard algorithm
Comparing with original three algorithms
Three methods
Comparison of various types of picture effects
Comparison of running time of different algorithms
Image sharpness evaluation software gradient algorithm scoring
Landscape
Nightscape
Portrait
Architecture
Oil painting
Score
Total score
Conclusion

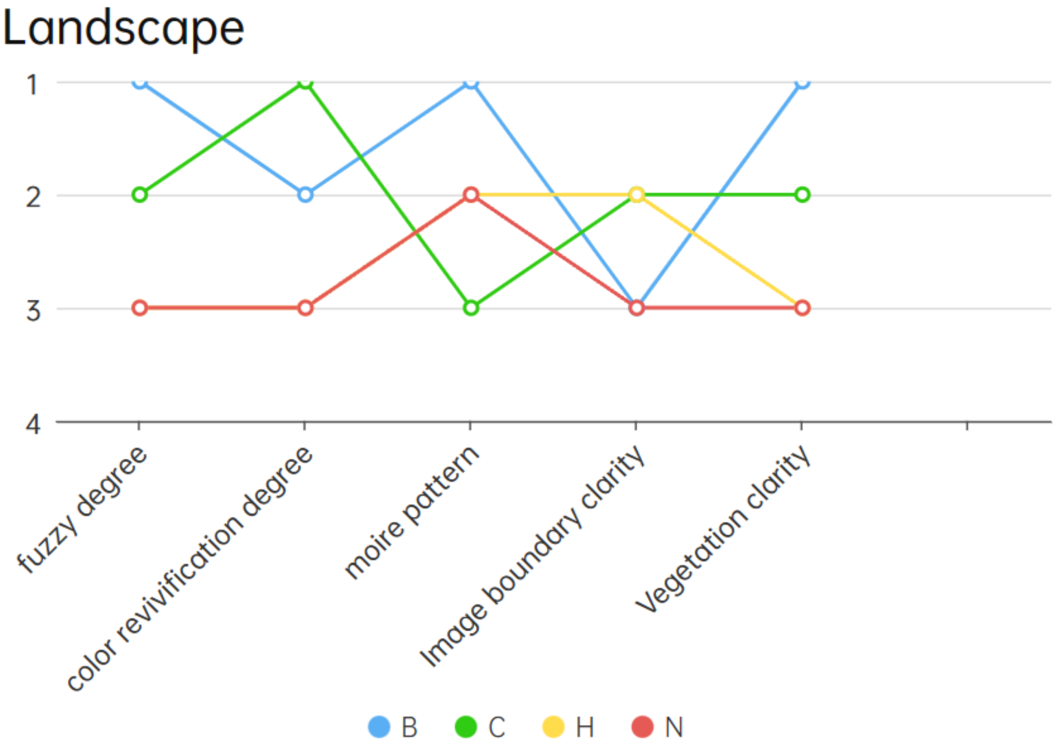
Figure 2. Flow chart.

4.2. Comparison of effects of various types of pictures

In order to compare the new algorithm with the other three old algorithms more comprehensively, we decided to select different kinds of pictures for simulation comparison. As the Figure 2. shows, we selected five types of pictures: landscape photos, buildings, night scenes, portraits and oil paintings for comparison. There are five evaluation criteria, four of which are fixed, and the remaining one will be adjusted according to the type of picture. The full score of each standard is 3 points, and the minimum is 1 point. The simulation results of each type of picture are 15 points, and the total score of the five types is 75 points. After the comparison, we will make a table to compare the total score and average score of the simulation results of each algorithm.

4.2.1. Landscape photos

Figure 3. Different landscape photos.

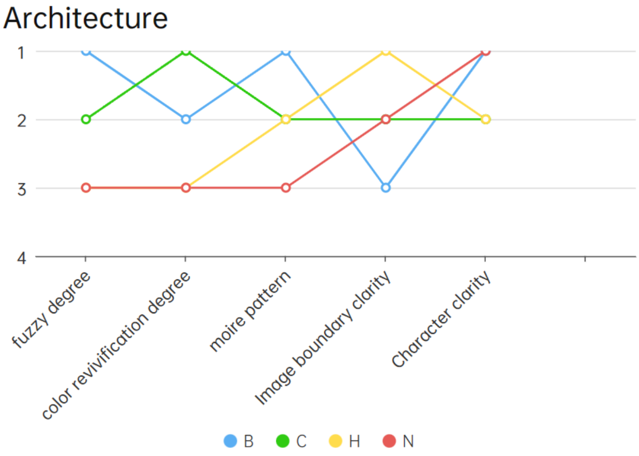


Effects:

Figure 4. Scores of various algorithms for landscape simulation.

4.2.2. Architectures

Figure 5. Different architecture photos.

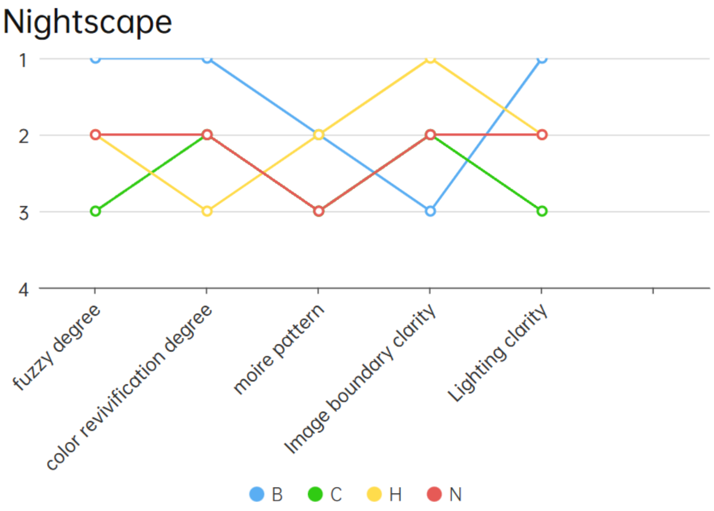


Effects:

Figure 6. Scores of various algorithms for architecture simulation.

4.2.3. Nightscape photos

Figure 7. Different nightscape Photos.

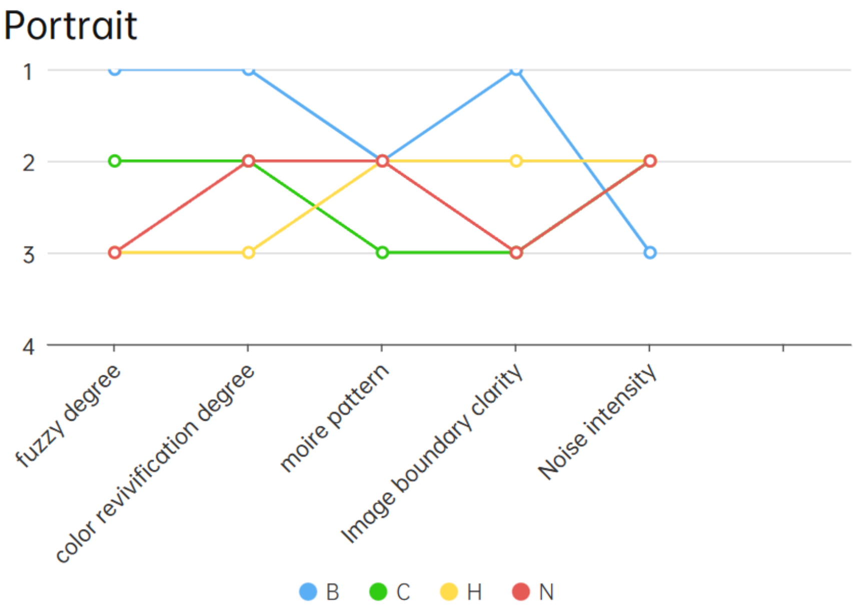


Effects:

Figure 8. Scores of various algorithms for nightscape simulation.

4.2.4. Portrait

Figure 9. Different Portraits.



Effects:

Figure 10. Scores of various algorithms for portrait simulation.

4.2.5. Oil paintings

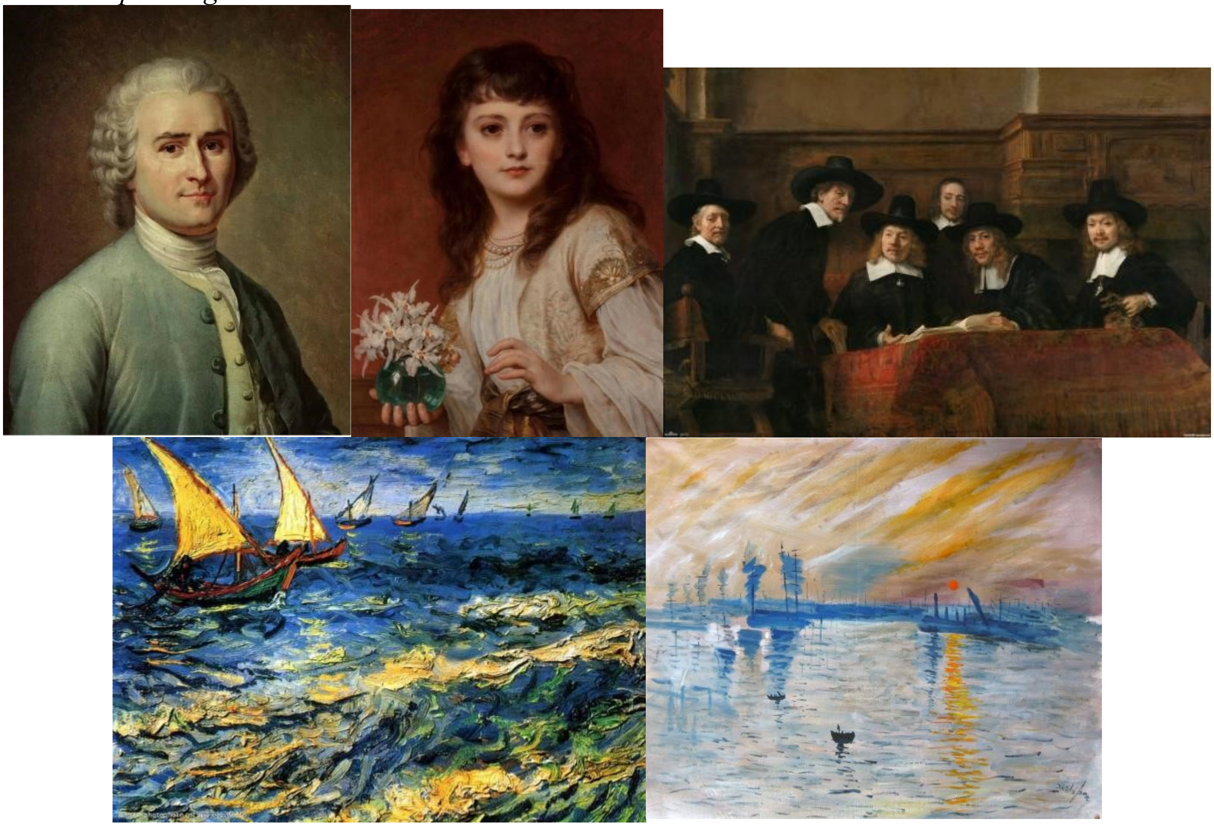


Figure 11. Different oil paintings.

Effects:

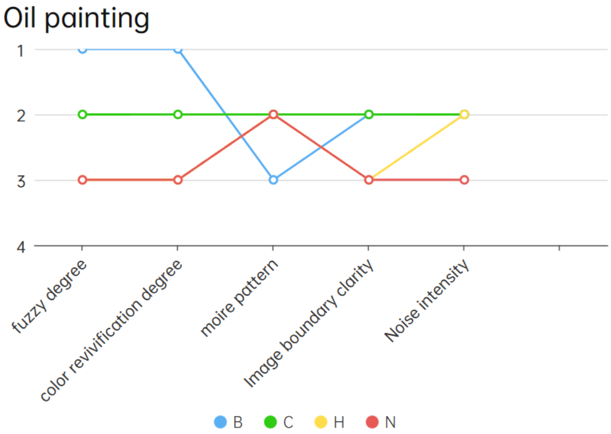


Figure 12. Scores of various algorithms for oil painting simulation.

4.2.6. Data summary

According to Figure 4,6,8,10 and 12, by adding the simulation results of five different types of pictures, we get the data of the following table.

Table 1. The scores of 4 algorithms for 5 different types of picture simulation.

Algorithm	Landscape	Architecture	Nightscape	Portrait	Oil painting	Overall	Average
Billinear	8	8	8	8	10	42	8.4
Cok	10	9	13	12	9	53	10.6
Hibbard	13	11	10	12	13	59	11.8
New algorithm	14	12	11	12	14	63	12.6

Note: The higher the score, the better thr effect.

*4.3. Comparison of the time required for various interpolation algorithms to run.*In Table 2, we get the data according to the running time of the four algorithms.

Table 2. The running time comparison of 4 algorithms.

Unit: second

Average running time	Billnear	Cok	Hibbard	New algorithm
Landscape	1.48	1.54	1.64	1.70
Architecture	1.69	1.64	1.58	1.68
Nightscape	2.13	2.21	2.50	2.80
Portrait	1.71	1.81	1.71	1.92
Oil painting	1.61	2.10	2.15	2.40
Average of all the types	1.72	1.86	1.92	2.10

It can be seen that the Cok algorithm is close to the Hibbard-Laroche algorithm, the Bilinear algorithm has the shortest running time, and the new algorithm takes the longest time. According to the ranking, the shortest algorithm adds 10 points, the second adds 5 points, the third adds 2 points, and the longest algorithm does not add points.

*4.4. Gradient algorithm scoring using image sharpness evaluation software.*The functions of the software used are as follows :

*4.4.1. Brenner function.*The gradient filter method, also known as the gradient filter method, only needs to calculate the difference between two pixels in the x direction, that is, to calculate the second-order gradient, with less calculation.

$$F=\sum_{x}\sum_{y}\{[f(x+2,y)-f(x,y)]^{2}\} \tag{14}$$

4.4.2. Tenengrad function. The Sobel operator is used to extract the gradient values of the horizontal and vertical directions of the pixel points. The Tenengrad function is defined as the sum of squares of the pixel gradient, and a threshold T is set for the gradient to adjust the sensitivity of the function.

Set Sobel convolution kernel to Gx,Gy, then the gradient of the image at the I point:

$$S(x,y) = \sqrt{G_x * I(x,y) + G_y * I(x,y)} \tag{15}$$

The Tenengrad value of the image is defined as : (where n is the total number of pixels in the image)

$$Ten = \frac{1}{n} * \sum_x \sum_y S\,(x,y)^2 \tag{16}$$

Or not average: Evaluation function F (k):

$$\mathrm{F(k)} = \sum_{\mathrm{x}} \sum_{\mathrm{y}} \quad [\mathrm{G(x,y)}]^2(\mathrm{G(x,y)>T}) \tag{17}$$

And T is the given edge detection threshold.

$$g_x = \frac{1}{4}\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad , \quad g_y = \frac{1}{4}\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{18}$$

The weight of the two is 50 %, and the original image is 100 points.

4.4.3. Effects of Software Assessment

Table 3. Software scoring of image clarity after 4 algorithms processing.

Types	Billnear	Cok	Hibbard	New algorithm
Landscape	39.6818	58.3340	76.9311	82.4084
Architecture	43.1878	67.7712	80.3566	86.2876
Nightscape	60.1294	86.9294	97.4237	96.5636
Portrait	44.5932	74.1739	90.3777	94.7440
Oil painting	39.4804	84.8219	89.7351	87.8616
Average score	45.4145	74.4061	86.9648	89.5730

Note: The weight of the two functions is 50% respectively, and the original image is 100 points.

It can be seen that the new algorithm has the highest clarity of the image, the Hibbard-Laroche algorithm is in the second place, the Cok algorithm gets the third hightest score, and the Billinear algorithm has the lowest score.

4.5. Total score

In Figure 13, by summarizing the data of the above three different algorithms for the processing results of the pictures, we obtain the scores of the following table.

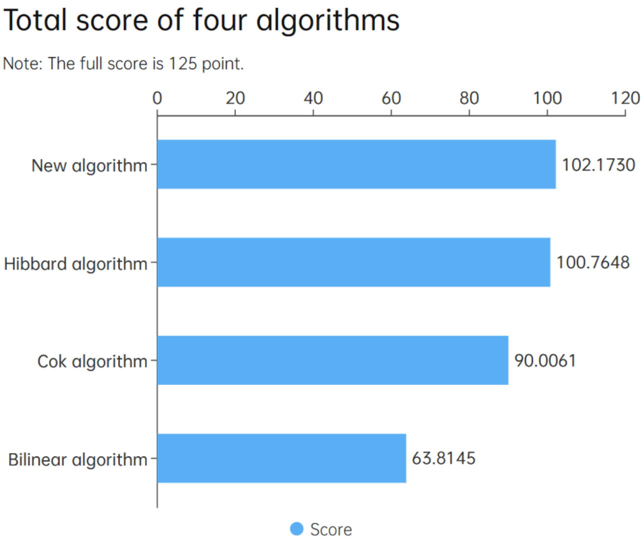


Figure 13. Total score of 4 algorithms.

5. Conclusion

As shown in the figure, the three evaluation methods are combined to obtain the highest score of the new algorithm. The Hibbard-Laroche algorithm has the second highest score, the Cok algorithm is the second, and the Bilinear algorithm has the lowest score. It can be seen that the new algorithm is the most restoration of image clarity and has obvious advantages. However, the three algorithms are also different in different types of images. In general, different interpolation algorithms for different images have different recovery effects. For example, Cok algorithm can be selected when night scene images need to be processed, but the effect is better than Hibbard-Laroche algorithm and new algorithm. However, in general, the image restored by the adaptive color layer interpolation algorithm has the best effect and the bilinear interpolation has the lowest computational complexity. This also leads to the calculation speed of the bilinear difference algorithm is the fastest of the three algorithms. In summary, although the new algorithm is not as fast as the original three algorithms, the new algorithm is the best choice if both the image effect and the computational complexity are taken into account.

Appendixes

The code of 3.4.2:

```
if max(G1,R1,B1)==G1
    if
        if a>b

        elseif a<b

        else

        end
    end
elseif max(G1,R1,B1)==R1
     if

     elseif

     elseif
```

```
            end
         else
            if

            elseif

            elseif

            end
        end
    end
end
```

References

[1] Jie X, Li-na H, Guo-hua G, et al. Real color image enhancement based on the spectral sensitivity of most people vision and stationary wavelet transform[C]//2009 2nd IEEE International Conference on Computer Science and Information Technology. IEEE, 2009: 323-328.

[2] HUA Ying,PENG Hongjing. An Image Interpolation Algorithm for Single CCD Image Sensor[A]. Nanjing:School of Information Science and Engineering,Nanjing University of Technology,2010:570-07.(in Chinese)

[3] Lamb A B, Khambete M. Image Quality Assessment Database for Demosaicing Artifacts[C]//2018 International Conference on Communication and Signal Processing (ICCSP). IEEE, 2018: 1100-1105.

[4] Xu T, Yu M. An improved Hibbard interpolation algorithm based on edge judgement[C]//International Conference on Algorithms, High Performance Computing, and Artificial Intelligence (AHPCAI 2021). SPIE, 2021, 12156: 35-42.

[5] LIU Sai. Research and comparison of classical interpolation algorithm based on color image[A].Suzhou:Suzhou University of Science and Technology,2020:049-04. (in Chinese)

[6] Lu J. Analysis and Comparison of Three Classical Color Image Interpolation Algorithms[C]//Journal of Physics: Conference Series. IOP Publishing, 2021, 1802(3): 032124.

[7] HE Qin,LIU Wenyu. Research on color interpolation algorithm of digital image sensor[A].Wuhan:Huazhong University of Science and Technology,Wuhan National Laboratory of Optoelectronics,2007:1482-04. (in Chinese)

[8] Wang J F, Wang C S, Hsu H J. A novel color interpolation algorithm by pre-estimating minimum square error[C]//2005 IEEE International Symposium on Circuits and Systems (ISCAS). IEEE, 2005: 6288-6291.

[9] Cloutier E, Beaulieu L, Archambault L. On the use of polychromatic cameras for high spatial resolution spectral dose measurements[J]. Physics in Medicine & Biology, 2022, 67(11): 11NT01.

[10] Robert C. Implementing Process Color Printing by Colorimetry[J].

Research on pathologic myopia recognition based on vision transformer

Chen Yang

School of Information Science and Technology, Beijing University of Chemical Technology, Beijing, 102200, China

2020040267@mail.buct.edu.cn

Abstract: Currently, the diagnosis of pathological myopia is mostly done through manual diagnosis, which not only requires experienced ophthalmologists but is also time-consuming and labour-intensive. In order to improve the diagnostic efficiency and accuracy, and to prevent irreversible visual impairment caused by missed diagnosis, misdiagnosis, and delayed treatment, this paper presents a fine-grained image analysis task of classifying fundus images of patients with pathological myopia and non-pathological myopia. To accurately identify subtle differences in features among similar fundus images, a pathological myopia recognition model based on Vision Transformer (ViT) is proposed. The model incorporates a feature selection module using self-attention mechanism that can effectively select important features in the fundus images, thereby eliminating the influence of irrelevant regions on recognition. Experimental results demonstrate that this method outperforms traditional ViT models, achieving high accuracy in pathological myopia recognition.

Keywords: pathologic myopia, fine-grained, ViT, feature selection.

1. Introduction

Pathological myopia, also known as degenerative myopia, is a type of near-sightedness that goes beyond the normal range and can lead to vision loss. It is caused by excessive elongation of the eyeball, leading to structural changes in the eye. Symptoms of pathological myopia may include blurred vision, difficulty seeing objects at a distance, eye strain or fatigue, headaches, and an increased risk of retinal detachment or other eye complications. As the condition progresses, patients may experience a progressive loss of vision that cannot be corrected with eyeglasses or contact lenses. Early diagnosis and treatment are important to prevent complications and preserve vision.

The difficulty in classifying the fundus images of patients with pathological myopia and non-pathological myopia lies in two main aspects: similarity and variability. For similarity, in both pathological and non-pathological myopia fundus images, the macular area may show some degree of deformation or thinning. In terms of variability, the interpretation of fundus images may be influenced by different factors such as intraocular pressure, age, genetic factors, etc. This can pose a challenge in identifying and classifying pathological myopia and non-pathological myopia.

The current diagnosis of pathological myopia relies mainly on experienced ophthalmologists who manually diagnose patients based on a comprehensive eye examination. This process is not only time-consuming and labor-intensive but also difficult to achieve accurate diagnosis, especially in developing countries or impoverished areas with a shortage of experienced ophthalmologists and inadequate medical facilities. This can lead to irreversible vision loss due to delayed treatment. As approximately 89% of people with visual impairments live in low- and middle-income countries [1], visual impairment and blindness remain significant challenges in underdeveloped countries and related poverty-stricken areas. Therefore, an efficient and automated machine diagnosis method is required to assist doctors in making timely diagnosis decisions without requiring massive human involvement or medical device intervention, which can lay a solid foundation for future remote medical assistance.

This paper proposes a ViT-based fundus feature selection model for pathological myopia recognition tasks. The model first divides a complete image into equally sized image patches. Then, these image patches are inputted into the encoder of the Transformer, which learns the self-attention weights between each image patches. Next, the model selects the image patches with higher contribution to the classification based on the self-attention weights. Finally, the selected image patch features are inputted into the classifier. The proposed method using self-attention to select image patches can reduce the influence of similar parts between different categories of fundus images and achieves good classification performance on the iChallenge-PM which is a dataset of pathological myopia recognition.

2. Related Work

Traditional convolutional neural networks, such as AlexNet [2], ResNet [3], and GoogleNet [4], have achieved good classification results for the coarse classification of images. However, traditional classification models extract features for the entire image without focusing on subtle differences in features, which results in these classification networks performing poorly on fine-grained image classification.

In order to extract local detail features with discrimination, many previous methods need to rely on local feature labels of images. Part-based R-CNNs extends R-CNN architecture by introducing a new branch to predict object part locations, in addition to bounding boxes and class labels [5]. This allows for more fine-grained localization and segmentation of objects in images. Besides, based on manually labeled strong part annotations, PS-CNN uses a fully convolutional network to locate parts and a dual-stream classification network to encode features of objects and parts [6]. However, the most common situation of clinical medical image data is with a small amount of labelled data and a large number of raw images and these approaches requires labeling local features of the target, which consumes a significant amount of human labor to annotate data [7].

When it comes to specific diagnostic methods for pathologic myopia, Liu et al. proposed the PAMELA (Pathological Myopia Detection Through Peri-papillary Atrophy) system, which automatically receives retinal fundus images, performs region of interest (ROI) extraction and optic disc segmentation, and uses support vector machine (SVM) to automatically diagnose pathological myopia based on the feature of peripapillary atrophy (PPA) in a dataset containing 80 fundus images [8]. Zhang et al. utilized the Minimum Redundancy-Maximum Relevancy (mRMR) feature selection technique to select and rank candidate features, and then used SVM classifier to diagnose pathological myopia [9-10]. However, these approaches belong to machine learning, which requires manual feature extraction and selection, resulting in relatively high workload.

3. Method

3.1. Vision Transformer (ViT)

Transformer [10] was originally proposed for the field of natural language processing (NLP) and achieved great success in this area. Dosovitskiy et al. were inspired by this and introduced the Vision Transformer (ViT) model [11]. Without modifying the original Transformer structure, this model applies Transformer to the field of vision by dividing images into patches, and has achieved very good results.

An overview of ViT is depicted in Figure 1. The model first crops the original image into fixed-sized patches, these patches are flattened and converted into sequences of embeddings which are fed into a transformer encoder. The transformer encoder processes the embeddings through multiple layers of self-attention and feed-forward networks, allowing the model to learn the relationships between the patches and capture spatial information across the image. The output embeddings from the final transformer layer are fed into a feedforward neural network (classifier head) that predicts the class label of the input image.

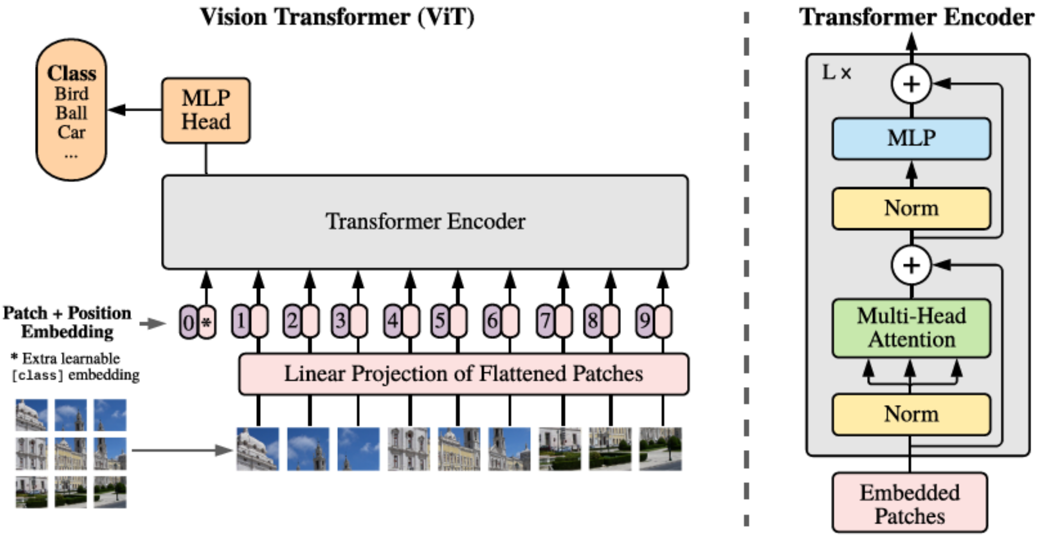


Figure 1. ViT overview.

3.2. *Feature Selection Module*

Attention mechanism is a computational technique used in machine learning and artificial intelligence to enable models to focus on specific parts of input data while processing information. The attention mechanism assigns weights to different parts of the input data based on their relevance to the task at hand.

The attention mechanism is commonly used in natural language processing (NLP) tasks such as machine translation, sentiment analysis, and text classification. In these tasks, the attention mechanism can help the model to identify the most important words or phrases in a sentence or document. For instance, the self-attention mechanism proposed by Vaswani et al. has improved the performance of Transformers in many natural language processing tasks compared to previous RNN-based approaches [10]. The original ViT model uses self-attention mechanism in the transformer layers. Every transformer layer takes all image patches as input and does not eliminate the influence of irrelevant image regions on the classification results. Pathological myopia recognition belongs to fine-grained image classification, which requires focusing on representative features of the image rather than all regions of the image.

Using the self-attention maps outputted by the Encoder, this paper proposes a self-attention-based feature selection module that selects feature vectors that contribute more to pathological myopia recognition, reducing the influence of useless image patches. An overview of the proposed model is shown in Figure 2.

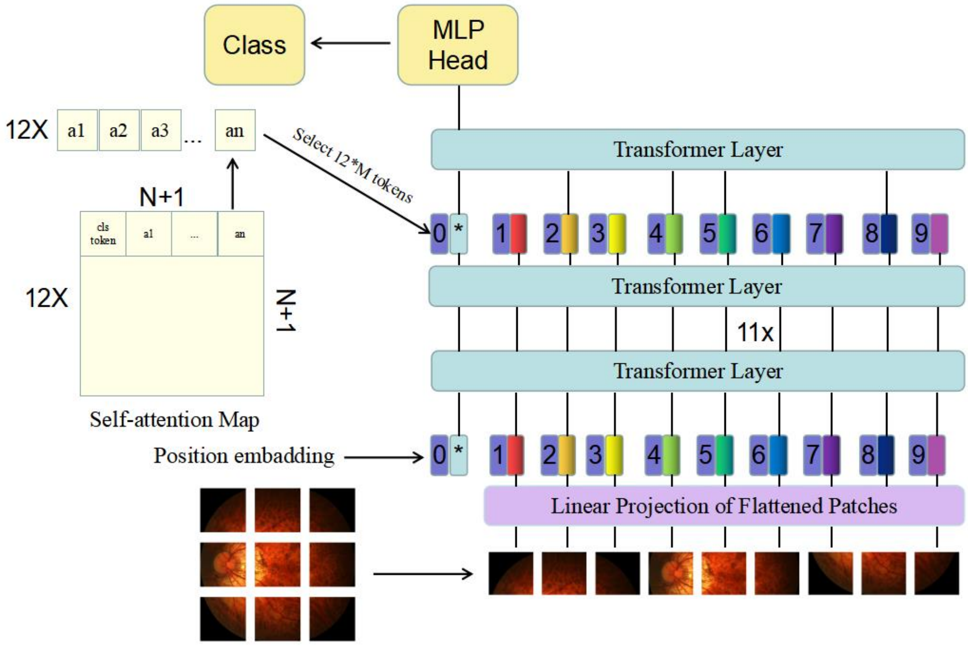


Figure 2. Proposed model overview.

The input image is transformed into (N+1) high-dimensional feature vectors (including a vector that represents the class of the input image) through linear projection. Next, the N feature vectors are inputted into the Encoder of the Transformer. The Encoder uses its Multi-Head self-attention module to calculate the weight size between (N+1) feature vectors and obtains K (K is the number of heads) (N+1) x (N+1) self-attention maps. These self-attention maps represent the correlation between each feature vector and other feature vectors, with a larger attention value between two feature vectors indicating a stronger relationship.

Although N image patches contain different parts of the original image, the contribution of different image patches to pathological myopia recognition is not the same. For example, image patches containing important parts such as the macula, retinal vessels, and optic disc are more important than other image patches. To select important image patches, a feature selection module was added to the penultimate transformer layer, which can pick out image patches that are strongly associated with the class vector. According to the method in TransFG [12]. The original output of layer L-1 is

$$Z_{L-1} = [Z_{L-1}^{0}; Z_{L-1}^{1}, Z_{L-1}^{2}, \dots, Z_{L-1}^{N}] \tag{1}$$

The attention weights of the previous layers can be written as follows:

$$a_l = \left[a_l^0, a_l^1, a_l^2, \dots, a_l^K\right]\ l \in 1,2, \dots, L-1 \tag{2}$$

The attention weight in each head is

$$a_l^i = \left[a_l^{i_0}; a_l^{i_1}, a_l^{i_2}, \dots, a_l^{i_N}\right]\ i \in 0,1, \dots, K \tag{3}$$

A matrix multiplication to the raw attention weights in all the layers as

$$a_{\text{final}} = \prod_{l=0}^{L-1} a_l \tag{4}$$

The a_{final} contains K (N+1) x (N+1) self-attention maps. The feature selection module selects the first row of attention values except the first value from each self-attention map. These N attention values represent the correlation between the N feature vectors outputted by the Encoder and class vector. The tokens corresponding to A1, A2,..., AK*M, which are the index of top M maximum values in each row, are selected and concatenated with the classification token as input sequence which is denoted as

$$\boldsymbol{z}_{\text{local}} = \left[z_{L-1}^{0}; z_{L-1}^{A_1}, z_{L-1}^{A_2}, \cdots, z_{L-1}^{A_{K*M}}\right] \tag{5}$$

The feature selection module not only preserves global information, but also allows the model to pay more attention to subtle differences between different categories.

4. Experiments

4.1. Datasets

The dataset used in this experiment is the iChallenge-PM dataset, which is a medical dataset on Pathologic Myopia (PM) provided during the iChallenge competition jointly organized by Baidu Brain and Zhongshan Ophthalmic Center of Sun Yat-sen University. The dataset includes 800 fundus retina images from subjects categorized into two classes: pathologic myopia and non-pathologic myopia. The non-pathologic myopia class includes two sub-classes: highly myopic and normal vision. In this experiment, the dataset was split into a training set of 480 images a validation set of 140 images and a test set of 140 images, with a ratio of 6:2:2.

4.2. System environment and experimental setup

The system is based on a 64-bit Windows operating system, and equipped with an AMD Ryzen 7 4800H CPU and a NVIDIA GeForce RTX 2060 GPU. The image in this experiment is resized to a size of 224*224 and divided into 196 image patches, with each block consisting of 16×16 pixels. The number of heads in the multi-head attention mechanism is set to 12, which means that in the feature selection module, 12 most important feature vectors will be selected from the 196 feature vectors.

The SGD optimizer provided by PyTorch is used in the model training. The hyperparameters of the learning rate are set to 0.001, momentum factor to 0.9, and weight decay to 5E-5. In addition, a learning rate scheduler based on the cosine annealing strategy is also utilized. The period of the cosine function is set to 10, and the scheduler includes a lower bound value, 0.01, which represents the minimum value of the learning rate. This optimizer and scheduler are chosen because they have performed well on similar tasks and datasets, and can effectively control the learning rate and convergence speed of the model. The model is trained for 10 epochs and a pre-trained ViT model—ViT-B_16 provided by Google is used in this experiment.

4.3. Result

In order to compare with the method proposed in this paper, ViT is tested for classification accuracy on the iChallenge-PM dataset. The experimental results are shown in Table 1. The classification accuracy of the pathological myopia recognition model based on ViT proposed in this paper is 94.9%, which is 2.5% higher than using the ViT model alone. This demonstrates that the feature selection module has made good selections of important local features in the images and eliminated the influence of irrelevant features on classification.

Table 1. Comparison of different methods on iChallenge-PM dataset.

Method	Backbone	ACC (%)
ViT	ViT-B_16	93.1%
Ours	ViT-B_16	97.5%

Figure 3 shows the accuracy and loss values of two models on the validation set.

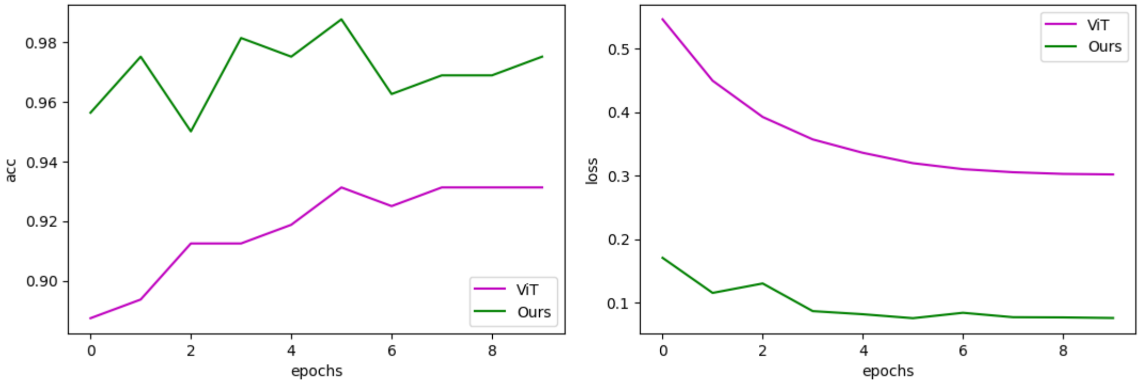


Figure 3. Performance of two models on the validation set.

This experiment also tested the impact of different M values on accuracy in the feature selection module, as shown in Table 2. When M is set to 6 or 8, the model achieves the highest accuracy on the iChallenge-PM dataset, which is 97.5%.

Table 2. Ablation study on value of M on iChallenge-PM dataset.

Value of M	ACC(%)
1	96.9
2	96.9
6	97.5
8	97.5
10	97.3

5. Conclusion

This paper proposes a ViT-based model which contains the feature selection module and study the impact of the number of selected features on the classification accuracy to address the issue of pathological myopia recognition. The feature selection module uses self-attention mechanisms to select important parts of the image, reducing the impact of irrelevant areas on classification. This approach demonstrates good recognition of similar but different classes of fundus images. On the iChallenge-PM dataset, the proposed model has higher classification accuracy compared to the ViT model.

References

[1] Zheng L, Yang Y, Tian Q. SIFT meets CNN: A decade survey of instance retrieval. 2017, IEEE transactions on pattern analysis and machine intelligence, 40(5): 1224-1244.

[2] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural

networks. 2017, Communications of the ACM, 60(6): 84-90.

[3] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition, 2016, IEEE conference on computer vision and pattern recognition. 770-778.

[4] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions. 2015, IEEE conference on computer vision and pattern recognition. 1-9.

[5] Zhang N, Donahue J, Girshick R, et al. Part-based R-CNNs for fine-grained category detection,2014, Computer Vision–ECCV 2014: 13th European Conference: 834-849.

[6] Huang S, Xu Z, Tao D, et al. Part-stacked CNN for fine-grained visual categorization, 2016, Proceedings of the IEEE conference on computer vision and pattern recognition. 1173-1182.

[7] Wang Z, Li T, Zheng J Q, et al. When cnn meet with vit: Towards semi-supervised learning for multi-class medical image semantic segmentation, Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, 424-441.

[8] Liu J, Wong D W K, Lim J H, et al. Detection of pathological myopia by PAMELA with texture-based features through an SVM approach. 2010, Journal of Healthcare Engineering, 1(1): 1-11.

[9] Zhang Z, Cheng J, Liu J, et al. Pathological myopia detection from selective fundus image features. 2012 7th IEEE Conference on Industrial Electronics and Applications: 1742-1745.

[10] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. 2017, Advances in neural information processing systems, 30.

[11] Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16x16 words: Transformers for image recognition at scale, 2020 arXiv preprint arXiv:2010.11929.

[12] He J, Chen J N, Liu S, et al. Transfg: A transformer architecture for fine-grained recognition. 2022, Proceedings of the AAAI Conference on Artificial Intelligence. 36(1): 852-860.

Using end-to-end learning and PyAutoGUI to apply gesture recognition for human-computer interaction

Junhao Tang

School of Mechanical, Electrical & Information Engineering, Shandong University, Weihai, 264200, China

202000800119@mail.sdu.edu.cn

Abstract. Contact-less human-machine interaction is becoming increasingly important due to the growing number of special environmental needs and accessibility situations. Gesture recognition has also been a hot topic in computer vision and machine learning in recent years. In this paper, a real-time computer manipulation system based on hand gesture recognition is studied and deployed. A relatively mature end-to-end target recognition model, the YOLOv5 model, is trained in this paper to achieve real-time detection and recognition of hand gestures. According to the result of the recognition, it is translated into the corresponding operation on the computer according to a set of rules, and then PyAutoGUI is used to actually control the computer. At the end of the research, the trained YOLOv5 model exhibited excellent performance and verified the feasibility and scalability of the solution. This is a good inspiration for developing a more convenient and efficient related software.

Keywords: computer vision; end-to-end learning; hand gesture recognition; YOLOv5; human-computer interaction.

1. Introduction

Hand gesture recognition is a technology that enables computers to recognize human hand movements and translate them into data that the computer can use. In the past, hand gesture recognition required special input devices, such as data gloves, but now this technology can be accomplished simply by analyzing the image signal acquired by the video input device, such as color camera. Further, hand gesture recognition has already become an important research direction in computer vision. It has been widely used in virtual reality, smart home, medical, and gaming.

Hand gesture recognition is more natural and convenient than traditional human-computer interaction, and more conform to human intuitive habits of use. It broadens the freedom of humans to use machines and makes it possible that get rid of the limitations of traditional hardware. Meanwhile, hand gesture recognition as a touchless data entry method provides a viable solution in situations where physical contact is not available or where there is a need for accessibility. For example, in healthcare, it can be a reliable way to allow disabilities to interact with electronic devices, which allowing them to live more independently. Because of the need like these, the speed (usually need to run in real time), accuracy (not to get the exact opposite of what the user expects), and reliability (the process is always working properly) of hand gesture recognition are very important.

In recent years, with the development of deep learning technology, gesture recognition methods based on deep learning have become popular of research. Deep learning models can learn features automatically and have high recognition accuracy and robustness. Although some of the existing studies have proposed and validated many hand gesture recognition solutions, most of the studies have just used traditional machine learning method [1] and dedicated sensors [2] to recognize gesture for computer control [3,4,5]. Or some of the solutions that use end-to-end learning only do the recognition part [6]. Therefore, this paper proposes a computer control and data input system with hand gesture recognition based on end-to-end learning. This paper focus on improving the speed of manipulation computers with hand gesture by simplifying the complexity of the data processing in recognition.

This paper chooses You Only Look Once v5(YOLOv5) as end-to-end learning algorithm. Compared with the previous version, YOLOv5 has improved the model structure and performance. The backbone network used in YOLOv5 is CSPDarknet [7]. Cross-Stage-Partial (CSP) structure can improve the efficiency and accuracy of the model. In addition, YOLOv5 also adopts a new method called Swish Activation, which can make the model converge faster [8]. In practical applications, YOLOv5 enables real-time target detection and achieves leading performance on some benchmark datasets, such as COCO. When YOLO recognizes hand gestures from the video image captured in the camera, the system sends the results to a computer-controlled program using PyAutoGUI. PyAutoGUI is an automation framework for python. It can be programmed to emulate input devices such as mice and keyboards or to control applications directly. PyAutoGUI is available on Windows, Mac and Linux, and supports multi-monitor control, making it possible for use in complex environments with multiple systems [9]. The combination of the two, YOLOv5 and PyAutoGUI, is the key to implementing gesture-controlled computers in this paper. In this way, real-time and efficient hand gesture recognition can be achieved, in order to control the computer with hand gestures.

2. Methods

2.1. System flowchart

Figure 1 is the complete system flowchart. The system has two parts, model training part and control part. The model training part trains the YOLOv5 model with the hand gesture dataset. When it is iterated to have a good performance, the system uses trained model for gesture recognition of images captured from the camera. The results of the recognition are passed to the control part, and the PyAutoGUI controls the computer accordingly to the different hand gestures.

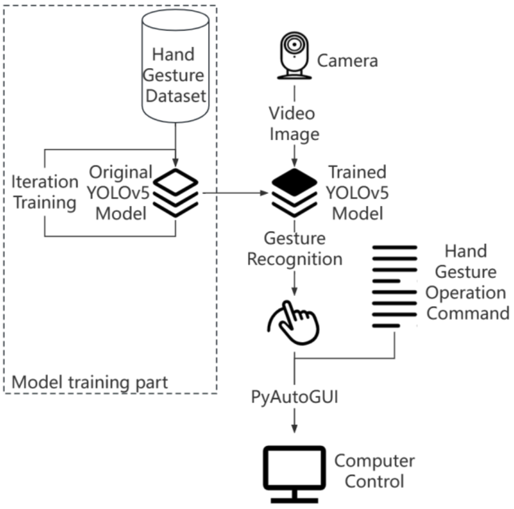


Figure 1. System flowchart, from model training to using gesture to control computer.

2.2. Data acquisition and pre-processing

The dataset for this study consists of two parts. The first part is the public dataset HAnd Gesture Recognition Image Dataset (HaGRID) provided by SberDevices, a Russian IT company. The dataset has a total of over 550,000 data, including 18 representative easily recognizable hand gestures, and a "no gesture" extra class [10]. All the data images are in 1920*1080 resolution. To improve training efficiency and reduce training time, the training set needs to be simplified. Here, reducing the image resolution does not affect the training results too much, and not all gestures are needed, so the simplified HaGRID dataset is used in the end. The simplified dataset has a total of about 40,000 data, which contains 6 class of hand gestures in Figure 2.

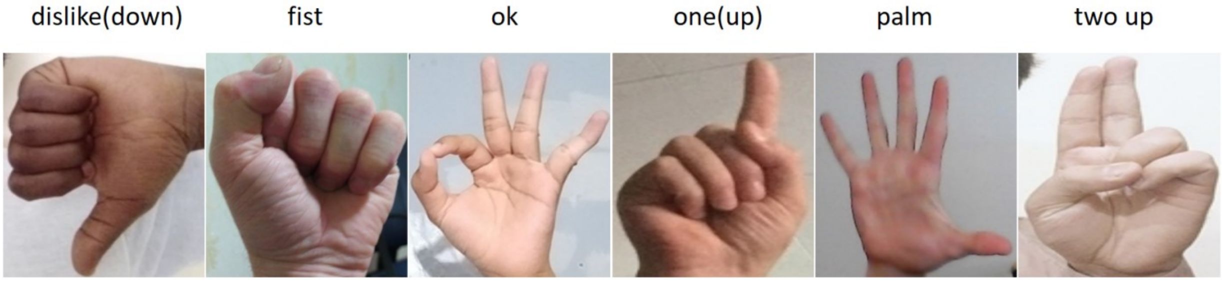


Figure 2. Selected 6 hand gesture (The names in brackets are modified by the author for subsequent use)

The second part of the data was created manually. In this study, two types of gesture data were added to improve the functionality of hand gesture manipulation. They are the leftward and rightward pointing hand gestures, like Figure 3.

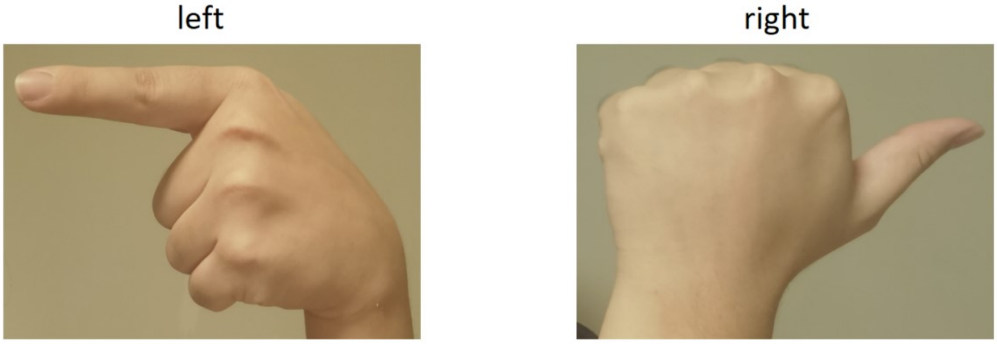


Figure 3. Leftward and rightward pointing hand gestures.

2.3. End-to-end learning-based hand gesture recognition

In this paper, the YOLOv5 model developed by Ultralytics is used for training. PyTorch framework was first applied to YOLOv5 [7], making it easier and more convenient to deploy. Meanwhile, PyTorch's well-established community provides great support for the implementation of the solution.

YOLOv5 was released in multiple versions with some disparity in performance. According to YOLO publishers [11] and related studies [12,13], YOLOv5x performs the best in accuracy, but YOLOv5s improves detection speed with slightly reduced accuracy. Considering the real-time interaction as the focus of this system, it is necessary to choose the fastest model in preference. However, for the sake of research rigor, the above two versions of the model are deployed in this paper, and the actual performance metrics are given separately at the end.

2.4. PyAutoGUI-based computer control

PyAutoGUI is a python library for automating operations with cross-platform support for Windows, Mac, Linux and other systems. It is very flexible and can take over control of keyboard, mouse and other input devices, as well as direct control of the view window. This paper needs a simple to use but scalable way of operating to handle the various results that may be obtained from the gesture recognition process. That's why this paper has chosen PyAutoGUI. Furthermore, PyAutoGUI is an open-source library, so it is easy to modify and further apply it to various situations.

When the system is working, the gestures appearing in the camera screen are detected in real time. The result of the detection is converted into a status output, and the computer control program starts to control when it receives the change of status.

In order to control the computer with hand gestures for corresponding operations, this paper designs a rule as shown in Table 1 below.

Table 1. Hand gesture operation command.

Hand gesture	Operation
fist	0. Default status, no operation
up, down, left, right	1. Move the mouse pointer in the corresponding direction (The longer the gesture is held the faster the pointer moves)
fist→palm→fist (in 2 seconds)	2. Left mouse button click (Do the gesture twice in 2 seconds for double click)
fist→palm (Hold for more than 2 seconds)	3. Long press on the left mouse button (Dragging can be done by doing up, down, left, right gestures without returning to the fist)
fist→two up→fist	4. Right mouse button click
fist→ok→fist (in 2 seconds)	5. Minimize current focus window
fist→ok (Hold for more than 2 seconds)	6. Minimize all windows, return to desktop

Note that the system uses fist as the default state. The program only starts to take over computer control when the system accepts fist as the start signal. Any gesture needs to return to the fist state after it is completed. If a meaningless gesture combination is entered, the system will automatically return to the default state and wait for the fist signal.

3. Results & discussion

3.1. Experimentation platform

Windows 10 computer with 8GB video memory rtx2070s graphics card. CUDA version 11.6. The programming environment is python 3.9. The learning model is YOLOv5s and YOLOv5x. Based on the actual training, the average amount of video memory used is about 6.9GB when using the YOLOv5s model.

3.2. Hyper parameters

The training hyper parameters are set as shown in the following Table 2.

Table 2. Hyper parameters and values.

Hyper Parameters	Value
Initial learning rate	0.01
Final OneCycleLR learning rate	0.2
Epochs	300
Optimizer	Stochastic Gradient Descent (SGD)
SGD Momentum	0.937
Batch Size	32

In the model parameter settings, since the gestures to be trained in this paper have different left and right directions, it is necessary to turn off the left and right flipping of the data augmentation.

3.3. Training

This paper uses about 40,000 images for training and 1500 images for verification. Each epoch has about 1300 batches. Due to the performance limitations of the device used, the device can process about 3 batches per second. A complete epoch takes 7 minutes of training time. This is the data for using the YOLOv5s model. If the YOLOv5x model is used for training, the time for one epoch will increase to 50 minutes. It means that the total training time will be up to about 11 days. But YOLOv5s can be trained in less than 2 days. It is obvious that YOLOv5s has a very great advantage in terms of efficiency.

3.4. Performance indicator

In this section this paper gives the performance indicators such as mean average precision (mAP), box_loss, cls_loss and the others after the training of YOLOv5s model.

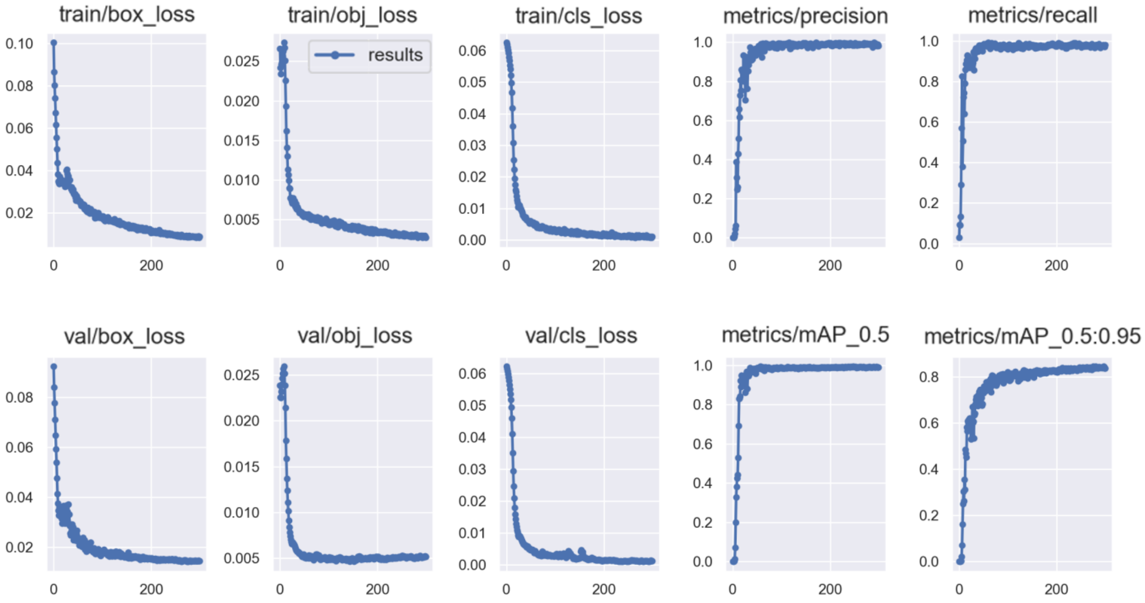


Figure 4. Loss and mAP (YOLOv5s).

According to the above scatter plot (Figure 4), the loss function drops significantly to a better metric during the previous epoch, and the accuracy converges rapidly from nearly 0% at the beginning to over 90%. After reaching a good level, the model is continuously optimized at a smaller rate. Finally, an mAP@0.5 of 0.992 and mAP@0.5:0.95 of 0.838 were achieved. The following Figure 5 is PR curve.

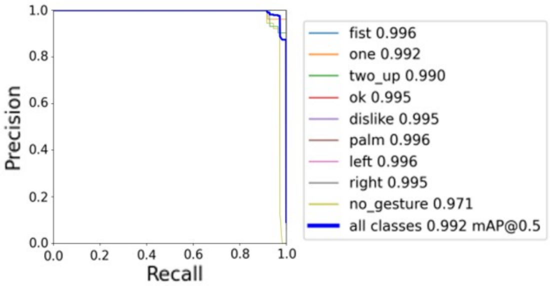


Figure 5. PR Curve (YOLOv5s).

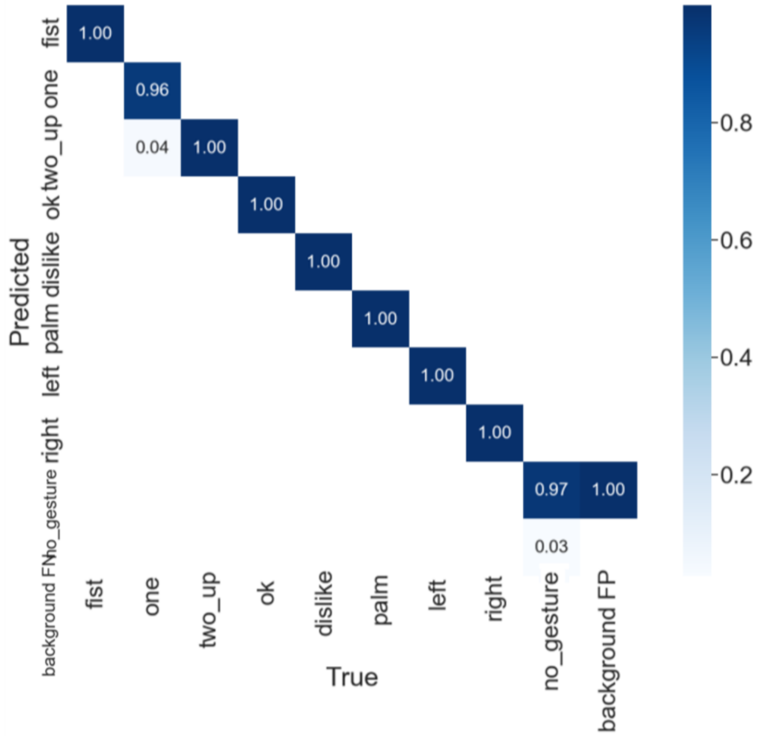


Figure 6. Confusion Matrix (YOLOv5s)s.

From Figure 6, the final trained model has good accuracy and differentiation for each different class of recognition. Most of the hand gesture categories achieved almost 100% correct results in the test. One gesture and Two_up gesture are similar in form, so there is a small probability of incorrect recognition in the test data where some of the feature information is relatively vague. Overall, the performance is still very good.

3.5. Computer control

After obtaining the trained model, the system was tested by 10 testers in this research. The system shows good recognition rate for different positions appearing in any position in the camera under various environments. The recognition speed of the system can reach 50 frames per second (limited by the performance of the test computer), and the response speed meets the use demand of real-time operation. Examples of the testing process are displayed in Figure 7

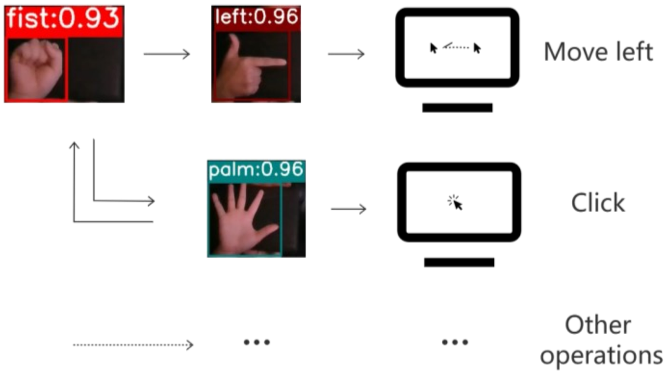


Figure 7. Recognizing hand gestures to control computer.

3.6. Discussion

What can be acknowledged is that YOLOv5s performance is very good. Because of its small model size and fast computing speed, it can complete the training in a short time that would take an extremely long time for traditional machine learning. Even compared to its predecessor, YOLOv3, YOLOv5s shows better accuracy and speed. Thanks to the end-to-end detection method, the complex processing of the data is removed, making real-time monitoring in a high frame rate environment possible. The author also tried testing with a small data set of less than 100 images, and after a very short training time, YOLOv5s also achieved good recognition of brand-new features. This means that the system allows users to enter some photos of the gestures they need to recognize, and then the system can accurately recognize these custom gestures. This has a high practical value.

However, it is a caution that hand gestures with high similarity may lead to confusion in the detection results under some conditions. The use of more different hand gestures would be beneficial to improve the recognition accuracy of this system.

In general, the performance of this system for computer control is quite satisfactory. Attributed to PyAutoGUI's open source, during the study, the modifications to the operation details were easy and reliable, and there were basically no errors reported in the use of PyAutoGUI. Moreover, this function library runs quickly and matches well the real-time gesture recognition output. It is enough to prove that the choice of PyAutoGUI as the interface to the computer controller is very appropriate.

4. Conclusion

In order to achieve more efficient and accurate computer manipulation and data entry with hand gestures, this paper constructs a computer control solution with end-to-end learning-based hand gesture recognition. Firstly, the video images containing the user's hand gestures is captured by the camera. Then after simple pre-processing, the YOLOv5 model based on end-to-end learning is used to analyze and recognize the user's different hand gestures. Finally, this paper designs a series of command correspondence. By calling the PyAutoGUI function library in python, the computer can make corresponding real-time responses to different hand gestures according to the recognition results. This paper uses some ready-made and collected data to make a training set and test set for model training. When testers use the trained system in a variety of simulated real-world usage situations, the system shows good respond speed and accuracy. Basically, it can be said that this system has some practical utility.

In the future, the author plans to apply the system to more situations and make adaptive modifications. The current system can only handle simple operations and can only select operations in a fixed set mapping. The author will try to enrich the interaction of the system so that it can accomplish more complex and detailed manipulation of the computer. For example, set up a virtual keyboard that recognizes the location of gestures for keyboard input, or allow users to develop their own usage specifications. At the same time, the extension of the application device to smart devices such as cell phones is something that can be considered. It should be mentioned that the system still lacks feedback other than visual feedback, such as hearing feedback. For the purpose of accessibility, the system will be developed to include more multi-sensory feedback to help people with disabilities to use electronic devices better.

References

[1] Jagnade, G., Ikar, M., Chaudhari, N., & Chaware, M. Hand Gesture-based Virtual Mouse using Open CV. 2023 International Conference on Intelligent Data Communication Technologies and Internet of Things 820-825.

[2] Gavale, S., & Jadhav, Y. Hand Gesture Detection Using Arduino and Python for Screen Control. 2020 International Journal of Engineering Applied Sciences and Technology, 5, 271-276.

[3] Oudah, M., Al-Naji, A., & Chahl, J. Hand gesture recognition based on computer vision: a review of techniques. 2020 Journal of Imaging, 6(8), 73.

[4] AlSaedi, A. K. H., & AlAsadi, A. H. H. A new hand gestures recognition system. Indonesian Journal of Electrical Engineering and Computer Science, 18(1), 49-55.

[5] Guo, L., Lu, Z., & Yao, L. Human-machine interaction sensing technology based on hand gesture recognition: A review. 2020, IEEE Transactions on Human-Machine Systems, 51(4), 300-309.

[6] Mujahid, A., Awan, M. J., Yasin, A., Mohammed, M. A., Damaševičius, R., Maskeliūnas, R., & Abdulkareem, K. H. Real-time hand gesture recognition based on deep learning YOLOv3 model. 2021 Applied Sciences, 11(9), 4164.

[7] Thuan, D. Evolution of Yolo algorithm and Yolov5: The State-of-the-Art object detention algorithm. 2021Journal of Electrical Engineering and Computer Science,1-10.

[8] Doherty, J., Gardiner, B., Kerr, E., Siddique, N., & Manvi, S. S. Comparative Study of Activation Functions and Their Impact on the YOLOv5 Object Detection Model. 2022, Pattern

[9] Sweigart, A. PyAutoGUI documentation. Read the Docs, 25.

[10] Kapitanov, A., Makhlyarchuk, A., & Kvanchiani, K. HaGRID-HAnd Gesture Recognition Image Dataset.2022 arXiv preprint arXiv:2206.08219.

[11] Ultralytics. YOLOv5 (Version 7.0). https://github.com/ultralytics/yolov5

[12] Sozzi, M., Cantalamessa, S., Cogato, A., Kayad, A., & Marinello, F. Automatic bunch detection in white grape varieties using YOLOv3, YOLOv4, and YOLOv5 deep learning algorithms. 2022 Agronomy, 12(2), 319.

[13] Liu, K., Tang, H., He, S., Yu, Q., Xiong, Y., & Wang, N. Performance validation of YOLO variants for object detection. 2021 International Conference on bioinformatics and intelligent computing 239-243.

Deep learning methods used in movie recommendation systems

Lexi Liu

Chengdu Foreign Languages School, Xipu Street, Pidu District, Chengdu City, Sichuan Province, China

1502806106@qq.com

Abstract. As the amount of internet movie data grows rapidly, traditional movie recommendation systems face increasing challenges. They typically rely on statistical algorithms such as item-based or user-based collaborative filtering. However, these algorithms struggle to handle large-scale data and often fail to capture the complexity and contextual information of user behavior. Therefore, deep learning techniques have been widely applied to movie recommendation systems. This paper reviews movie recommendation algorithms based on traditional statistical models and introduces three main deep learning techniques: Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN). ANN can extract features at different levels of users and movies; CNN can capture features of movie posters and movie data to recommend similar movies; RNN can consider user historical behavior and contextual information to better understand user interests and demands. The application of these deep learning techniques can enhance the accuracy and user experience of movie recommendation systems. This paper also demonstrates the advantages and disadvantages of these models and their specific application methods in movie recommendation systems, and points out the direction for further development and improvement of deep learning models in this field.

Keywords: recommendation system, deep learning, artificial neural network, CNN, RNN.

1. Introduction

Machine learning plays an extremely important role in movie recommendation systems [1]. It can help improve the accuracy and personalization of recommendations. With the development of the internet and the digital entertainment industry, people are increasingly inclined to watch movies at home. However, it is not easy to find content that one likes from millions of movies and TV programs. This is why movie recommendation systems have become so important. Machine learning can learn data features from a large amount of user data such as viewing history and ratings, and use this as the basis for training models to better predict user preferences and recommend movies and TV programs that users may like. Machine learning can also help recommendation systems solve the cold start problem, where new users joining the system may not have enough personal preference data for accurate recommendations. In this case, machine learning can use some data such as age, gender, and geographic location to infer new users' preferences. Through machine learning, recommendation systems can recommend according to each user's specific needs, making recommendations more accurate, personalized, and targeted. Therefore, the application of machine learning in movie recommendation systems is of great significance. It can help recommendation systems better serve users, improve user experience, and also promote the development of the digital entertainment industry [2].

Traditional machine learning was initially used in recommendation systems to predict user preferences based on pre-defined features and corresponding relationships. For movie recommendation systems, this involved analyzing user preferences and movie characteristics to recommend movies. However, traditional machine learning methods have limitations in effectively learning from growing massive internet data, resulting in limited accuracy of recommendation systems. The emergence of deep learning technology as a powerful tool for recommendation systems can be attributed to advancements in graphics cards and parallel computing. With its powerful parameter space, deep learning can effectively learn from massive internet data, allowing for the discovery of hidden features and complex relationships between data without the need for manual feature extraction [3-6]. Compared with traditional machine learning algorithms, deep learning can improve the accuracy of recommendation systems and reduce the time and effort required for manual intervention, ultimately enhancing personalization and real-time performance. In this article, we explore the application of different deep learning models for various types of movie data on the internet, and provide a summary of their advantages and disadvantages. Our analysis provides guidance for the future construction of movie recommendation systems, emphasizing the importance of deep learning in leveraging large amounts of data to better serve users and promote the development of the digital entertainment industry.

2. Traditional machine learning methods

Traditional machine learning algorithms have been widely used in early movie recommendation systems. These methods analyze the relationship between different users and movies by building statistical models, extracting sensitive features of different user groups for different types of movies, in order to better meet the recommendation needs of users.

In our research on traditional machine learning, the focus was mainly on comparing the similarity of different groups of data to achieve recommendation through machine learning methods. Here we introduce two key recommendation algorithms.

The first method is to use similarity calculation to classify different users or movies to achieve movie recommendations. This method can be divided into two aspects. In the case of a movie recommendation system, they are based on user-based collaborative filtering and item-based collaborative filtering. Both of them mainly require the user's rating data for different movies as input features for the recommendation system. The essence of user-based collaborative filtering is to identify a group of users who have similar preferences to the target user based on existing ratings. By analyzing the evaluation of this group for a given movie, the system can decide whether to recommend the movie to the target user. In item-based collaborative filtering, the system first finds the movies rated highly by the target user, compares their ratings with other movies, and then recommends similar movies to the target user based on their similarity. These similarity-based methods have high flexibility and can use different correlation metrics, such as Pearson correlation coefficient and Spearman correlation coefficient [7-9].

$$r = \frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i-\bar{X})^2(Y_i-\bar{Y})^2}}$$

$$\rho = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n^3-n}$$

The second method involves using the K-Nearest Neighbor (KNN) algorithm to classify users or movies into categories [10,12]. The KNN algorithm is the simplest non-parametric supervised classification algorithm. In the KNN algorithm, we need to calculate the distance between the object to be classified and its neighboring nodes based on its features (such as Euclidean distance, Chebyshev distance, etc.), and then select the majority category among its K neighbors as the target object's category. The advantage of this method over the first method is that when recommending movies, we only need to consider the category of users or movies without comparing the similarity of individuals one by one, which improves the efficiency of the recommendation system. At the same time, this algorithm does not require parameters, which means that the model does not make any assumptions about the data, reducing the impact of special cases on the recommendation system. However, the KNN algorithm faces problems such as high computational complexity and high spatial complexity. At the same time, the imbalance of samples can also affect the effectiveness of this method.

3. Deep learning

Against the backdrop of the huge amount of data in the Internet today, traditional machine learning algorithms have shown their limitations. On the one hand, a large amount of data does not significantly improve the accuracy of traditional machine learning. On the other hand, traditional machine learning algorithms cannot fully explore the rules behind the data. In the research process, the continuous development of neural network direction has led to the concept of deep learning, which can deeply analyze the inherent features and rules of the samples, and its performance is positively correlated with the size of the data. That is to say, under the background of big data, the performance of deep learning will be better.

4. Artificial neural network

Artificial neural networks (ANNs) are information processing systems composed of a large number of interconnected neurons, analogous to the neural systems that transmit information through synapses in biology. ANNs typically consist of an input layer, hidden layers, and an output layer. In general, the more complex the problem and the more variables involved, the more layers and neurons in the hidden layers are required [13,14]. When we input data into an ANN system, the data is processed by a parameter matrix in the hidden layer, and the signal transmission between neurons is simulated through a nonlinear activation function [14]. With numerous model parameters, ANNs can learn the hidden patterns in the data, greatly improving the accuracy of recommendation systems. To evaluate the performance of an ANN system, we typically use error functions such as RMSE. The training process involves adjusting the parameters to minimize the error function, which can be achieved through gradient descent. The specific process of gradient descent involves using the backpropagation algorithm to obtain the gradient of the error function, and then continuously adjusting each parameter in the opposite direction of the gradient until the error function converges. However, due to the large number of parameters required by ANNs, the high cost of adjusting parameters is inevitable, which limits the application scenarios of ANNs.

5. Convolutional neural network

Convolutional Neural Network (CNN) is a deep learning model mainly used for processing image data [15]. In a movie recommendation system, we can use CNN to extract features from movie screenshots, posters, and other image data to help the system obtain more information about movies. During the process of using CNN to process input information, the information goes through a series of layers including convolutional layer, ReLU layer, pooling layer, and fully connected neural network, and eventually outputs a result [16]. In the convolutional layer, a convolutional kernel is used as a set of weighted elements to perform a weighted sum with elements in the input information. The role of the convolutional layer is to represent local features of the input information with more concise and distinctive numbers. The larger the number obtained through the convolution process, the more correlation there is between the local feature and the given template. Using convolutional kernels to process information greatly reduces the number of parameters and computational complexity compared to fully connected artificial neural networks, which improves efficiency. Additionally, CNNs largely reduce data volume and preserve spatial information in images by utilizing convolutional and pooling layers, avoiding the limitation of one-dimensional representation of all information. This makes CNNs perform well in image classification and other tasks. However, although CNNs reduce the number of parameters through sparse connections, they still have high computational requirements and require a large amount of training data to continuously adjust parameters, which reflects the high operating costs of CNNs.

6. Recurrent neural network

Movie ratings are an important source of information for recommending movies to users, and an objective and accurate rating can correct many misleading movie information. Usually, movie ratings are presented in the form of text, and our information has a chronological relationship that needs to be combined and read together to be properly understood. Therefore, people have proposed recurrent neural networks (RNNs) to connect and process the chronological information [17-20]. RNNs can predict user preferences based on their historical behavior (such as which movies they have watched before and their ratings), thus recommending movies to them. In addition, RNNs can model sequence data over time, so they can sort recommendations based on time and provide users with time-based recommendations. Recurrent neural networks process input with a chronological relationship in sequence. In each step, the information retained from the previous step is passed to the current moment. At the same time, another part of the information comes from the current input, and the output information is passed to the next moment. However, this processing method has the disadvantage of severe data loss when facing a large amount of input data. Because each input data can only be passed on to the next loop in a certain proportion, the initial input data may decay to a very small proportion during the processing. However, even so, it still requires a very large amount of training data to adjust the parameters, resulting in high training costs.

7. Discussion

With the advancement of deep learning technology, movie recommendation systems can now effectively utilize big data on the internet with minimal human intervention. Deep learning algorithms can autonomously analyze and utilize larger and more diverse data, while also extracting data features and correlations that are not easily observed by humans. This results in significantly improved utilization rate and recommendation accuracy of data. However, while the advantages of deep learning are evident, there are still certain shortcomings that need to be addressed.

First of all, the aforementioned deep learning models still have certain shortcomings. For example, in the application of convolutional neural networks, the original information undergoes highly abstract processing through multiple layers of CNN, which leads to information loss during the processing. To address this issue, scientists have proposed residual structures, which preserve some of the previous data during each step of processing, reducing data loss and avoiding the problem of gradient vanishing [5]. In the case of recurrent neural networks, severe data loss is also a barrier to their development, which led to the emergence of LSTM (Long Short-Term Memory) and GRU (Gated Recurrent Unit) [17-20]. LSTM demonstrates great advantages in processing long-term memory, but has the limitation of extremely high computational complexity; GRU is a simplified form of LSTM, with reduced computational complexity and can be considered as a compromise between the original RNN and LSTM. The recent popular ChatGPT model is based on the Transformer model, which can better adapt to information processing in the context of big data and can be widely applied to various scenarios, but still has the drawback of high computational complexity and training cost, as is common with deep learning models [21].

Secondly, data imbalance can lead to a decrease in the accuracy of the recommendation model. In the era of the Internet, data is inevitably biased due to the preferences of the user group, and it cannot guarantee an equal relationship in terms of data volume for each category. In the movie recommendation system, this will lead to insufficient data for specific categories of movies, and users with niche preferences may not get ideal recommendation results. To address this issue, we can supplement data through other means to avoid learning biases caused by data. We can also perform data augmentation on existing data, such as rotating and adding noise to image information, to increase the volume of data.

Data augmentation operations need to consider the application scenario of the model, that is, whether the accuracy requirements are strict [22]. In addition to processing data, we can also use transfer learning to strengthen the connection between multiple tasks. For example, when recommending movies to users with niche preferences, we can use the parameters previously used to recommend similar users as the basis for improving the model, and then make adjustments to reduce the error caused by insufficient data to a certain extent [23]. In addition, Generative Adversarial Networks (GAN) can also be used to mitigate the obstacles caused by data imbalance. In the GAN model, there are mainly two parts: the generator and the discriminator. The generator generates data and the discriminator judges whether the data is true or false, and affects the judgment to have stronger discrimination ability. The results of discrimination can in turn affect the parameters of the generator to generate more realistic data. The generator and discriminator are trained in turn, making the generator have a strong ability to create data. At this time, the data created by the generator can be used to solve the problem of data imbalance [24].

Finally, not all recommendation system models are better with increased complexity. Deep learning faces limitations due to the huge computational requirements, and the value of traditional machine learning cannot be denied even in cases with small data and clear features, despite the superiority of deep learning.

References

[1] Marappan, R. & Bhaskaran, S. Movie Recommendation System Modeling Using Machine Learning. Int. J. Math. Eng. Biol. Appl. Comput. 12–16 (2022).

[2] Goyani, M. & Chaurasiya, N. A Review of Movie Recommendation System: Limitations, Survey and Challenges. ELCVIA Electron. Lett. Comput. Vis. Image Anal. 19, 18–37 (2020).

[3] Deng, L. & Yu, D. Deep Learning: Methods and Applications. Found. Trends® Signal Process. 7, 197–387 (2014).

[4] Kamilaris, A. & Prenafeta-Boldú, F. X. Deep learning in agriculture: A survey. Comput. Electron. Agric. 147, 70–90 (2018).

[5] He, K., Zhang, X., Ren, S. & Sun, J. Deep Residual Learning for Image Recognition. (2015).

[6] LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. Nature 521, 436–444 (2015).

[7] Myers, L. & Sirois, M. J. Spearman Correlation Coefficients, Differences between. in Encyclopedia of Statistical Sciences (John Wiley & Sons, Ltd, 2006). doi:10.1002/0471667196.ess5050.pub2.

[8] Benesty, J., Chen, J. & Huang, Y. On the Importance of the Pearson Correlation Coefficient in Noise Reduction. IEEE Trans. Audio Speech Lang. Process. 16, 757–765 (2008).

[9] Sedgwick, P. Pearson's correlation coefficient. BMJ 345, e4483 (2012).

[10] Ahuja, R., Solanki, A. & Nayyar, A. Movie Recommender System Using K-Means Clustering AND K-Nearest Neighbor. in 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence) 263–268 (2019). doi:10.1109/CONFLUENCE.2019.8776969.

[11] Ahmed, M., Seraj, R. & Islam, S. M. S. The k-means Algorithm: A Comprehensive Survey and Performance Evaluation. Electronics 9, 1295 (2020).

[12] Guo, G., Wang, H., Bell, D., Bi, Y. & Greer, K. KNN Model-Based Approach in Classification. in On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE (eds. Meersman, R., Tari, Z. & Schmidt, D. C.) 986–996 (Springer, 2003). doi:10.1007/978-3-540-39964-3_62.

[13] Jain, A. K., Mao, J. & Mohiuddin, K. M. Artificial neural networks: a tutorial. Computer 29, 31–44 (1996).

[14] Krogh, A. What are artificial neural networks? Nat. Biotechnol. 26, 195–197 (2008).

[15] Li, Z., Liu, F., Yang, W., Peng, S. & Zhou, J. A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. IEEE Trans. Neural Netw. Learn. Syst. 33, 6999–7019 (2022).

[16] Gu, J. et al. Recent advances in convolutional neural networks. Pattern Recognit. 77, 354–377

(2018).

[17] Staudemeyer, R. C. & Morris, E. R. Understanding LSTM -- a tutorial into Long Short-Term Memory Recurrent Neural Networks. Preprint at https://doi.org/10.48550/arXiv.1909.09586 (2019).

[18] Smagulova, K. & James, A. P. A survey on LSTM memristive neural network architectures and applications. Eur. Phys. J. Spec. Top. 228, 2313–2324 (2019).

[19] Chung, J., Gulcehre, C., Cho, K. & Bengio, Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. Preprint at https://doi.org/10.48550/arXiv.1412.3555 (2014).

[20] Zhao, R. et al. Machine Health Monitoring Using Local Feature-Based Gated Recurrent Unit Networks. IEEE Trans. Ind. Electron. 65, 1539–1548 (2018).

[21] Vaswani, A. et al. Attention Is All You Need. (2017) doi:10.48550/ARXIV.1706.03762.

[22] Chlap, P. et al. A review of medical image data augmentation techniques for deep learning applications. J. Med. Imaging Radiat. Oncol. 65, 545–563 (2021).

[23] A survey of transfer learning | Journal of Big Data | Full Text. https://journalofbigdata.springeropen.com/articles/10.1186/s40537-016-0043-6.

[24] Creswell, A. et al. Generative Adversarial Networks: An Overview. IEEE Signal Process. Mag. 35, 53–65 (2018).

A review of AI-based game NPCs research

Guanwei Zeng

College of Optical Mechanical and Electrical Engineering, Zhejiang A&F University, Hangzhou, Zhejiang 311300, China

zenggw2022@163.com

Abstract. The application of artificial intelligence has increasingly penetrated into the field of game development. Among them, the non-player characters in the game, namely NPC, are part of the applications of AI. The virtual characters in the game characters are collectively called NPC, which enhances the fidelity and complexity of the game by having contact with with the players. Using AI can make NPCs in the game more vivid, thereby increasing the playability of the game and creating more possibilities. This article will review the research and application of AI-based game NPCs in recent years.

Keywords: AI, NPC, limitations, neural networks, deep reinforcement learning.

1. Introduction

In contemporary games, the importance of NPCs (non-player role) are almost indispensable. NPCs can be any form of virtual characters, from businessmen and teammates to enemies and mission targets, playing a wide variety of roles, interacting with players, enhancing the fidelity and complexity of the game. The appearance, behavior and dialogue of NPCs are programmed by game developers, which also require a lot of time, manpower and financial resources. With the improvement of AI (artificial intelligence) technology, people's requirements for games have gradually increased, and the requirements for game AI are also increasing. This also makes AI systems more complex and diverse in the game [1]. As a result, more and more game developers began to explore the use of AI to create NPCs. This method can not only save developers' time, but also AI can make NPCs achieve more functions, make the game richer and more playable. However, after a literature search, no one has reviewed the current situation in this area in recent years. Therefore, this paper aims to review the related research and application of using AI to create game NPCs in recent years, and sort out the current research status.

2. Historical development

With the gradual growth of the game industry, AI technology has gradually developed and has made great strides in game development. Zhao Lexuan proposed that AI directly and significantly reduced the threshold of the game, and players could play the game more easily [2]. Moreover, in many kinds of games, there will be various NPCs. In the course of development, two methods have been the most popular in the past, the first is the finite state machine (FSM), the second is the behavior tree.

Shi Boxuan mentioned that the state machine appeared in about the 1970s, when AI controlled hostile NPCs [3]. After that, many game manufacturers began to pay attention to the application and quality of AI in games. For example, arcade game 'Qwak', 'Speed Race', and PC-side game 'Star Trek'.

NPCs in these games are controlled by AI, but they no longer simply move, but are unpredictable because they process more information. After that, developers added more changes to NPCs, such as 'PAC-MAN', and the AI strategy in this game became more complex. Then came 'Karate Champ', a role-playing game that featured the first AI combat character in game history. The behavior tree model was first proposed by Next-Gen AI in the game field. In 2001, they also published some key ideas about behavior trees [4]. Under the method of behavior tree, many popular games have been created, such as 'Red Dead Redemption'.

3. Research statues

3.1. A review of the limitations of AI game NPCs

With the rapid development of game engines, game companies are able to create many beautiful pictures. Therefore, the playability and innovation of the game itself has also become the key to standing out from many games [5]. The NPCs in the game directly or indirectly determine the fun and playability of the game [6]. Because of this, many game developers already have many theories and methods to create AI that meets the needs of players. But contrary to expectations, Jeff suggested that even if you went to some game developer conferences with many powerful companies at the time, using a finite state machine was still a game solution that most people could give [7]. This statement has also been confirmed by domestic companies. Shi shared his own experience, in 2016, he attended the Unity developer conference, and communicated with the personnel of NetEase, Tencent and other companies. The solution still coincided with each other and still used state machines or behavior trees [3]. This practice was broken only in 2018. At that time, a rare AI component appeared in the test version, which had the function of machine learning.

K Tupe et al. also pointed out the limitations of many in-game NPCs[8]. NPCs act in limited scope and act in limited ways. These limitations still exist in a large number of games today. As a result, players can often quickly determine where NPCs' limitations are. In order to pursue better and more credible NPCs, the article mentioned, including but not limited to optimizing offline automation behavior technology, adjusting actions and learning better countermeasures online automation technology, real-time control mechanism and so on. This paper highlights the shortcomings and optimization directions of NPCs, and also mentions the relatively basic principle of AI and the prospect of the future. The limitations are worth thinking about. As for how to realize these in detail, it also needs further research, which is not mentioned in this paper.

Mao Xiangyu proposed two methods for the implementation of NPC intelligent behavior based on Uinyt3D, which are still finite state machines and behavior trees [9]. Mao only summarized these two methods in the paper, and did not propose implementation schemes and specific experiments, and the limitations of such NPCs should still be within the scope of K Tupe [8]. It can be seen that at that time in the same year, there was still no direction for this in China. Shi [3] conducted research on behavior trees and finite state machines, and found that in the game industry at home and abroad, neither method can cope with the AI control of multiple character NPCs. Among them, the advantage of the state machine is that it performs a very fast process, simple operation and high flexibility. But admittedly, it scales poorly. Because of this drawback, many game AI behaviors are restricted, especially detailed movements. As more states increase, the number of transitions between states becomes difficult to control. At the same time, different NPCs have their own different state machines, and then add scenes, difficulty and other aspects in this situation, it is estimated that such a game will need hundreds of state machines, and the development and maintenance work brought by this, as well as the corresponding costs, are completely unrealistic. However, although behavior trees are highly reusable, the decision-making speed and readability of this method rapidly decreases as complexity increases. In summary, behavior trees also encounter the same bottlenecks as state machines, slowing down game development. This summary clearly explains the limitations of in-game NPCs using two common methods.

But for so long, the reasons why most game makers still choose to use finite state machines and behavior trees remain to be investigated. First of all, machine learning requires fault tolerance, but in most games, there is basically no fault tolerance and it is very prone to failure. Second, the process of machine learning is not easily manipulated, and there are many uncertainties. Third, AI training and learning require high training costs [10]. Finally, the game update will most likely lead to the loss of the training results, so such an AI can not give a definitive result at all, which is very detrimental to the development of the game, and requires more manpower, which also cannot bring more economic benefits [11].

3.2. A review of breakthrough research based on AI game NPC

The method proposed by Shi Yuan is different from the above two. Based on the reinforcement learning theory, the proximal strategy algorithm is optimized and designed, and the non-player character AI system is designed based on the hierarchical architecture, which is divided into perception layer, strategy layer, request layer and behavior layer [12]. This method reduces the coupling of system code, increases the flexibility of the system, and facilitates later maintenance. After designing and testing, Shi found that the system has significantly improved in scalability and development difficulty in a Rogue-lite game designed by himself. All in all, the optimized algorithm is used instead of the behavior tree, which enhances the action force of NPCs and increases the playability of the game. However, the study only targets NPCs in combat behavior, which has certain limitations and is currently only suitable for small game groups.

In addition, Shi also proposed a game AI design method for multi-NPCs and applied it to NPCs in competitive mobile game(Sohu Changyou "Agent Squad") [3]. In his design structure, the dimension and behavior abstraction layer is used to abstract the behavior of NPCs and the factors that can determine the behavior, and then design the appropriate model for each dimension, configure the file, parse and classify the file, then transfer the relevant data to the computational model, and finally select the most expected execution in the behavior. Through testing, this model can meet the needs of multi-NPC game AI, and the performance and consumption cost are also within a reasonable range. Compared with the previous two initial design methods, this construction not only shows high scalability, but also improves the efficiency of development and reduces maintenance and debugging costs. Nevertheless, there are still some functional deficiencies in this structure, the workload is preserved, it needs to be optimized, and some limitations are still present compared to the study of stone [12]. But it is not difficult to see that the current game has many algorithm optimizations for competitive AI hostile NPCs, which makes it possible for these NPCs to challenge humans.

Compared with the above proposed methods, Tang Zhentao et al. described the problems of real-time fighting games and elaborated a series of methods to achieve the combination of AI and fighting games [14]. They are heuristic rule type, statistical forward planning type (including Monte Carlo tree search algorithm and rolling time domain calculus method) and deep reinforcement learning type. Compared with the consistent heuristic rule-based strategy search method, the statistical forward planning method and the deep reinforcement learning method have good environmental adaptability and strategy model optimization. However, its generalization and efficiency need to be enhanced. If it is possible to combine several of them, such as statistical forward planning and deep reinforcement learning, it may have more potential.

3.3. A review of research based on the perspective of game developers

The functions of artificial intelligence systems in shooting games summarized by Yu Kechun are divided into three modules, among which they can sense enemies, assign teammates and initiate decision-making, which are very detailed and comprehensive [13]. It can be seen that in terms of combat, the functionality of artificial intelligence is very strong. Based on some game cheats that use code to make the character strong, AI can also do it, especially in competitive games such as shooting and implementing strategies. However, too strong enemies will frustrate players, resulting in bad gaming experience. So making AI make mistakes appropriately in games, or simulating different levels of human technology, may enrich the game.

In addition to these types of NPCs, the behavior of some neutral or friendly NPCs will be more technical, especially when dialogue is required. At this time, the single way of dialogue has considerable fixation. But there are actually few related studies, because of its complexity, and the reality of combining with games, are not easy to achieve. Zhu Peng mentioned in the article that it will take a long time for the breakthrough in algorithms, about 2 to 5 years, AI will present a pyramid structure [15]. The existing OpenAI and ChatGPT models can already provide a natural and comfortable continuous conversation experience, as well as the ability to write personal documents, and so on. So whether NPCs in the game can also use this kind of dialogue AI to integrate or not, perhaps in the future, it will be a way to make NPCs more vivid and more playable.

Throughout the current research status, a single computational intelligence method has its own unique advantages, but there are also corresponding shortcomings. How to organically integrate the existing intelligent methods to form complementary advantages is a direction of current game AI. In addition, in the case of more comprehensive development of hostile AI, how to make players more experiential and playful is also a problem. When AI is neutral or friendly, making dialogue and decision-making more flexible and humane is also a trend to make games richer and more intelligent. Finally, how to solve the problems caused by the application of AI in the game is also one of the difficulties that need to be broken.

4. Conclusion

This paper reviews the research and application of using AI to create game NPCs. Although AI faces some challenges in creating NPCs, it is still a very promising field in game development. With the development of AI and game industry, it is believed that this will get more attention and research. In the future, more intelligent and more playable NPC designs are expected to enrich the game experience of players.

References

[1] Bao Fang 2014 Research and Implementation of Intelligent Decision-making System in Simulated Business Games (Hangzhou: Dianzi University)

[2] Zhao Lexuan 2021 What Does AI Bring to Video Games? (Wenzhou: People's Post and Telecommunications)

[3] Shi Boxuan 2019 Research on a Design Method for Multi-NPC in the Development of Electronic Game AI System (Beijing: Beijing University of Chemical Technology)

[4] Xu Qiang 2015 The Design and Implementation of Game Role Control System Based on Behavior Tree (Harbin: Harbin Institute of Technology)

[5] He Wenjun 2018 Game AI Design and Implementation Based on Behavior Tree (Chengdu: Chengdu University of Technology)

[6] Chen Yunlin 2017 Design and Implementation of Basketball Game AI System Based on Unity3D Engine (Nanjing: Nanjing University)

[7] Orkin J 2006 Three States and a Plan: The A.I. of F.E.A.R.

[8] Tupe K and Singh P et al 2016 AI & NPC in Games

[9] Mao Xiangyu 2016 Intelligent Behavior Analysis of Non-human Player (NPC) Based on Unity3D (Wuhan: Digital Technology and Application) p 11

[10] Li Kun and Li Ping et al 2018 Design and Implementation of Artificial Intelligence for MOBA Games (Changsha: Computer and Information Technology) pp 8-11

[11] Liu Xiaowei and GaoChunming 2016 Combining Behavior Tree and Q-learning to Optimize Agent Behavior Decision in UT2004 (Changsha: Computer Engineering and Application) pp 113-118

[12] Shi Yuan 2019 Design and Implementation of Non-player Role AI System Based on Reinforcement Learning (Shanghai: Donghua University)

[13] Yu Kechun 2022 Research and Analysis of Computer Game Software Development Technology Based on Artificial Intelligence (Huizhou: Software) pp 39-41

[14] Tang Zhentao and Liang Rongqin et al 2022 Intelligent Decision-making Methods for Real-time Fighting Games (Beijing: Control theory and applications)

[15] Zhu Peng 2023 It Will Take 2 to 5 Years for the Algorithm to Break Through (Chengdu: Daily Economic News)

Proceedings of the 5th International Conference on Computing and Data Science
DOI: 10.54254/2755-2721/15/20230828

A comparison of feature extraction methods in image stitching

Junxin Zheng

School of Computer and Information Engineering, Shanghai Polytechnic University,
2360 Jin Hai Road, Pudong District, Shanghai 201209, China

20201112625@stu.sspu.edu.cn

Abstract. The usage of feature detectors for image stitching has become a popular research area in computer vision. Various feature extraction algorithms can be used in the process of image stitching process, but they perform varyingly when handling different images and no single algorithm could outperform all others. This paper focuses on the comparison of feature extraction algorithms used for panoramic image stitching. The research utilizes the SIFT, ORB, AKAZE, and BRISK to conduct feature points and match feature points on a group of image sets. The RANSAC algorithm is then used to filter out the outliers and calculate the homograph matrix. Completes the panoramic with image splicing and smoothing through the matrix transformation. Derived from the comparison of the matching and stitching results, the AKAZE detector is found to be the fastest feature point detection and extraction algorithm, while the SIFT detector will provide more feature points to make more accurate matches possible. These findings have implications for the development of efficient and effective computer vision technologies for various applications.

Keywords: feature extraction, computer vision, image stitching, panorama images.

1. Introduction

As computer graphics and technology for image merging have advanced, the demands for computer graphics detection have become increasingly complex and diverse, and greater precision and depth in computer image feature detection is a significant fundament of numerous image processing procedures, such as image stitching, camera calibration, dense reconstruction, scene understanding, as well as face recognition and identity verification for security. For instance, panoramic splicing, which involves stitching several continuous photos together, along with filtering and matching feature points, has become a popular solution for creating complete, detailed, high-quality wide-angle 360-degree panoramas [1]. With inaccurately matched point pairs, the estimated warp between adjacent images can be tremendously different from the actual transformation, which will lead to screwed and unusable result images.

Current techniques and methods for image stitching remain imperfect, which means ample room for growth, as it combines a range of fields, such as optics, computer vision, and computer graphics. The rapid advancement of this technology has the potential to facilitate interdisciplinary collaboration and provide significant technological support for other fields, including virtual reality, object recognition, and aiding devices for individuals with vision impairments. By improving the accuracy and efficiency of feature detection, computer systems can analyze and understand input images more precisely and thoroughly.

This paper aims to apply different feature detection methods on the same image resource to compare images for similarities using point features as the main determining criteria. The purpose of this experiment is to evaluate the performance of different feature detection methods in the stitching process.

2. Methods

In this chapter, the algorithms used in the experiment will be introduced, as also the platform and evaluation procedure.

2.1. Feature points extracting methods

In the field of computer vision and image processing, a feature point (also known as a keypoint or interest point) refers to a location in an image that possesses visually distinctive and recognizable local features, such as edges, corners, or unique texture patterns [2]. Two basic requirements for image feature points are difference and repeatability, meaning that differences should be recognizable from visually salient points and the same feature is repeatable and matchable from different perspectives [3].

The selection of a suitable corner/edge detection algorithm typically relies on various factors such as the particular application and the balance between speed and precision required for the task at hand.

The detecting algorithms used in this experiment include Oriented Features from Accelerated Segment Test and Rotated Binary Robust Independent Elementary Features (ORB), Scale-Invariant Feature Transform (SIFT), Binary Robust Invariant Scalable Keypoints (BRISK), and Accelerated-KAZE (AKAZE).

2.1.1. ORB. ORB, or Oriented Features from Accelerated Segment Test (FAST) and Rotated Binary Robust Independent Elementary Features (BRIEF), as the name suggests, is a computer vision algorithm that combines two others widely used algorithms, FAST corner detector and BRIEF descriptor. ORB operates by first using FAST to detect corners in an image, and then computing a brief binary descriptor for each detected key point using BRIEF. However, ORB improves on the original BRIEF algorithm by making its descriptors rotation-invariant. This is achieved by calculating the orientation of each key point using a modified version of FAST, and then rotating the descriptor to align with this orientation [4].

ORB has several advantages over other feature detection and description algorithms. One of its strengths is its computational efficiency, which is superior to other popular algorithms like SIFT and Speeded-Up Robust Features (SURF). Additionally, ORB is robust to changes in scale and rotation, making it well-suited for tasks like object recognition and tracking.

2.1.2. SIFT. SIFT, or Scale-Invariant Feature Transform, is one of the most widely used algorithms for feature detection and description in computer vision. To detect feature points, SIFT will first identify scale-space extrema in an image where the Difference of Gaussian (DoG) function reaches a maximum or minimum. These extrema are filtered to keep only those that are stable under different image transformations, such as scale and rotation while eliminating low-contrast points and edge responses [5].

After identifying key points, SIFT generates a descriptor for each key point that is invariant to scale and rotation. The descriptor is a vector of floating-point values that encodes the image gradient orientations and magnitudes in a local neighbourhood around the key point. SIFT is known for its robustness to scale changes and rotation, making it suitable for tasks such as object recognition and tracking. However, one disadvantage of SIFT is its computational complexity, which can make it slow to compute on large images or in real-time applications [6].

2.1.3. BRISK. BRISK (Binary Robust Invariant Scalable Keypoints) is a keypoints/features detection and description algorithm, it has been optimized for being fast and robust for various use cases. When BRISK processes the images, it detects keypoints first and then computes binary descriptors that are resilient to brightness and viewpoint variations, efficiently matched using Hamming distance. In order to find scale-invariant keypoints and generate scale-invariant descriptors that can withstand changes in the size of local picture patches around each keypoint, the algorithm employs a scale-space pyramid for detecting keypoints at multiple scales.

By dividing the local image patch surrounding each keypoint into 4 x 4 square areas and computing binary values for each area based on local image gradient orientations, BRISK computes descriptors using a pattern-based methodology. The final descriptor for each keypoint is created by joining the binary codes together. BRISK has proven to be highly effective on several computer vision tasks, including 3D reconstruction, picture matching, and object identification. Nevertheless, it might not perform as well as more recent algorithms, such as AKAZE, which are created especially to tackle complicated image conditions [7].

2.1.4. AKAZE. AKAZE (Accelerated-KAZE) is a feature detection and description algorithm, designed to operate robustly under various complexities of input images, which is an essential requirement in computer vision applications. AKAZE improves on the KAZE algorithm by effectively detecting features using an accelerated version of nonlinear scale space and by proposing a unique descriptor with insensitivity to noise and blur. The nonlinear scale space method of extracting multiscale-oriented patches around each keypoint is used to calculate the AKAZE descriptor, which is then normalized to account for blur and variations in lightning conditions [8].

Eventually, a special feature vector based on the gradient orientation patterns inside the patch is generated, producing a very distinctive and reliable descriptor.

2.2. Feature points matching

Brute-force matches the descriptors with hamming metrics with 2 descriptors obtained in the previous step using the match_descriptors function provided by scikit-image package. A pair of feature points is considered matched when the distance between their descriptors is below a certain threshold [9].

However, some matched feature points could be mismatched, and it is necessary to eliminate them to ensure the accuracy of the final result. To address this issue, the Random sample consensus (RANSAC) algorithm is often employed, it can effectively reduce the impact of noise in the data by identifying and excluding outliers (Table 1).

Table 1. RANSAC procedure.

RANSAC Procedure
1) Select a subset of n data points randomly from all the keypoints.
2) A transformation matrix is estimated using selected data points.
3) Evaluate remaining data by their distance from the model.
4) Remove the points having a distance exceeding the threshold.

2.3. Image stitching

Apply the homography transformation matrix to all 4 corners of one of the images, to see if the top left corner has a negative **x** value, indicating whether that image is on the left side of the resulting image [10]. Wrap that image with translation and homography transformation according to its relative position to the other image, resize the unmodified image, and stitch the warped image into that.

2.4. Image blending and cropping

2 smoothing windows with a linear decreasing slope from 1-0 and an increasing slope from 0-1 are applied to the borders between the 2 images for merging them. Crop the resulting image along straight lines.

3. Experiment

3.1. Experiment platform and evaluation procedure

The code interpreter used in the experiment is Python 3.9.0, used libraries include OpenCV 4.7.0, pandas1.5.3, scikit-image 0.19.3, numpy 1.23.5, and matplotlib 3.6.2.

Calculating the average time consumption and match rate of 10 batches of processing with each algorithm on both image sets. Evaluate the performance of the algorithms by their accuracy and efficiency.

3.2. Source images

The data source for the experiment comprises a collection of image sets of various live-action scenes. The images contained in Figure 1 are used for the experiment.

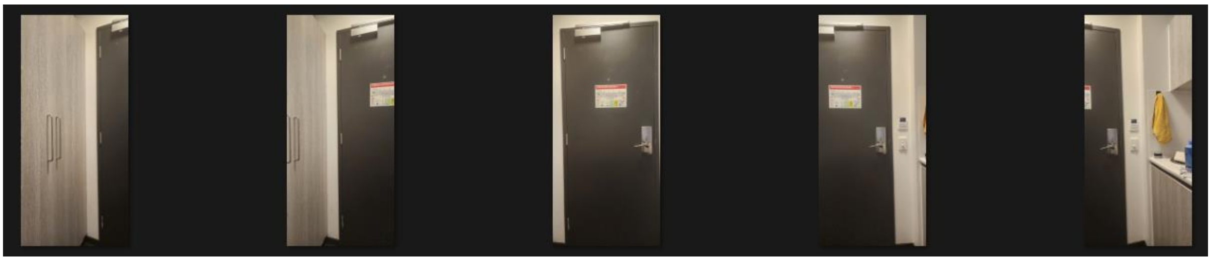


Figure 1. Entryway 001-005 (from left to right).

3.3. Matches plots comparison

Figure 2 and Figure 3 contains result images two steps of the whole matching process with all four different methods. Matches between image2 and image3 (figure 2).

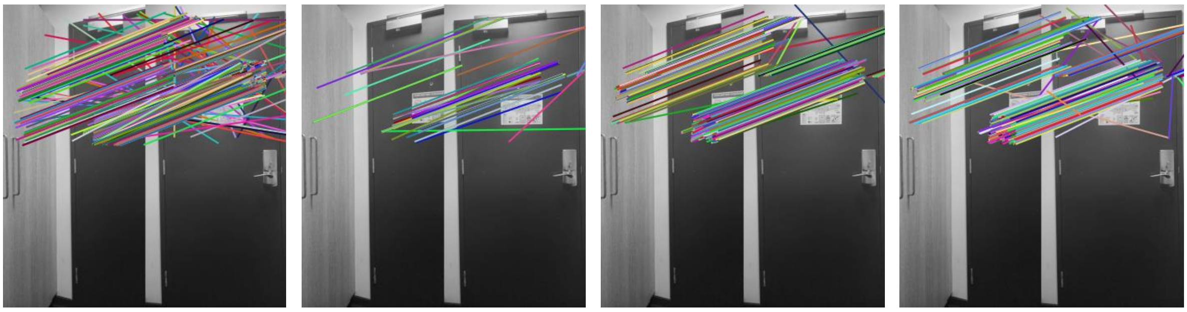


Figure 2. Matches between img2 and img3, from left to right: SIFT, ORB, AKAZE, BRISK.

Matches between image4 and image3 (figure 3)

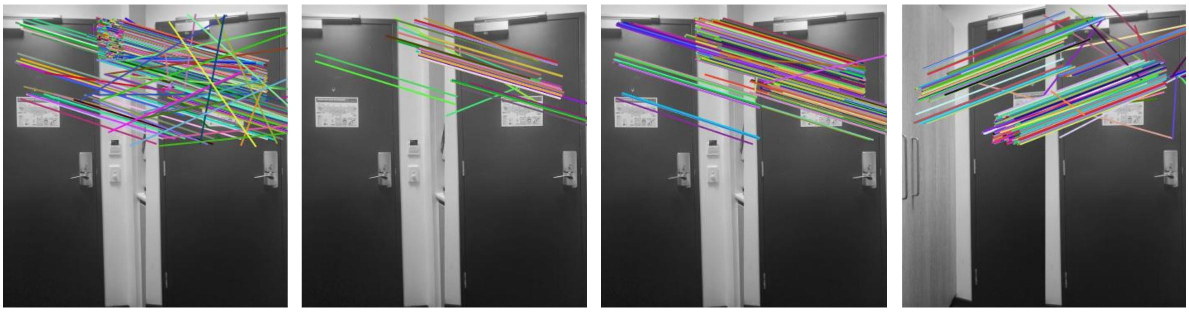


Figure 3. Matches between img4 and img3, from left to right: SIFT, ORB, AKAZE, BRISK.

As the figures above show, ORB has a fixed and limited feature points count, which also affected the accuracy of the matching process. AKAZE generally has fewer outliers and fewer feature points than SIFT and BRISK as result. Stitching results (figure 4).

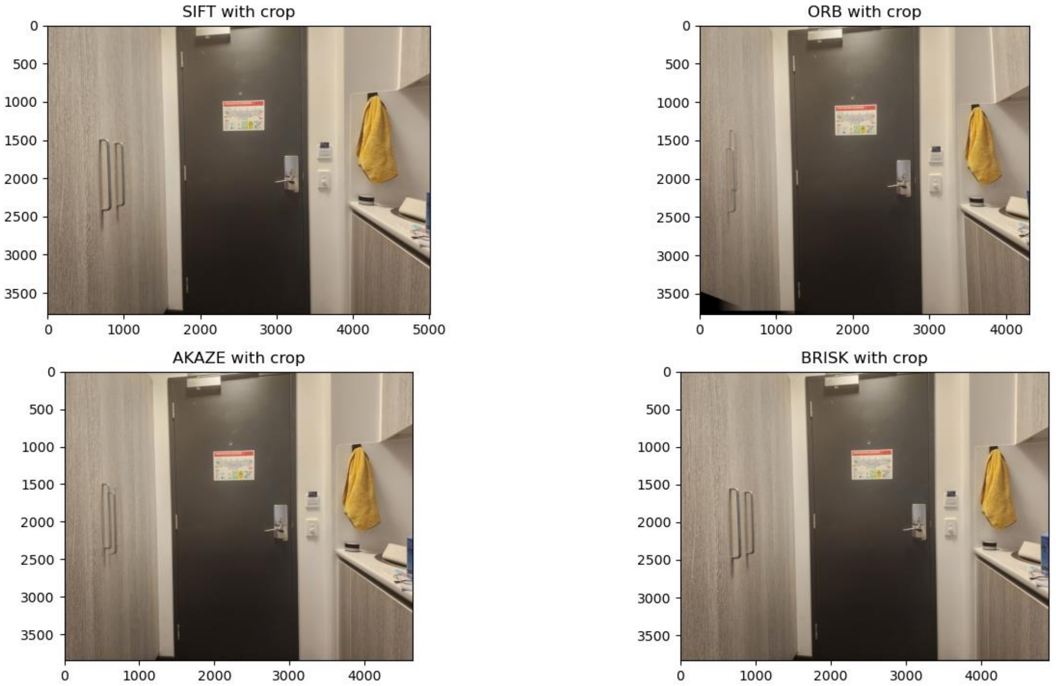


Figure 4. Stitching results, with used algorithms in the title.

As Figure 4 shows, the width of the four result images ranks as follows: SIFT> BRISK>AKAZE>ORB. SIFT preserved the most information from the source images and didn't glitch when dealing with the handlers on the closet.

3.4. Data comparison and analysing

Data collected during stitching img2 and img3 (Table 2).

Table 2. Data collected during stitching img2 and img3.

	Keypoint 1	Keypoint 2	Matches	Match Rate	Time
SIFT	8574	3667	827	0.23	4.26
ORB	500	500	58	0.11	1.55
AKAZE	1540	1847	407	0.26	2.26
BRISK	8916	5575	520	0.09	3.20

Data collected during stitching the combination of img2 and img3 with img1(Table 3).

Table 3. Data collected during stitching img2+3 and img1.

	Keypoint 1	Keypoint 2	Matches	Match Rate	Time
SIFT	23895	13845	585	0.04	23.52
ORB	500	500	22	0.04	1.41
AKAZE	949	1861	119	0.12	2.41
BRISK	19434	7251	188	0.02	6.66

Data collected during stitching img1+2+3 with img5+4+3(Table 4).

Table 4. Data collected during stitching img1+2+3 with img5+4+3.

	Keypoint 1	Keypoint 2	Matches	Match Rate	Time
SIFT	39784	22680	3333	0.15	75.46
ORB	500	500	209	0.41	1.56
AKAZE	3981	4969	1439	0.36	3.78
BRISK	13028	19620	3021	0.23	11.30

As the histogram below (figure 5) shows, the AKAZE algorithm keeps a relatively high match rate throughout the whole process, only to be surpassed by ORB at the final step.

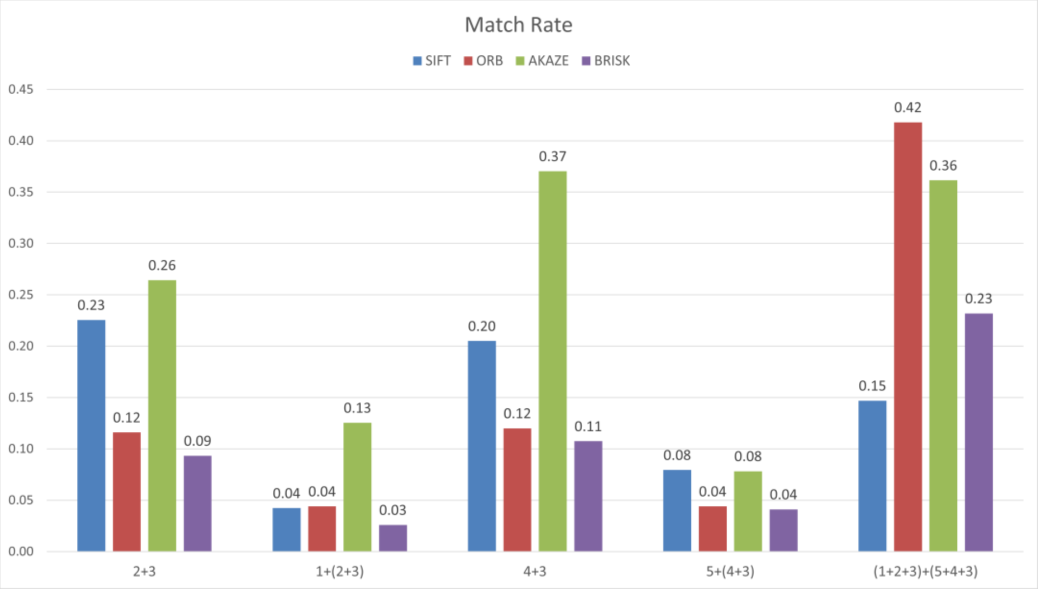


Figure 5. Match rates comparison.

As the histogram below (figure 6) shows, SIFT has generally the highest time costs because of the detecting procedure, and ORB has the lowest and the most stable time consumption due to fixed keypoints counts.

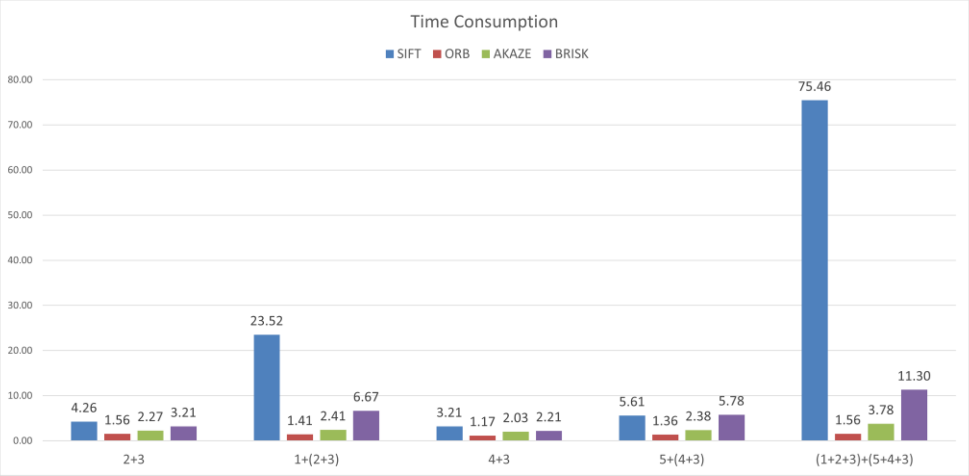


Figure 6. Time consumption comparison.

Throughout the entire testing process, the ORB detector has been the most time-efficient method in most cases and is currently the most stable feature detection and extraction algorithm. On the other hand, AKAZE usually will have higher match rates. SIFT is the most time-consuming one among all 4 algorithms, which doesn’t necessarily lead to higher catch rates or more accurate matches.

4. Conclusion

In conclusion, this study focused on the use of feature detection algorithms for image processing and computer vision applications. Specifically, this work investigated the effectiveness of SIFT, ORB, AKAZE, and BRISK detectors for feature point detection and matching in image sets. Through the use of the RANSAC algorithm to filter out outliers and calculate the homography matrix, this paper was able to compare the results of panoramic image splicing and smoothing using the four different feature detectors.

In the experiment section, target image sets were processed by 4 different detecting algorithms, to measure the match rate and time consumption of different detectors. Our findings indicate that the AKAZE detector is the fastest algorithm for feature point detection and extraction, while the SIFT detector is able to provide a larger number of feature points for more accurate matches, and ORB is the most time-efficient one.

References

[1] Brown, M., & Lowe, D. G. Recognizing panoramas. 2003, Inter. Conf. Com. Vis. 3, 1218.

[2] Szeliski, R. Computer Vision: Algorithms and Applications. 2011 Sci. Bus. Media.387.

[3] Jolhip, M. I., Minoi, J. L., & Lim, T. A comparative analysis of feature detection and matching algorithms for aerial image stitching. 2017, J. Tele., Elec. Com. Eng., 9(2-10), 85-90.

[4] Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. ORB: an efficient alternative to SIFT or SURF. 2011 Inter. Conf. Comp. Vis. 2564-2571.

[5] Lowe, D. G. Distinctive image features from scale-invariant keypoints. 2014 Inter. J. Comp. Vis, 60, 91-110.

[6] Karami, E., Prasad, S., & Shehata, M. Image matching using SIFT, SURF, BRIEF and ORB: performance comparison for distorted images. 2017 arXiv preprint arXiv:1710.02726.

[7] Leutenegger, S., Chli, M., & Siegwart, R. Y. BRISK: Binary robust invariant scalable keypoints. 2011 Inter. Conf. Comp. Vis. 2548-2555.

[8] Alcantarilla, P. F., Bartoli, A., & Davison, A. J. KAZE features. 2012 Euro. Conf. Comp. Vis., VI 12, 214-227.

[9] Ou, Y., Cai, Z., Lu, J., Dong, J., & Ling, Y. Evaluation of Image Feature Detection and Matching Algorithms. 2020 Inter. Conf. Comp. Comm. Sys. 220-224.

[10] Nickfarjam, A. M., Ebrahimpour-Komleh, H., & Tehrani, A. A. Binary image matching using scale invariant feature and hough transforms. 2018 Adv. Sci. Eng. Tech. Inter. Conf. 1-5.

How AI evolved with game and implementation of modern AI in game

Bohan Shen

School of Computer Science and Technology, Beijing Jiaotong University, Beijing, 100000, China.

20722015@bjtu.edu.cn

Abstract. For a long time, game was a relatively unrecognized area by academic community, which lacks detailed and sufficient discussion. But with the growth of game industry, game AI has become a heated topic in recent years. As an important and evolving application of AI, there is a need to better discuss the application and future improvement of game AI technologies. This paper introduced history and breakthrough of AI made in game area. And made discussion centered on current implementation of some popular approaches for game AI, followed by the possible future of these technologies. Some new implementations like procedural content generation were then covered to further discuss future implementations of AI in game area. All in all, hot spots and development prospects of this research topic were prospected to enlighten future development of game AI.

Keywords: game AI, supervised learning, reinforcement learning, deep learning.

1. Introduction

AI has been accompanying the game since the appearance of computer games. Eventually, the task of artificial intelligence is to accomplish tasks comparable to human. From the extending content of the game that provide entertainment for players to the AI that can perform better than the best players from human. The challenge of developing game AI has brought advance in computational intelligence, reinforcement learning and other AI methos.

The game design can be described as a process to build and provide player with well-orchestrated game content for players, including virtual characters' behavior, sounds, game mechanics, timing of different events and the entities that directly interact with the player [1]. In recent years, with the involvement of AI in more and more fields of game development, like AI generated video, model, voice and even codes, AI is now not simply the program that controls characters to react to players' movement. The game AI can be defined as the artificial intelligence that helps generate, manage and provide game content for game designers and players. With the growing complexity and diversity of games, AI techniques will be indispensable in the future game development.

While the industrial game production has a strong requirement of commercial and practical effect, the academic game AI methods mainly focus on advanced, but non-scalable approaches with limited paybacks. This gap results in the limited interconnection and exchange between research and industry during last decades and also the lack of recognization by the academic field and general public. However, some researchers argued for strengthen in game AI research and justified this research field [2]. Main arguments include:

- Problems in game can be considered as simplified tasks of real world. The research about game AI can help improve AI algorithms works in reality and provides an easier modeling way compared to modeling reality.
- By solving problems in game AI modeling can inspire how to solve problems in reality.

These arguments regard game as a form to simulate the reality. However, while game has been considered as an excellent field to evaluate the performance of AI systems, the increasing proportion of virtual world in people's life and users' higher requirement for the performance of game AI also requests for more systematic and academic development of game AI.

This paper mainly focuses on approach to AI algorithms in games. Introduce some innovations and breakthroughs appeared in the process of developing game AI. Then introduce how different AI techniques are implemented in modern games. Finally, having a further discussion about other implementation of AI in game industry, like procedural content generation and then take a further look into their future.

2. Backgrounds

Artificial intelligence has gained huge progress in recent years. From the early success in board games like AlphaGo to the recent popular area of various game play AIs, the evolution of algorithm makes researchers paying increasing attention to fields of video games with more variate and requirement of human-like performance, challenging the current AI research works [3]. In this section, we will first go through the development of game AI, focus on main breakthroughs in last decades to see how AI evolved with games. And then introduce the modelling problem of game AI at current stage to further discuss AI techniques and game types in the next section.

2.1. Game AI Evolution

At early stage, with limited hardware performance and immature AI model, finite state machine was implemented as a computation model that can simulate sequential logic and some computer programs. Situations in reality were predefined as input for the state machine to generate reaction as output like the figure 1 shown below.

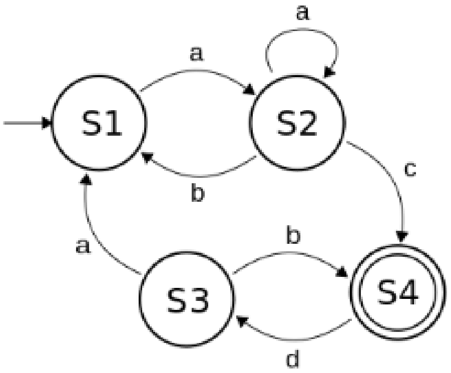


Figure 1. Finite state machine.

Later behavior tree was developed, using a tree-like structures to create and perform AI behaviors. But just like finite state machine, the cost of design limited their input possibilities and also scope of application. Therefore, most early research on game AI was focused on classic board games like checkers and chess as they have strictly constrained rules and clear input [4]. The complexity of these games keeps people's interest for thousands of years and set the goal for AI to beat the best human player.

The first breakthrough was made by Gerald Tesauro in 1992, he developed TD-Gammon using temporal difference learning to beat human players in backgammon [5]. Later in 1997, Deep Blue from IBM defeated the grandmaster of chess, marking the new development stage of game AI [6]. And in 2017, the viectory of AlphaGo to the world champoin Ke Jie declares game AI's success in mastring classic board games [3]. With the development of AI algorithms, classic board games are now relatively easy for game AI. Video based game with features of much more input possibilities, information asymmetry and the need of interacting with real person like a human has got the attestation of game AI research.

Started by Google in 2014, AI was trained to play classic video games [7]. Benefited from early research done on classic board games, the research on video games made rapid progress. Then in 2019, Google's game AI beat professional players of RTS games and game AI built by OpenAI beat professional players of MOBA games [8].

2.2. Modeling of Game AI

It could be seen that as a branch of AI, game AI gained huge successful over last ten years with the development and evolution of AI. But it is worth mentioning that approaches modeling and improving game AI is still quite different from other implementation of AI.

From the universal view, as a strong interactive activity, game must be understood so that it can be played by user and so does the AI. However, compared to other popular field of AI like computer vision and language mode, the implementation of game AI does not necessarily have to understand the current situation or extract useful information from the game, playing game is a process that need AI to keep engaging in the game environment and make decision to help it reach certain game purpose or performing like a real human based on the environment. For this point of view, game AI is actually a decision model that handles its choice of action and input environment, the process of making decision is the main issue concerned while building game AI.

The dominant work procedure of game AI is to divide the interaction between AI agent and the input environment into discrete steps [2]. Every time AI enters a new step of running, it makes decisions based on the input get from the environment and its decision, together with other possible action from other AI or player will lead the environment to a new state. Then AI will repeat strategies considering current state as new input environment of its step of running. This process is the intelligence of game Artificial Intelligence. Based on model constructed, there are also algorithms helps to evaluate decisions made by AI and help AI to choose the action with best evaluation in the whole process.

3. AI techniques and implementations

3.1. AI techniques

Based on the modeling method of game AI mentioned above, there are many approaches for building game AI. It is worth having more discussion on their implementation and possible improvements.

3.1.1. Reinforcement Learning. Reinforcement learning is a training method trains game AI to make decisions based on rewards and punishments received from the environment. In reinforcement learning for game AI, game AI will be given a specific goal, such as defeating the player, and then interacts with the game environment to achieve that goal, there will be certain standard of measuring AI's decision. As the AI takes actions in the game, it receives rewards or punishments based on the outcome of those actions and the measuring standards.

Over time, game AI can learn from these rewards and punishments, and uses that information to make better decisions in the future. The reinforcement learning algorithm continually updates the AI's decision-making process based on the outcomes of its actions, allowing it to improve its performance over time. Reinforcement learning can assign game AI with complex and adaptive behaviors. For example, AI using reinforcement learning could learn to adapt to the player's strategies and change its own behavior accordingly. This can result in a more engaging and challenging game experience for the player.

However, reinforcement learning can be computationally intensive and requires a significant amount of training data to achieve good results. Additionally, designing a game environment that is well-suited for reinforcement learning can be challenging, as it requires careful consideration of the rewards and punishments that the AI may receive. From recent research, combining deep neural networks with reinforcement learning may have remarkably performance in many generes of games, and its application could be extended to wider domain including level design and automated balancing in game [9].

3.1.2. Supervised Learning. Supervised learning is another approach that involves training using labeled data. Training AI on a set of labeled examples. For example, AI may be trained to recognize and respond to a specific type of player behavior with dataset of labeled examples that show how it should respond to players' certain behavior and then tested on a separate set of data to evaluate its performance.

AI model trained by supervised learning can be used to have specific behaviors or decision-making processes. Advantages of this approach include that it can be relatively fast and efficient compared to other machine learning techniques. And within limited game conditions, like MOBA games or FPS games focusing on separated rounds of game play, it can have relatively better performance [10]. However, there are still challenges for supervised learning in game AI. On the one hand, it requires a large amount of labeled training data to achieve good results, which can be time-consuming and expensive to obtain. On the other hand, it could be difficult to ensure that training data can accurately reflects the game environment. And AI model trained on a specific set of labeled examples may not perform well in situations outside of that training data, leading to unexpected or undesirable behaviors [8].

Overall, supervised learning can be a useful tool in game AI for creating NPCs with specific behaviors or decision-making processes. However, it must be used carefully and with consideration of the specific challenges and limitations of the game environment.

3.1.3. Deep Q Networks. In addition to neural network for Reinforcement Learning, Deep Q Networks (DQNs) is almost the most widely used approach for game AI [11]. It is mainly implemented to learn optimal policies for decision-making in games. The basic idea behind a DQN is to use a deep neural network to approximate the Q-function, which is a function that takes state and player action as input, and action of AI as output based on the expected reward for taking certain action in that state.

To be more specific, the state could be the current game environment or different parameters of the character controlled by AI, while the action could represent a move or decision that the AI can make. And the expected reward would be the expected outcome of taking that action in that state, the outcome is usually calculated and measured by scores to show determine the gain or loss of AI. The DQN works by updating the Q-function based on a loss function that measures the error between the predicted Q-values and the actual Q-values. The actual Q-values are calculated using a process called experience replay, which involves storing the AI's experiences (i.e. state, action, reward, and next state) in a memory buffer and randomly sampling from this buffer during training.

Eventually, the use of DQN could help AI learn to make better decisions over time as it gains experience and refines its Q-function approximation. This can lead to more challenging and realistic gameplay, as well as more competitive AI opponents for human players. For example, Deep Q Networks were used for AI competition and visual fighting games on Atari2600 games [12] (see Figure 2).

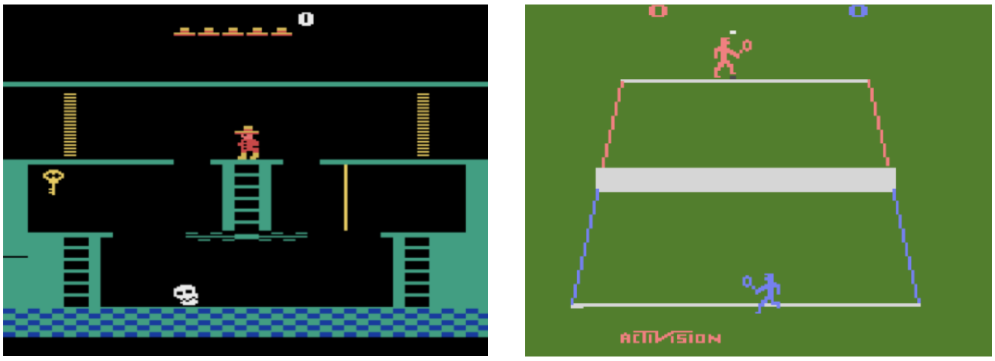


Figure 2. Atati2600 games used for Deep Q Networks training.

3.2. Strategies for implementing game AI

It could be seen that game AI has become indispensable part of modern game content, and as a complex and evolving approach, game AI needs strategies for its implementation. In this context, there are several strategies for implementing game AI that can be used to create engaging and complex gameplay.

3.2.1. Clear Design. Implementing a successful AI in a game starts with having a clear plan of what the AI needs to do and how it fits into the game design. A well-defined AI design can help identify what kind of AI is required, what behaviors it needs to exhibit, and what kinds of data it will need to process. While implementing game AI, well designed network structure would be the basement for future AI development and there should be serious consideration about acceptability of possible input that may added in the coming development. A poor designed network can lead to both undesired results and limit of reusability in future development.

3.2.2. Use of existing frameworks and tools. There are many AI frameworks and tools available to help simplify the development process. These can include libraries for pathfinding, behavior trees, an5o 06uyyurrrrrrry d decision-making, which can help the reduction of the need for complex code and improve the efficiency of development process.

3.2.3. Balancing simplicity with complexity. The AI should be well structured and simple to implement and maintain, while being complex enough to create engaging and challenging gameplay. Finding the right balance between these two aspects is essential to the success of the game AI. This issue becomes more important when considering the limit of users' hardware devices. To be more specific, being too complex means the higher threshold for user community while being not simple enough may fail to provide game content that can attract users for long-term play.

3.2.4. Testing and iteration. AI development is an iterative process, and it is essential to test early and often. This can involve creating test environments and scenarios to see how the AI performs, and making necessary adjustments. Continuous testing and iteration can be helpful for improving the AI's performance and ensure that it meets the design goals.

4. Procedural content generation in game production

4.1. Advantages of procedural content generation

Procedural content generation for game (PCG-G) is a subfield of computational creativity that focuses on the automatic generation of game content using algorithms and machine learning techniques. It can be used in game development to create game content, such as levels, landscapes, and characters algorithmically instead of designing them manually. The use of PCG-G in games has become more prevalent in recent years due to the increasing demand for personalized and dynamic game experiences.

This has led to the integration of AI algorithms into PCG-G systems to create more sophisticated and intelligent game content. According to Barriga and Nicolas (2019), approaches of PCG-G can be classified into two kinds, traditional Search-Based methods and Machine Learning methods [13].

Traditional methods mainly depend on Pesudo-random Number Generators, Gnerative Grameers, Fractals and Noise with advantages of efficiency and easiness [14]. However, as the complexity of video game is growing exponentially with the prohibitive content and cost of games in recent years, there raises needs for more intelligent and creative approaches for generating game content to provide personalized and dynamic game experiences. And this has led to the integration of AI algorithms into PCG systems to create more sophisticated and intelligent game content.

Although most AI solutions are designed for classification and prediction problems, including game AI models mentioned above, there are few ways work relatively for generating game content, like recurrent neural networks (RNNs) and Generative Adversarial Networks (GANs).

Take recurrent neural networks, for example. RNNs work by using feedback loops in the network architecture, allowing the network to maintain an internal state that can be updated based on previous inputs. This makes RNNs well-suited for generating game content that is dependent on previous actions or events in the game. Neural networks can be trained on existing game content to learn the patterns and structures of the game, and then generate new content that is similar in style and quality as inputs are given to the network cumulatively.

For GANs' approach, AI algorithms can analyze and learn from existing game content to try to generate and improve the performance of network. In this process, one algorithm will generate content based on users' requirements as input, and another algorithm evaluates the quality of the generated content, providing feedback to the generator algorithm to improve its output [15].

4.2. Controversy and future implementation of procedural content generation

While procedural content generation for game offers many benefits to game developers and players, it also raises some concerns and controversies that need to be addressed.

One major concern is that PCG-G could lead to games that lack creativity and originality. Critics argue that relying too heavily on PCG-G could result in games that feel repetitive and formulaic, with little room for innovation and surprises. And another concern is that PCG-G could perpetuate biases and stereotypes if the algorithms used to generate content are not carefully designed and tested. For example, if an AI algorithm is trained on a biased dataset, it may generate content that reflects those biases, perpetuating harmful stereotypes and reinforcing existing inequalities.

To mitigate this risk, game developers need to ensure that their PCG-G algorithms are designed with diversity and inclusion in mind, and that they are regularly audited to identify and address any biases. In terms of future implementation, PCG-G is likely to become even more prevalent in the game industry as technology advances and AI algorithms become more sophisticated. As PCG-G becomes more mainstream, we can expect to see more games that offer personalized experiences tailored to individual players, and more games that adapt and evolve based on player behavior and preferences. Additionally, PCG-G may become more integrated with virtual reality and other emerging technologies, providing even more immersive and interactive game experiences.

5. Conclusion

This paper mainly focused on the history and current implementation of game AI. Among which the history of game AI covered how AI technology evolved with game development and some important breakthroughs happened in this process followed by their corresponding implementations in game production in their own time. Then there was a brief introduction about modeling methods for game AI to introduce the system and approaches for implementing AI in modern games. Based on the introduced modeling methods, the paper expanded the game AI technology to some well-performed network structures that are widely used in game production and made a discussion about possible improvements for these network structures for further development. After that, there were some strategies concerned with implementing game AI to help the performance of AI in game production. Lastly, the heated spot procedural content generation that appeared recent years were evaluated by its advantages and controversy about it, followed by a look at its implementation in the future. Overall, the paper focused on the history and current implementation of various game AI technologies and discussed them from multi-aspect of view, and formed own opinions to their future based on the progress made by them.

References

[1] Westera, W., Prada, R., Mascarenhas, S., Santos, P.A., Dias, J., Guimarães, M. and Georgiadis, K. 'Artificial intelligence moving serious gaming: Presenting reusable game AI components', 2020 *Edu. Infor. Tech.*, **25(1)**, 351+.

[2] Risi, S., & Preuss, M. From chess and atari to starcraft and beyond: How game ai is driving the world of ai. 2020 *KI-Künst. Intell.*, **34**, 7-17.

[3] Silver, David, et al. Mastering the game of Go with deep neural networks and tree search. 2016 *Nature* **529.7587**: 484-489.

[4] Lu, Yunlong, and Wenxin Li. Techniques and Paradigms in Modern Game AI Systems. 2022 *Algorithms* 15.8: 282.

[5] Tesauro, Gerald. Temporal difference learning and TD-Gammon. 1995 *Commun. ACM* **38.3**: 58-68.

[6] Campbell, Murray, A. Joseph Hoane Jr, and Feng-hsiung Hsu. Deep blue. 2002 *Artif. Intell.* **134.1-2**: 57-83.

[7] Mnih, Volodymyr, et al. Playing atari with deep reinforcement learning. 2013 *arXiv preprint arXiv:1312.5602*.

[8] Berner, Christopher, et al. Dota 2 with large scale deep reinforcement learning. 2019 *arXiv preprint arXiv:1912.06680.*

[9] Oh, Inseok, et al. Creating pro-level AI for a real-time fighting game using deep reinforcement learning. 2021 *IEEE Trans. Games* **14.2**: 212-220.

[10] Ye, Deheng, et al. Towards playing full moba games with deep reinforcement learning. 2020 *Adv. Neur. Infor. Proce. Sys.* **33**: 621-632.

[11] Gunawan, Leonardo Jose, et al. Analyzing AI and the Impact in Video Games. 2022 *4th Inter.Conf. Cyber. Intell. Sys.,*1-9.

[12] Torrado, Ruben Rodriguez, et al. Deep reinforcement learning for general video game ai. 2018 *IEEE Conf. Comput. Intell. Games*, 1-11.

[13] Barriga, Nicolas A. A short introduction to procedural content generation algorithms for videogames. 2019 *Intern. J. Artif. Intell. Tools* **28.02**: 1930001.

[14] Zhang, Yuzhong, Guixuan Zhang, and Xinyuan Huang. A Survey of Procedural Content Generation for Games. 2022 *Inter. Conf. Cul.Orient. Sci. Tech.*, 1-10.

[15] Liu, Jialin, et al. Deep learning for procedural content generation. 2021 *Neu.l Comput. Appl.* **33.1**: 19-37.

Semantic information based solution for visual SLAM in dynamic environment

Chuqi Shao

College of Mathematics and informatics College of Software Engineering, South China Agricultural University, Guangzhou 510642, China

chuqi@stu.scau.edu.cn

Abstract. In recent years, the utilization of visual SLAM with a camera as a sensor has become increasingly widespread, particularly in the context of rapidly developing artificial intelligence such as mobile robots, VR, and AR. This approach is favored due to its affordability, lightweight design, ability to capture comprehensive information, and other advantages. The traditional slam technology has achieved a very mature effect with the static scene as the assumption condition, and the classic representatives are LSD-SLAM and ORB-SLAM which will be introduced in the following. However, since dynamic scenarios are unavoidable in the real world, overcoming the influence of dynamic objects becomes a challenge if researchers want to move forward with more applications. At present, under the dynamic environment of the visual slam algorithm faces positioning problems of low accuracy and poor robustness. To address the challenges posed by dynamic objects in a scene, many researchers are incorporating deep learning techniques and exploring the use of reference semantic information to collaboratively resolve the issue. This paper reviews this and summarizes the development process and important algorithms.

Keywords: visual SLAM, semantic segmentation, dynamic environment.

1. Introduction

SLAM, which stands for Simultaneous Localization and Mapping, is a crucial and essential function of robotic applications. Building a map of an unexplored area based on data from the sensors is a key objective of SLAM, which aims to accomplish this task while minimizing the system's weight. SLAM is mainly used in robot navigation, autopilot, Augmented Reality(AR), and other areas of the widely. Depending on different types of sensors, SLAM can be split in two different directions. One is laser SLAM based on Lidar, and the other is visual SLAM implemented by cameras. Visual SLAM has a wider application than laser sensors since the camera can capture more environmental information for the system. In other words, its core is to obtain RGB and depth information.

Most visual SLAM methods are basically assumed to be in a static environment with no dynamic objects so as to facilitate the implementation of the algorithm and excellent performance has been achieved in this way. Such ideal environments limit most applications of visual SLAM systems under dynamic environments. Currently, accurately and reliably providing real-time information about the position or whereabouts of objects in physical settings is a significant challenge for nearly all existing visual SLAM systems due to their inability to handle dynamic targets in dynamic environments.

Moving objects cause errors in the calculation of camera motion in dynamic environments, which ultimately leads to low localization accuracy and poor robustness in the system. In addition, the emergence of dynamic objects will also affect the ability to efficiently and accurately process sensor data and provide updated estimates of the camera pose and map, increasing the computing cost and delay.

To solve the above problems, the key lies in the correct detection of dynamic targets in the visual field, and how to deal with the dynamic target subsequently. Currently, several solutions have been proposed for visual SLAM techniques to deal with scenes that contain moving objects or changing lighting conditions. Facing the additional challenge of operating in a state of flux, Cheng J et al. proposed DM-SLAM, which is a combination of optical flow information and instance segmentation network [1]. Tete Ji proposed that RGB-D SLAM uses semantic information to identify dynamic objects and eliminate their effects, and only maintains static maps containing camera tracking [2]. Based on the characteristics of point and line dynamic scene of PLD - SLAM calculates camera position, the robustness and position precision are also improved [3]. In addition, RDS-SLAM uses semantic segmentation results to detect dynamic targets and remove any abnormal or irrelevant data points, while maintaining the whole process in real-time [4].

In short, the challenges visual SLAM faces in dynamic environments are constantly being proposed, although it is facing great challenges. In addition to these methods, several papers have proposed new frameworks for both localization and semantic segmentation, improving their performance through the intermediate results of the two modules. These new methods and frameworks are developing and improving constantly, which provides more ideas and directions for solving the challenges.

The rest of the paper is structured as follows: Sect. 2 describes the structure of the visual slam and the details of each part. In Sect 3: LSD-SLAM and three generations of OGR-SLAM are introduced in detail, which are visual SLAM in a static environment represented by the direct method and feature method respectively. Section 4 is to summarize the current visual slam algorithm combined with semantic information in the dynamic environment. The final section provides conclusions and a discussion about the challenges that still need to be addressed in the future.

2. Visual SLAM

The basic framework structure of the visual SLAM algorithm can be seen in figure1. Visual SLAM has the capability of initiating from an unknown location in an unfamiliar environment based on mobile devices, such as robots, drones, and mobile phones, observing and locating its own position and posture through the camera in the process of movement, and then build incremental maps according to its posture. The ultimate goal is to achieve positioning and map construction at the same time.

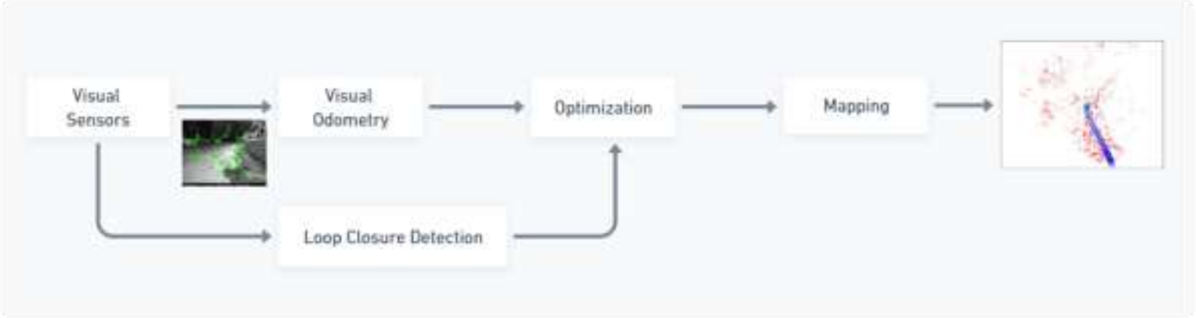


Figure 1. The structure of visual slam.

2.1. Visual sensors

Visual SLAM technology relies on a camera as its primary sensor to perceive and gather information about the surrounding environment, allowing for the simultaneous determination of the robot, vehicle, or camera's position in real time. This approach is cheap, lightweight, and versatile, making it an attractive option for a wide range of applications. At present, there have been various types of cameras used in practice and research, including but not limited to monoculars, stereo, RGB-D, pinhole, and fisheye cameras. Visual SLAM systems can even run in micro PCS and embedded devices, and it can be implemented with only a single monocular camera, which is the most cost-effective and smallest sensor device. The mainstream representative single-purpose Visual SLAM methods can be categorized into three groups: Mono-SLAM [5] which uses a filter, PTAM [6] based on keyframe BA and LSD-SLAM [7] based on direct tracking. However, the biggest drawback of a monocular camera is that depth information (distance) cannot be obtained from a single image, and there will be scale drift. The emergence of a binocular camera just solves this problem. It uses the parallax of left and right objects to calculate the distance of pixels, to realize its positioning. RGB-D cameras are preferable to monocular cameras as they can offer more accurate depth information, which enhances the robustness and accuracy of visual SLAM.

2.2. *Visual odometry*

The visual odometer technique is a process in computer vision where the camera's movement and position are approximated through several multiple consecutive images. Due to the different types of sensors, Visual Odometry can be divided into three categories: direct method, feature point method, and RGB method. The Feature-based approach is the dominant method of the visual odometer, which involves the following three steps: identifying and extracting significant features from consecutive images, matching the features between the images to track their motion, and estimating the camera's movement by optimizing the feature point correspondences. Feature point extraction and matching are often implemented using descriptors and key points, such as ORB-SLAM, and PTAM. Completely different from the feature points, the direct method bypasses the feature point extraction step and instead directly uses the pixel value of consecutive images, specifically the gray-level intensity values of pixels, to estimate the camera's motion. It is considered a complementary approach to the feature points, such as LSD-SLAM. In addition, more scene information can be provided based on the depth information and color(RGB) provided by the RGB-D sensor. However, due to the high cost and high condition requirements, the application is relatively rare. Some algorithms fuse the feature method with the direct method, such as SVO [8].

2.3. *Loop closure*

Loop closure also referred to as closed-loop detection, is a critical component of Visual SLAM which is utilized to distinguish whether the current observation is from a repeated arrival position or an unexplored location. Davison et al [5]. found the problem of cumulative errors in the MonoSLAM algorithm during the experiment, which caused the deviation and inaccuracy of the SLAM trajectory. This is later referred to as the cumulative error. At present, the popular loop closure method is Bag of Words (BoW) which is generally applied for feature points such as ORB-SLAM. However, this method uses extracted features to judge whether the situation is the same or not, while the direct method does not extract features. If loop closure is needed, the model based on the direct method may need to extract additional features. There are also cases like DSO [9] that are not a complete slam because it lacks loopback detection, resulting in it not eliminating the cumulative error, albeit small.

2.4. *Optimization*

Optimization in the way of implementation is mainly two kinds of filter method and nonlinear optimization method, also known as the back-end. This module is optimized for receiving information about camera poses from visual odometry measurements at different times, as well as loop detection, resulting in a globally consistent map. The filter optimization method includes the Kalman filter and extended Kalman filter, while the nonlinear optimization method includes Bundle Adjustment(BA), PoseGraph optimization, and factor graph optimization. Due to the existence of frame loss, this sparsity, reflected in the matrix operation can be solved by mathematical techniques such as elimination, which makes the nonlinear optimization method can be applied to a real-time SLAM system, called graph optimization. PTAM, ORB-SLAM, LSD-SLAM, and so on are all graph-optimized backends.

3. Classic approaches

Traditional methods can be broadly classified into two categories: direct method and feature-based method, both of which use image information to process problems. Next, the direct approach leading to semi-dense and dense constructs in the case of LSD-SLAM, and the feature-based approach leading to sparse constructs using the three generations of ORB-SLAM as an example, are presented respectively.

3.1. LSD-SLAM

A direct method-based monocular SLAM algorithm, represented by LSD-SLAM, can construct large-scale and consistent semi-dense maps. It exploits the characteristic of monocular slam, namely scale-ambiguity, to seamlessly switch between environments of different scales. LSD-SLAM is based on the direct method to match image feature points, that is, to perform direct, scale-shift-aware image alignment on sim (3), avoiding the problem of scale drift. Because the direct method is to use the pixel value of the image to match, independent of the feature points [7]. The traditional feature method can extract feature points and calculate descriptors. This method can ensure the accuracy of matching to a certain extent, but it also leads to scaling uncertainty.

The algorithm typically comprises three primary components: tracking, depth map estimation, and map optimization. After obtaining the camera pose and map point position in the initialization phase, LSD-SLAM utilizes the direct method for tracking, which involves calculating the camera’s motion through the displacement information between the current image captured by the sensor and the previous frame. During the depth map estimation process, the depth of map points is calculated by the triangulation method, and the estimated value of the depth map is updated by optimizing the position and orientation of the camera in the environment and the map point position. In the map optimization process, LSD-SLAM uses direct rendering for map optimization. The whole process uses the local luminosity error as the matching measure to calculate the data error. This new direct monocular slam algorithm shows more functionality, robustness, and flexibility.

3.2. ORB-SLAM

Visual SLAM can be executed with just one monocular camera, which is the most affordable and compact, but also versatile enough to function using a wide range of settings. ORB-SLAM improves on PTAM's algorithmic framework and contains three threads running in parallel: tracking, local mapping, and loop closure. ORB-SLAM [10] is a monocular complete SLAM system that is solely based on sparse feature points, and its fundamental principle is to employ Oriented FAST and BRIEF(ORB) as the key feature of the entire visual SLAM, which makes the system simpler and more robust. The ORB is extremely fast to calculate and match, with a good point-of-view invariance.

ORB-SLAM2 [11] added binocular stereo vision and RGB-D based on ORB-SLAM1 single purpose, which is a set of support for monocular, binocular, and RGB-D complete program, with three main parallel threads: map reuse, loop closing, relocation. An advanced version of ORB-SLAM2 and ORBSLAM-VI is ORB-SLAM3, which is a visual SLAM system that supports vision, vision-plus navigation, and hybrid maps, and can be operated on a variety of cameras [12]. The first major innovation refers to a tightly integrated Visual-Inertial SLAM system that can real-time closed loop and favors the map's sensors over already mapped areas. ORB-SLAM3 was the first visual system and visual inertia system that can accurately match current sensor data with previously mapped data at different time scales, across multiple maps, which was also the key to accuracy. In summary, Compared with other methods of the most advanced monocular SLAM, ORB-SLAM achieved unprecedented performance.

4. Deep learning

Most traditional slam algorithms, including those mentioned above, assume static scenarios, while dynamic objects cannot be avoided in real scenes, especially to the slam is applied to more scenes. With the rapid development of deep learning, if deep learning technology can be used to deal with dynamic object problems, there will be better development and research direction. In recent years, there are some research on feature extraction and motion estimation, especially on semantic information and depth information. The existing algorithm research combined with semantic information visual SLAM algorithm in recent years is summarized in Table 1. The fr3_walking_xyz and fr3_walking_static in the TUM RGB-D dataset and the Average Displacement (RMSE) error in the KITTI dataset were used as the basis for judging the excellence of the algorithm.

Table 1. Comparison of slam algorithms based on semantic information.

Year	SLAM	Author	Character	KITTI	fr3_walking_xyz	fr3_walking_static
2020	DM-SLAM [1]	Junhao Cheng	Feature-based, Support for monocular, stereo, and RGB-D sensors	2.190	0.0148	0.0079
2020	PSPNet-SLAM[13]	Zhihong Xi	Pyramid scene analysis network, Dynamic scene based on semantic segmentation	-	0.016	0.008
2020	SaD-SLAM [14]	Xun Yuan, Song Chen	Semantic and deep information, Based on ORB-SLAM2	-	0.0167	0.0166
2021	PLD-SLAM [3]	Chengyang Zhang	Point and line features, RGB-D dynamic SLAM method	-	0.0144	0.0065
2021	RDS-SLAM [4]	Yubao Liu	Semantic segmentation method, ORB-SLAM3 real-time visual SLAM	-	0.0269	0.0221
2022	STDC-SLAM [15]	Zgfang Hu	Real-time Semantic SLAM system, ORB-SLAM3, and Qtree-ORB algorithm framework, STDC network	-	0.018	-
2022	STDyn-SLAM [16]	Daniela Esparza	Stereo vision, Dynamic outdoor environment, Semantic segmentation	1.382	-	-

There are some novel visual SLAM techniques that achieve excellent performance in highly dynamic environments, regardless of the sensor used, such as DM-SLAM which incorporates optical flow and semantic masks. It leverages semantic segmentation, self-motion estimation, and dynamic point detection in conjunction with a feature-based SLAM framework to mitigate the impact of dynamic objects [1]. Similarly, RDS-SLAM [4] uses dependable feature points to estimate the state of the camera, which is based on the feature points in the static state of the movable object. Moreover, it also uses Mask_Rcnn to obtain semantic information and depth information from the RGB-D camera to discard moving feature points. Chenyang Zhang [3] also proposed that in the framework of RGB-D SLAM, the point-and-line features are used to calculate the posture of the camera in dynamic SLAM, and the semantic segmentation network, named MobileNet, and K-Means algorithm are combined to remove the dynamic features in the scene, which has improved the effect to some extent. Based on ORB-SLAM2, PSPNet-SLAM refers to a parallel semantic thread PSPNet for semantic segmentation at the pixel level. Through pyramidal network structure, it is more effective to obtain more context information and the relation between objects in pixels than the detection based on dynamic feature points. The algorithm also proposes a reverse ant colony search strategy that utilizes dynamic point community distribution to identify and use the most relevant and informative points, which enhances the robustness and achieves accurate and responsive visual SLAM in real-world applications [13].

Such methods typically have an architecture that requires waiting for semantic results in the tracing thread, and processing times that depend on the segmentation method used are not very friendly to demanding applications, such as tasks that need to be completed in real-time. Zgfang Hu et al. [15] proposed an approach that involves incorporating a semantic tracking thread and a semantic-based optimization thread into ORB-SLAM3 for improved performance. The key frame selection strategy of this design is adopted to obtain the latest semantic information to the maximum extent, and the processing of the segmentation method of different speeds, to ensure that the new thread and tracking thread run in parallel and maintain the real-time effect. STDCyn-SLAM [16] is also a real-time system and uses STDC as a semantic segmentation network for semantic thread analysis. Then the dynamic object segmentation graph is obtained. A refinement module is designed to improve semantic segmentation mapping by utilizing image depth information, which is superior to PSPNet-SLAM in localization accuracy and processing speed. Although using a semantic segmentation network can improve the precision and correctness of the system, the segmentation accuracy needs to be improved, as it is easy to lead to tracking faults where the surroundings are rapidly and constantly changing.

5. Conclusion

Nowadays Visual SLAM combined with deep learning technology has become one of the hot research fields. This article introduces the classic framework of visual SLAM, including sensors, the visual odometer, back-end optimization, and loopback detection. Two classical models, LSD-SLAM based on the direct method and ORG-SLAM based on the feature point method, are also introduced. For dynamic environments, there are research and discussions based on deep learning technology to solve dynamic objects. By using semantic information, pixels in the image are divided into different categories, such as sky, vehicle, and floor, which can better account for the presence of moving subjects and improve the precision and robustness of positioning and built figure. In addition, deep learning can also be used to improve feature extraction and motion estimation. For example, feature extraction based on Convolutional Neural Networks(CNN) can better deal with visual SLAM in dynamic scenes, and Recurrent Neural Networks (RNN) can be used to predict the trajectory of dynamic objects.

To some extent, using semantic information to eliminate dynamic objects improves robustness and accuracy in dynamic scenes. However, breakthroughs are still needed in the following aspects in the future. Firstly, the discernable types of moving objects are limited to some extent. Secondly, many static potentials moving objects such as parked vehicles are not well processed. Then, the accuracy of the semantic segmentation network needs to be enhanced, especially in a highly dynamic environment, which is easy to lead to tracking faults. Finally, the running speed is relatively slow, and improving the real-time performance would enable the system to process and analyze data more quickly, allowing for faster adaptation to changes in the environment.

References

[1] Cheng J, Wang Z, Zhou H, Li L and Yao J 2020 DM-SLAM: A Feature-Based SLAM System for Rigid Dynamic Scenes. ISPRS International Journal of Geo-Information; 9(4):202

[2] Ji T, Wang C and Xie L 2021 "Towards Real-time Semantic RGB-D SLAM in Dynamic Environments," 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, pp. 11175-11181

[3] Zhang C, Huang T, Zhang R and Yi X 2021 "PLD-SLAM: A New RGB-D SLAM Method with Point and Line Features for Indoor Dynamic Scene," ISPRS International Journal of Geo-Information. 2021; 10(3):163

[4] Liu Y and Miura J 2021 "RDS-SLAM: Real-Time Dynamic SLAM Using Semantic Segmentation Methods," IEEE Access, 9, 23772-23785

[5] A. J. Davison, I. D. Reid, N. D. Molton and O. Stasse 2007 "MonoSLAM: Real-Time Single Camera SLAM," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 6, pp. 1052-1067

[6] Klein G and Murray D 2007 "Parallel Tracking and Mapping for Small AR Workspaces," 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, Nara, Japan, pp. 225-234

[7] ngel, J., Schöps, T., and Cremers, D. 2014 "LSD-SLAM: Large-Scale Direct Monocular SLAM," In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds) Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, vol 8690. Springer, Cham.

[8] Forster C, Pizzoli M and Scaramuzza D 2014 "SVO: Fast semi-direct monocular visual odometry," 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, pp. 15-22

[9] J. Engel, V. Koltun and D. Cremers 2018 "Direct Sparse Odometry," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 3, pp. 611-625, 1

[10] R. Mur-Artal, J. M. M. Montiel and J. D. Tardós 2015 "ORB-SLAM: A Versatile and Accurate Monocular SLAM System," in IEEE Transactions on Robotics, vol. 31, no. 5, pp. 1147-1163

[11] R. Mur-Artal and J. D. Tardós 2017 "ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras," in IEEE Transactions on Robotics, vol. 33, no. 5, pp. 1255-1262

[12] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel and J. D. Tardós 2021 "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM," in IEEE Transactions on Robotics, vol. 37, no. 6, pp. 1874-1890

[13] X. Long, W. Zhang and B. Zhao 2020 "PSPNet-SLAM: A Semantic SLAM Detect Dynamic Object by Pyramid Scene Parsing Network," in IEEE Access, vol. 8, pp. 214685-214695

[14] X. Yuan and S. Chen 2020 "SaD-SLAM: A Visual SLAM Based on Semantic and Depth Information," 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, pp. 4930-4935

[15] Z. Hu, J. Chen, Y. Luo and Y. Zhang, 2022 "STDC-SLAM: A Real-Time Semantic SLAM Detect Object by Short-Term Dense Concatenate Network," in IEEE Access, vol. 10, pp. 129419-129428

[16] Esparza D and Flores G 2022 "The STDyn-SLAM: A Stereo Vision and Semantic Segmentation Approach for VSLAM in Dynamic Outdoor Environments," in IEEE Access, vol. 10, pp. 18201-18209

Image classification based on CNN with three different networks

Wenyan Sun

Shanghai Lixin University of Accounting and Finance, Shanghai 201209, P.R.China

201330122@stu.lixin.edu.cn

Abstract. Image classification refers to classifying images based on the different features reflected by the information in each image. Image classification is a fundamental issue in computer vision and has important significance. It has a wide range of applications, such as autonomous driving, face recognition, image retrieval, and other fields. This article first gives a bird's-eye view of the development of image classification and briefly introduces the factors that affect the accuracy of convolutional neural networks. Experiments and comparative analysis are conducted on the effects of convolutional layer numbers and optimizers on the accuracy of convolutional neural networks. Three convolutional neural networks with different convolutional layer numbers are built, their structural design is introduced in detail, and their structural diagrams were given. Different models are used to classify images from the CIFAR 10 dataset. The experimental results show that with the continuous increase of the convolutional layer, the accuracy rate is continually improving, but the program running time is continually increasing. In addition, using different optimizers can lead to changes in accuracy.

Keywords: image classification, deep learning, convolutional neural network.

1. Introduction

Image classification refers to distinguishing images based on their different characteristics reflected in the image information. It is a fundamental issue in computer vision. In addition, image recognition has important significance in daily life, and is applied in many fields, such as automatic driving, face recognition, automatic album classification, image retrieval, and so on.

Before proposing deep learning algorithms, traditional algorithms such as SVM and random forest classifiers are often used to solve image classification problems. Using traditional image classification algorithms usually includes several stages, such as feature extraction, feature coding, classifier design, and image classification. In 1998, Lecun Y et al. proposed using the LeNet5 algorithm to classify handwritten data sets (MNIST) and found that convolutional neural networks are superior to other techniques in processing two-dimensional shapes [1]. Since then, the application of convolutional neural networks in image recognition has entered people's vision, and related research has also begun to expand, such as image recognition, character recognition, gesture recognition, and so on. Hinton G E and Salakhutdinov R R (2006) first proposed the concept of deep learning and found that deep networks can better achieve dimensionality reduction of data than principal component analysis [2]. In 2006, deep learning algorithms received widespread attention again. Alex Krizhevsky et al. (2017) constructed and trained a large-scale deep convolutional neural network, and added a "discard" regularization method.

In the ILSVRC-2012 competition, they obtained a high accuracy rate and won first place, with a difference of nearly 10% from the second place [3]. The significant improvement in the accuracy of convolutional neural network algorithms and the development of discarding layers have caused more and more people to pay attention to convolutional neural networks. With this disruptive initiative, convolutional neural networks have begun to be widely used in various fields.

Convolutional neural networks (CNN) are mainly used in computer vision and natural language processing. Convolutional neural networks have long been one of the core algorithms in the field of image recognition. Chaganti S Y et al. (2020) used support vector machines (SVM) and CNN to classify images, and compared the accuracy of classification, proving that CNN has better results than traditional machine learning algorithms in image classification under large data sets for large-scale image classification problems [4]. In addition, in 2022, Tripathi, S, and Singh, R used CNN to classify cat and dog images, and found that CNN can automatically extract features without requiring feature engineering, and CNN has better accuracy in image classification than other algorithms [5].

Actually, the performance of convolutional neural networks (CNN) is affected by many factors. In 2015, Nielsen MA mentioned in Neural Networks and Deep Learning that increasing the number of convolutional layers used in convolutional neural networks (CNN) may improve the accuracy of image recognition. In addition, increasing the number of full connections, adding regularization, waiver, pooling, and other methods may improve the accuracy of the model [6]. In 2014, Kingma D and Ba J first proposed the Adam algorithm, and compared with other random optimization methods, Adam performed better in experiments. In addition, they also proposed a variant algorithm of Adam, Adamax [7]. Ruder S (2016) studied different variants of gradient descent, summarized the difficulties encountered, and introduced common optimization algorithms [8].

According to the existing literature, it can be found that CNN has good effects in image classification, image recognition, and other fields. However, the performance of CNN is affected by many factors, such as the number of convolution layers, the number of fully connected layers, pooling layers, and descending algorithms. However, there are few comparative experiments and studies on different convolution levels and different descent algorithms in existing papers. Based on this, this article will discuss and compare the impact of different convolution levels and descent algorithms on the accuracy rate when using the same dataset.

2. Method

2.1. Brief introduction of the CIFAR-10 dataset

The CIFAR-10 dataset consists of 10 categories of 60000 color images, including cats, dogs, airplanes, and so on, and these categories are completely mutually exclusive. Each class has 6000 images, with an image size of 32 * 32. The dataset is divided into five training batches and one test batch. Each training batch consists of a total of 10000 randomly selected images from each category, and randomly selected images are not repeatedly selected [9].

2.2. Data pre-processing

This time, 50000 training set data are used. For this 50000 data, first, convert all images into specific numbers, and then divide the data into training data and verification sets. The training set is used to train the model, and the verification set is used to verify the quality of the model and the calculation accuracy. There are 40000 training data and 10000 verification data. The dataset is classified using random grouping and the random_split() function. Then, use the DataLoader() function to group and bundle the data in the two datasets, and process the data in small batches each time. All the batch sizes in this model are 128.

2.3. Model structure adopted

2.3.1. Model construction and training

The model uses a convolutional neural network. First, the class ImageClassifier() is defined, which classifies images. For the Cifar10 dataset, images are divided into 10 categories. Establish a convolutional neural network, and establish a model based on the initial number of input channels of 3 and the number of convolutional layers to be established. The activation function used in this convolutional neural network is the ReLU function. Then, using the BCELoss loss function, select an appropriate descent algorithm, test the loss value of the training dataset, and continuously train the model through cycles to continuously reduce the loss value, achieving the goal of model optimization.

2.3.2. Model testing and calculation method of accuracy

According to the above training model, the previously separated test set is substituted into the trained model for testing. Through variable cycling, each image is predicted and then compared with a given type to determine whether the prediction is correct, thereby calculating the accuracy of the model prediction.

2.3.3. Experimental process of different models

The general structure of the model is described above. The main structures of the following different models are basically the same. In Model 1 and Model 2, only the number of layers of the convolution layer will be changed, keeping the optimization function, activation function, and so on unchanged, using the Adam optimization algorithm, In model 3, different optimization functions will be used for comparison, keeping the convolution layer number, activation function, etc. unchanged.

2.3.4. Model 1 – three convolution layers

The convolutional neural network in Model 1 is a 3-layer convolutional layer, which maintains the previously described data processing process, only changing the number of convolutional layers. Set the kernel size of each layer of the convolutional layer to 3 * 3, and the padding to 1. Change the number of input channels, output channels, and stride. The convolution layer of the first layer has an input value of 3, an output value of 96, and a default value of 1, The input channels of the second layer of the convolutional layer are the same as the out channels of the first layer of convolutional layer, which is 96. The out channels of the second layer are defined as 384, with a side of 2, Similarly, the in channels of the third layer of the convolution layer is 384, the out channels are 256, and the side is 2, Finally, add a pooling layer and select the MaxPool2d function where kernel size is 3.The structural diagram of model 1 is shown in Figure 1:

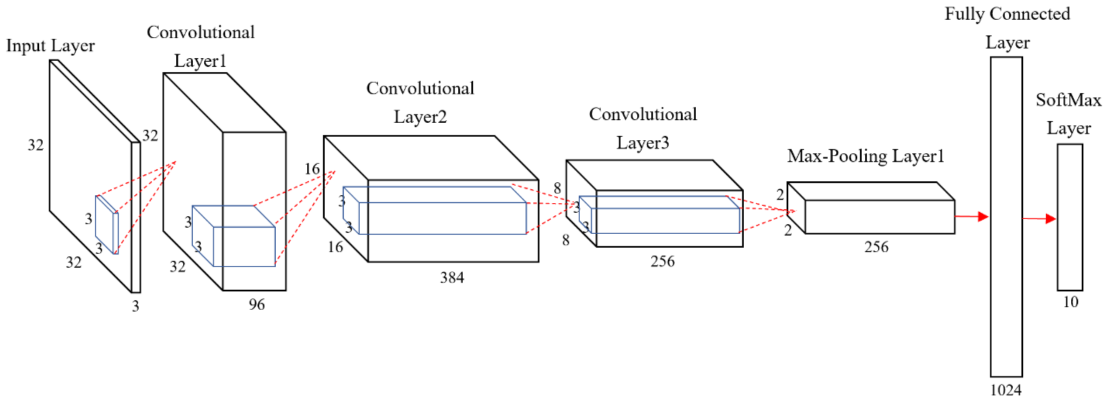


Figure 1. CNN Model 1 Structure Diagram

2.3.5. *Model 2 – four convolution layers*

The convolutional neural network in model 2 is a 4-layer convolutional layer, maintaining the same basic assumptions as model 1. The convolution layer of the first layer is the same as model 1, The input channels of the second layer of convolution layer 6 are the same as the output channels of the first layer of the convolution layer, and the output channels of the second layer are defined as 256, Add a pooling layer and select the MaxPool2d function, where the kernel_ Size is 2, Similarly, the input channels of the third layer of convolution layer are 256, and the output channels are 384, The fourth layer of convolutional layer has 384 input channels and 256 output channels, Finally, add a pooling layer and select the MaxPool2d function where kernel size is 2.The structural diagram of model 2 is shown in Figure 2:

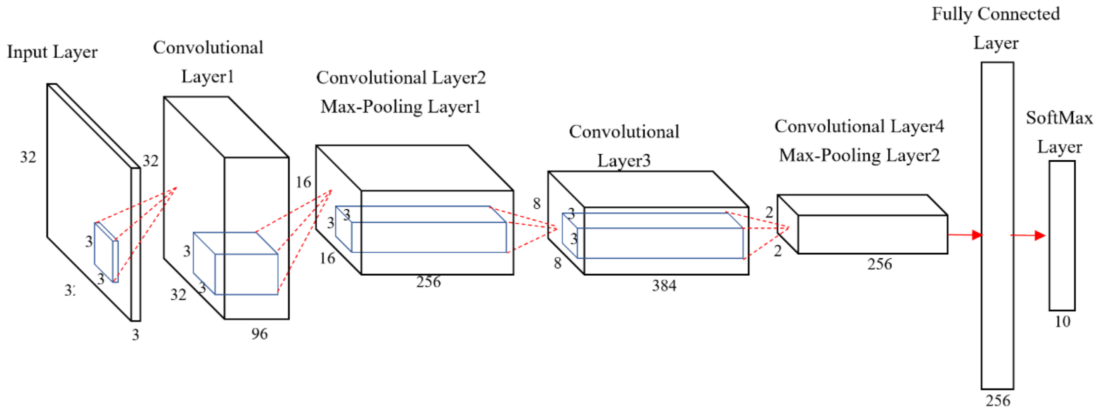


Figure 2. CNN Model 2 Structure Diagram

2.3.6. *Model 3 – five convolution layers*

The convolutional neural network in model 3 is a 5-layer convolutional layer, maintaining the same basic assumptions as model 1. The first and second convolution layers are the same as model two, Add a pooling layer and select the MaxPool2d function, where kernel size is 3, Similarly, the Input channels of the third layer of the convolution layer are 256, and the Output channels are 384, Add a pooling layer and select the MaxPool2d function, where kernel size is 3, The Input channels and Output channels of the fourth volume layer are 384 and 384 respectively, The Input channels of the last convolutional layer are 384, and the Output channels are also 256, Finally, add a pooling layer and select the MaxPool2d function, where kernel size is 3.The structural diagram of model 3 is shown in Figure 3:

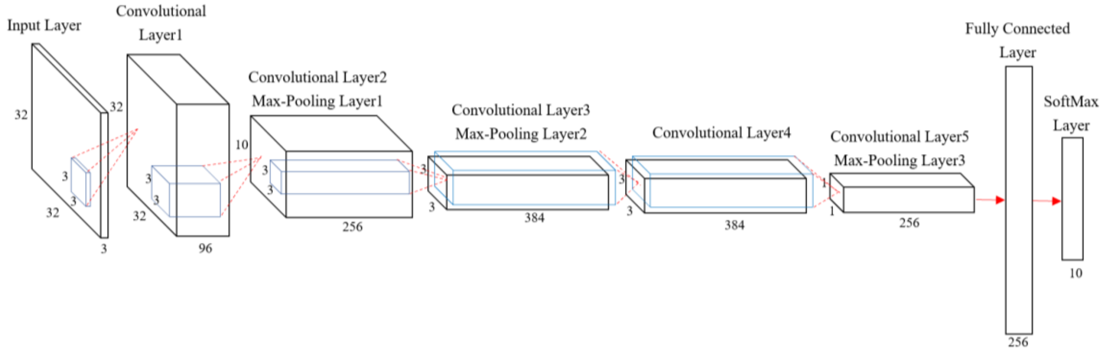


Figure 3. CNN Model 3 Structure Diagram

2.4. *Model Results and Comparison*

According to the above experimental results of different models that change the convolution layer (see Table 1), it can be seen that as the convolution layer continues to increase, the final calculated loss continues to decline, and the accuracy rate continues to improve. However, the running time of the program is also increasing, and GPU is also used in this experiment. Without the use of GPU, the program ran for more than 2 hours, which took too long to use. GPU was used in this experiment [10].

Therefore, convolutional neural networks should specifically select several layers of neural networks, not only considering the accuracy but also considering the time consumed. Improving accuracy while reducing time is still a worthwhile research direction.

Table 1. Prediction results of different convolution layer models

Model	Accuracy	Times	Loss
CNN_3	68.54%	11minutes	0.071
CNN_4	76.68%	15minutes	0.016
CNN_5	81.37%	23minutes	0.004

Changing the descent algorithm based on Model 3 can obtain experimental results (see Table 2 and Figure 4). Here, five different optimizers are selected [11], and it can be found that changing the optimization algorithm will change the accuracy and loss of the model. In addition, the accuracy rate of SGD is the lowest, while the accuracy rate of other optimizers is around 81%, with Adamax having the highest accuracy rate, up to 82.75%. By observing the Loss column, it can be seen that for most models, the lower the Loss value, the higher the accuracy rate. However, by observing Adadelta and SGD, it is found that the Loss values of both models are similar, with around 0.32, but the accuracy rate is significantly different. As shown in the figure, it can be clearly seen that different loss values continue to decline with the increase in training times. The declining trend of Adadelta and SGD loss values is basically the same, but when the epoch reaches about 10 times, it basically remains at 0.32 and no longer decreases. The remaining three optimizers maintain a steady downward trend.

Table 2. Prediction Results for Different Optimizer Models

Optimizer	Accuracy	Loss
Adam	81.37%	0.005
SGD	13.44%	0.328
Adamx	82.75%	0.001
Adadelta	82.01%	0.327
AdamW	81.68%	0.004

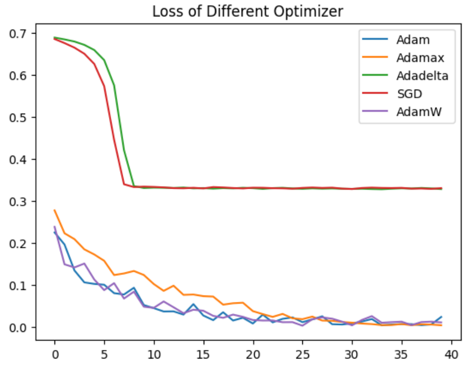


Figure 4. Loss of Different Optimizer (CNN_5)

3. Conclusion

This article first gives a bird's-eye view of the development of image classification and briefly introduces the factors that affect the accuracy of CNN. Next, this article conducted experiments and comparative analysis on the impact of convolution layers and optimizers on CNN accuracy. Three convolutional neural networks with different convolution layers were built, their structural design was introduced in detail, and their structural diagrams were given. Different models were used to classify images from the CIFAR 10 dataset. Finally, the accuracy results under different convolution levels and optimizers are given and compared. Research has found that as the volume layer continues to increase, the final calculated loss continues to decline, and the accuracy rate continues to improve. However, the running time of the program is also increasing, so there is still much room for improvement. In addition, it was also found that using a five-layer convolutional neural network for the CIFAR 10 dataset and using different optimizers can lead to changes in the accuracy rate, with Adam, AdamW, and Adamax having the highest accuracy rate and the best prediction effect.

In fact, the model still has many shortcomings. Firstly, with the increase of the convolutional layer, the running time of the program increases, which is an inevitable problem for many neural network models. Current solutions mostly use local or cloud GPUs. Another problem that has always been faced is the selection of optimizers, which is often compared through some experience and continuous parameter adjustment. Currently, there is no good method to select the appropriate optimizer and parameters for a specific dataset or model. These are all possible directions for future development, which are worth our in-depth consideration and exploration.

References

[1] LeCun Y, Bottou L, Bengio Y and Haffner P 1998 Gradient-based learning applied to document recognition. *Proceedings of the IEEE.* 86(11):2278-2323.

[2] Hinton GE and Salakhutdinov RR 2006 Jul Reducing the dimensionality of data with neural networks. *Science (New York, N.Y.)*.313(5786):504-507.

[3] Krizhevsky A, Sutskever I and Hinton G E 2012 ImageNet classification with deep convolutional neural networks. *Communications of the ACM, 60*, 84 - 90.

[4] Tripathy S and Singh R 2022 Convolutional Neural Network: An Overview and Application in Image Classification. *Advances in Intelligent Systems and Computing*.

[5] Chaganti S Y, Nanda I, Pandi K R, Prudhvith T G N R S N and Kumar N 2020 Image Classification using SVM and CNN, *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*, Gunupur, India, pp. 1-5.

[6] Nielsen M A 2015 Neural networks and deep learning[M]. San Francisco, CA, USA: Determination press.

[7] Kingma D and Ba J 2014 Adam: A Method for Stochastic Optimization[J]. *Computer Science.*

[8] Ruder S 2016 An overview of gradient descent optimization algorithms[J].

[9] Krizhevsky A 2009 Learning Multiple Layers of Features from Tiny Images.

[10] László E, Szolgay P and Nagy Z 2012 Analysis of a GPU based CNN implementation, 2012 13th International Workshop on Cellular Nanoscale Networks and their Applications, Turin, Italy, pp. 1-5.

[11] Ruder S 2016 An overview of gradient descent optimization algorithms.

Streaming process based on Spark and Kafka and its case application

Jiayi Fang

School of Computer and Information Engineering, Henan University of Economics and Law, Zhengzhou, Henan, 450000, China

631402070315@mails.cqjtu.edu.cn

Abstract. E-commerce platforms gradually integrate into our lives. Online shopping has become a daily habit for people. People who shop online will generate online shopping records in real time. These online shopping records are often massive. E-commerce platforms use relevant big data technology to conduct statistics on real-time user data. In response to the current demand for e-commerce platforms for big data real-time computing technology, this article provides an idea of big data streaming computing, a framework for combining two big data technologies, Spark and Kafka, to process streaming data. This framework uses Kafka to send real-time data to SparkStreaming for calculation. Then the obtained data is visualized. This article applies these two big data technologies to simulate and implement an application case of an e-commerce platform. This case design and implementation enables real-time statistics of the click volume of a mobile phone brand on an e-commerce platform and dynamic display of streaming data on the Web.

Keywords: Spark, Kafka, flow calculation.

1. Introduction

With the development of science and technology, the continuous integration of information technology into society has triggered an explosive growth of information. The background that online shopping has become the current mainstream shopping method. The overall trend of e-commerce platform transaction volume is rapid growth. This process will accumulate more user review data. These data reflect more product defect information and the actual needs of users for improving product functionality [1]. The data level may reach TB, PB, or even FB. Therefore, corresponding big data technologies are needed to store and process a large amount of information. Hadoop, as one of the earliest big data processing frameworks, can store and calculate massive amounts of data. The core of Hadoop design is Hadoop Distributed File System(HDFS) and MapReduce. Hadoop has the advantages of high reliability, high scalability, efficiency, and high fault tolerance. Compared to its advantages, the disadvantages of Hadoop are also very obvious. The MapReduce computing model is a dataset-based computing model. The data input and output method is to load data from physical storage, then operate the data, and finally write it to physical storage devices [2]. This disk-based computing model is suitable for handling most batch-processing tasks. The object corresponding to this batch processing method is generally static data. That is to say, a large amount of data can be processed in batches in sufficient time. However, if a large amount of data is calculated in real-time, such as viewing real-time clicks in the e-commerce field. So the response time obtained usually needs to be at the second level. This is where the above technology is no longer applicable. At this point, flow calculation can be adopted.

Stream computing is an important computing mode for big data. Unlike traditional batch computing based on determining the size of data, stream computing has the characteristics of unlimited data size, continuous, fast, and unordered data arrival, unstable data, and diverse data processing [3]. The process of streaming processing includes collection, calculation, and query. There are various streaming processing frameworks available for streaming processes, such as Storm, Flink, and so on. This article discusses Spark Streaming built on Spark. The data source of Spark Streaming can be obtained from Kafka. This article will introduce Spark and Kafka's stream processing frameworks and use them to simulate real-time click-through traffic of e-commerce platform products.

2. Techniques for building a stream process

This section will briefly introduce the Spark and Kafka technologies involved in the data processing process and the data visualization technology to be used after processing.

2.1. Spark

Spark is a general-purpose parallel computing framework similar to MapReduce that is open source by UC Berkeley AMP Lab, taking into account the characteristics of distributed parallel computing models and memory-based computing [4].

2.1.1. Features and applications of Spark. Spark is fast. Spark uses a combination of disk and memory to store intermediate results in memory without repeatedly dropping the disk. To support cyclic data and memory computing, Spark uses DAG to reduce secondary IO for shuffles and disks. In resource application and scheduling, Spark is based on coarse granularity, while MapReduce is based on fine granularity [2]. Spark will apply for all resources in advance, and there is no need to apply for resources when running tasks. Each task in Spark eliminates the time consumption of starting and destroying processes based on threads.

Spark has many flexible applications. Spark establishes multiple programming language APIs that can provide multiple language work environments for various developers and provides a shell environment for interactive queries. Spark can also provide various technologies, such as SQL queries, streaming computing, and machine learning libraries. It allows developers to implement stronger metafunctions.

Spark has more compatibility. Spark has multiple operating modes, which can be run independently or based on yarn. There is also a more flexible version of Mesos, which implements two forms of allocation: early allocation and on-demand allocation. It also runs based on multiple data sources, such as Hbase, Hive, HDFS, and so on. Spark can be applied to scenarios based on the above characteristics, such as complex batch data processing, the interactive query of historical data, real-time processing of data streams, and so on.

2.1.2. Spark ecology. The data processing process of big data technology is divided into several steps: integration, storage, distributed computing, analysis, and visualization. Spark is located in the stage of distributed computing.

The core of the Spark ecosystem is the Spark Core, which reads data from persistence layers such as HDFS, Amazon, S3, and HBASE. The job manager completes application calculations through the MESS, YARN, and Standalone instructions placed in Spark. These Spark applications can come from Spark Submit's batch processing, Spark Streaming's real-time stream count Data processing, interactive queries in Spark SQL, BlinkDB tradeoff queries, machine learning in Spark MLlib/MLbase, graph processing in Graphx, and mathematical calculations in SparkR [5].

2.1.3. Spark architecture. First, enter the program from the SparkContext created on the Driver side of Spark. During its initialization, a DAGScheduler and a TaskScheduler are created, respectively. DAG Scheduler divides job tasks into multiple stages. Then, each stage is divided into specific tasks. These tasks will be submitted to the task scheduler for task scheduling. The resources required during this process are requested from the cluster manager.

The values passed to different environment variables in SparkContext can lead to different operation modes. This allows Spark to run in local mode or pseudo-distributed mode, as well as the standalone mode that Spark comes with. Mesos or Yarn can also perform resource scheduling.

2.1.4. Resilient Distributed Datasets(RDD). As the core of the ecosystem, the core abstraction of Spark Core is RDD. RDD is a collection of read-only objects distributed in a cluster. Spark records a series of transformations that create RDDs. If a partition or part of the data of an RDD is lost, fault tolerance can be performed based on the reconstruction of its parent's RDD. This strategy is called lineage [6].

The operation of the RDD operator in Spark is divided into three steps: input, run, and output. Specifically, RDD is generated after reading data from a data source in memory. The generated RDD is then converted into a new RDD through various conversion operators. This can also cache intermediate results into memory. The job is then submitted using the Action operator. After data processing, it is stored in an external data space.

2.2. Spark streaming

Streaming computing processing also includes data collection, computation, and result presentation. But the entire process is real-time, which means improving the response requirements to the second level or even faster. The processed data objects also become data streams abstracted from unbounded data sets.

There are various frameworks for processing massive data streams. This article introduces the Sparkstreaming framework based on Spark. Spark Streaming is a streaming batch processing engine based on Spark, which can achieve high throughput and fault-tolerant real-time streaming data processing. It can seamlessly connect with RDD operators, machine learning, SparkSQL, and graphics and image processing frameworks [7].

2.2.1. Screaming structure. Spark Streaming is based on the API and memory computing provided by Spark, which splits the continuously input data stream into small batches for calculation and then generates new micro-batches. This is called incremental computation of data streams.

2.2.2. Discretized Stream(DStream). Spark Streaming uses an RDD sequence discretization data stream DStream as its data format.RDD is an abstract class, with DStream as an abstract subclass that implements functions. DStream separates data streams into RDDs and converts these RDDs into new RDDs.

2.2.3. Window operation. The concept of the time interval is not reflected in the above figure. Real-time refers to processing data in relatively short time intervals. However, from a microscopic perspective, these shorter time slices also include some RDDs processed in a shorter time. Before generating a new RDD, the processed RDD is aggregated within a time slice. This aggregation process is called a sliding window operation, and the window will slide according to the time slice.

2.3. Kafka

Apache Kafka is a distributed messaging middleware. It is a distributed publish-subscribe message system with high throughput, storing messages in a fault-tolerant manner. It has high performance, persistence, multi-copy backup, and horizontal scalability [8].

2.3.1. Kafka architecture. Before proceeding with the Sparkstreaming calculation process, data must first be collected. In e-commerce platforms, a large number of logs are generated in real-time every day. Kafka, similar to intermediaries, can be used to buffer data and provide it to the target system. Producers' messages are published to the primary agent of the partition, saved to the broker in Topics, and subscribed to by Consumers.

Producers serialize keys and values into byte arrays during the process of sending data. Kafka provides a flexible serialization process for transmitting different data types. The specific partition in the Topics to which the Producer sends messages can be specified. If not specified, the partition will be randomly selected. The agent will respond to the corresponding information after receiving the message. Kafka can also equip each partition with replicas to avoid proxy failures.

Consumers can read data from it. Consumers can read data from multiple agents simultaneously. The number of Consumers can also be multiple. The same group of Consumers forms a Consumer group. For feedback on Consumers' consumption of data, the submission offset can be used.

2.3.2. The combination of Spark and Kafka. On the Producer side, OGG For BigData transfers data to Kafka. On the Consumer side, Spark Streaming is used to continuously extract data from Kafka's agents for data processing and computation [7]. The process of pulling data has two modes:

The first mode is the Receiver mode. Use Kafka to continuously obtain and store data from the receiver's Zookeeper asynchronous thread. The reading time and offset can be configured through relevant parameters. After the Batch task is triggered, transfer the data to the remaining Executors for processing. The offset will be updated after processing is completed.

The second mode is the Director mode. This mode omits the above operation of connecting to the Zookeeper and directly reads data from Kafka. This is a regular query for the latest offsets from Kafka's themes and partitions. The data in each batch is processed within the defined offset range [8]. Let the Executor read the Partition data calculation.

2.4. Other technologies of the project

In this paper, Flask and Echarts technologies are used for data visualization.

2.4.1. Flask. Flask is a microframework developed based on Python and relying on the Jinja2 template rendering engine and Werkzeug WSGI routing service component as the core. It has good scalability and compatibility, which can help users quickly implement a website or Web service [9].

2.4.2. Apache ECharts(ECharts). There are a variety of tools for implementing data visualization. This article uses ECharts, an open-source JavaScript-based visualization chart library. It is very compatible with various browsers. It also has rich interactive functions. This allows it to highly customize data visualization charts [10].

3. Project design process

This article describes the process of building a stream using Spark, Kafka, and their combination. Now, a simple e-commerce case is simulated using the above technology.

In e-commerce, it is often necessary to make statistics on the real-time click volume of products. The statistical results can obtain the real-time popularity of the product and help make subsequent decisions. So a real-time click-counting system was created for certain products. Therefore, this article uses Spark and Kafka to simulate the real-time click volume of a different mobile phone brand on an e-commerce platform. The general process of the project is shown in Figure 1.

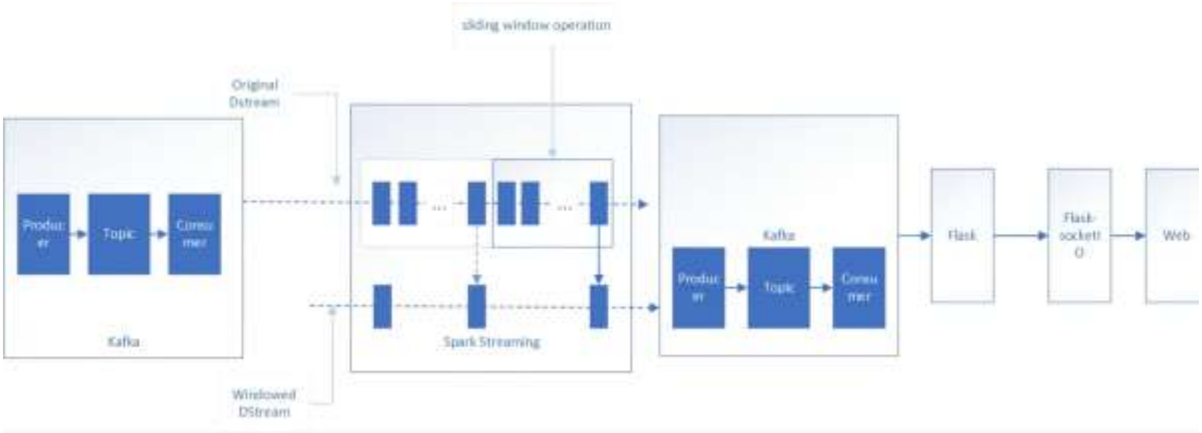


Figure 1. The general process of the project.

In Figure 1, the Kafka Producer sends the read user data to a topic. Spark Streaming then performs sliding window processing in the form of DStream from the mobile click data source formed by subscribing data from the first Kafka. The small blue rectangle above represents an original RDD in a time slice. Then the RDD enclosed by the sliding window is transformed to obtain a new RDD, which is the small blue box below. This is also the word frequency statistics of mobile phone brand clicks during window time. The DStream generated below is sent to a new topic in Kafka. Flask is used to build an application to subscribe to this topic. Flask Socket then pushes real-time phone clicks to the Web client for display.

3.1. Project process

Firstly, a random CSV file is generated using Python. This file contains click data for 10000 mobile phone brands. Then have Kafka Producers filter and read the relevant phone brand data in the CSV. The data is sent to the theme 'Mobile'. The data in the 'phone' is read and processed by Spark Streaming every two seconds in a sliding window. After processing, the processing results are sent to the new topic 'resultPhone'. Then the Flask Socket IO in Python is used to push real-time to the web client. The JavaScript library on the website is called to receive data and dynamically visualize it.

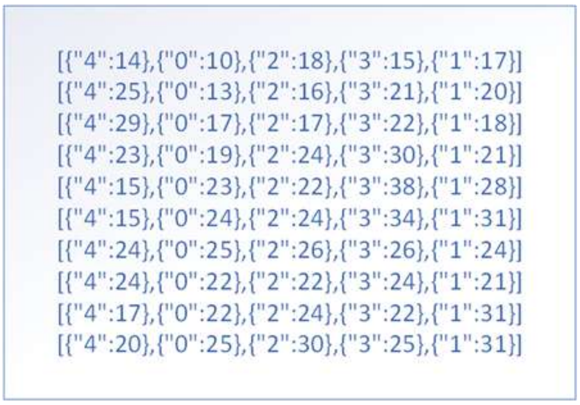
3.2. Project-specificity steps

The specific process of the project is divided into two parts: background data processing and data visualization.

3.2.1. Calculation of stream data. The phone entry in the above-generated CSV file includes natural numbers from 0 to 4. In ascending order, they represent five mobile phone brands: Huawei, Samsung, Apple, Oppo, and Xiaomi. A project was created in Pycharm. A Producer created in the project reads the numbers in the phone column and sends them to the 'Phone' topic.

Then an IDEA project 'KafkaParams' Consumer is created. It subscribed to the data stream in 'phone' in the project. Consumer-related parameters are configured. Then an instance of the SparkConf project, 'SparkConf', is created and set to local run mode. An instance 'StreamingContext' using SparkConf's stream project is created and specified to process data every 1 second.

The newly created 'KafkaParams' and' StreamingContext 'are combined together. A window reading data source lineMap is created. A 2s sliding window is set to read the Dstream data source sent every 1s. A new Producer is created. The result of calling the relevant operator for word frequency statistics is sent to the new Topic 'resultPhone'. The partial running results of the backend data calculation are shown in Figure 2.



```
[{"4":14},{"0":10},{"2":18},{"3":15},{"1":17}]
[{"4":25},{"0":13},{"2":16},{"3":21},{"1":20}]
[{"4":29},{"0":17},{"2":17},{"3":22},{"1":18}]
[{"4":23},{"0":19},{"2":24},{"3":30},{"1":21}]
[{"4":15},{"0":23},{"2":22},{"3":38},{"1":28}]
[{"4":15},{"0":24},{"2":24},{"3":34},{"1":31}]
[{"4":24},{"0":25},{"2":26},{"3":26},{"1":24}]
[{"4":24},{"0":22},{"2":22},{"3":24},{"1":21}]
[{"4":17},{"0":22},{"2":24},{"3":22},{"1":31}]
[{"4":20},{"0":25},{"2":30},{"3":25},{"1":31}]
```

Figure 2. Running results of the background data calculation section.

Figure 2 represents the word frequency statistics results in the new Topic in Josn format. Taking the first row as an example, the data in the first row represents the real-time click-through volume of Xiaomi brand phones in two seconds, which is 14 times, Huawei 10 times, Apple 18 times, Oppo 15 times, and Sunsumg 17 times. The remaining number of lines represents the same meaning. The number of console rows continues to increase in the continuous calculation of data streams.

3.2.2. Data visualization. Then a new Consumer created in Pychar is used to read word frequency statistics results in 'resultPhone'. Flask Socket is created to receive the results just read. The received results are pushed in real-time to the web client. The JavaScript library that is then called is used to write HTML display files. The real-time data pushed by SocketIO is visualized using the line chart. This chart shows the results of streaming processing. Dynamic data transformation is also performed every two seconds. The display effect is shown in Figure 3.

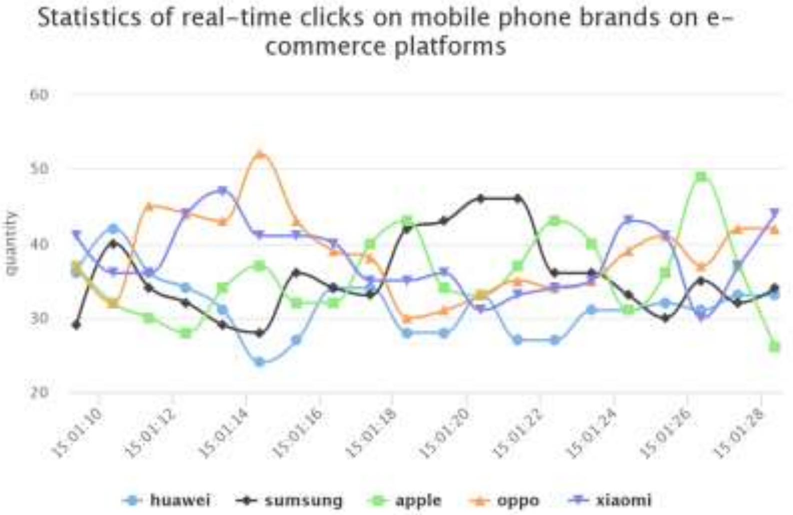


Figure 3. Data visualization of projects.

Figure 3 visualizes the statistical results of Spark Streaming. Display the real-time clicks of mobile phones of all brands in the current period in the form of a line chart, which changes every two seconds.

Figure 3 visualizes the statistical results of Spark Streaming. The results of real-time statistics show the real-time hits of mobile phones of all brands in the current period time in the form of a line chart. Figure 3 will also change every two seconds.

4. Conclusion

Therefore, this article introduces Sparks based on MapReduce optimization. It also introduces the functional framework and data format of Spark. Then it introduces the streaming computing framework in Spark that can meet real-time processing requirements. The data source for the calculation process is Kafka. Then this article simulates an application case where Kafka and Spark are combined on an e-commerce platform. In this case, this article introduces that the CSV file generated by data is produced by Kafka into a topic. The data stream generated by subscribing to this topic is sent to Sparkstreaming for calculation and then sent to a new topic for consumption. The consumption results are sent to the Web using a socket for data visualization.

However, in this experiment, there were flaws in the generation of data. Firstly, the data in this article are randomly generated using Python rather than generated by e-commerce users. The acquisition of real data requires corresponding interfaces on e-commerce platforms. Secondly, real e-commerce data does not directly generate the data form of this article. Before processing production data, complex logs generated by users in real time should first be converted into a standardized and readable format. Therefore, before using Kafka for production and consumption, the collected raw data can be processed first. The collected logs are processed using Flume. The processed results are then sent to Kafka. At this point, the data obtained by Kafka will be processed and displayed using the process mentioned in this article.

References

[1] Suo Hongsheng. Design and Research of Big Data Mining System Based on E-commerce Platform [J]. Internet Weekly, 2023 (06): 29-31

[2] Quan Zhaoheng, Li Jiadi. Innovation from Hadoop to Spark Technology [J]. Computer Knowledge and Technology, 2019, 15 (08): 265-268. DOI: 10.14004/j.cnki.ckt.2019.0823

[3] Meng Yunfei. Research on Key Technologies of Big Data Streaming Computing [J]. Heilongjiang Science, 2022,13 (14): 55-57

[4] Song Lingcheng. Comparative Analysis of Flink and Spark Streaming Streaming Computing Models [J]. Communication Technology, 2020,53 (01): 59-62

[5] Jiang Yongdu, Cheng Desheng, Zhao Zhiwu, Wang Li, Jiang Feng. Big Data Computing Platform Based on Spark Framework [J]. Network Security Technology and Application, 2020 (03): 65-66

[6] Wu Xindong, Ji Shengdong. Comparison of MapReduce and Spark for Big Data Analysis [J]. Journal of Software, 2018,29 (06): 1770-1791. DOI: 10.13328/j.cnki.jobs.005557

[7] Gao Zongbao, Liu Limei, Zhang Jiaming, Song Guoxing. Kafka Offset Reading Management and Design in the Park Platform [J]. Software, 2019,40 (07): 118-122

[8] Ye Huixian. The practice of Building a Data Center Processing Engine Based on Spark Streaming and Kafka [J]. Network Security Technology and Application, 2023 (03): 51-53

[9] Chen Jiafa, Huang Yujing. Application of Flask Framework in Data Visualization [J]. Fujian Computer, 2022,38 (12): 44-48. DOI: 10.16707/j.cnki.fjpc.2022.12.009

[10] Jing Guowei, Huang Dachi. Research on Data Visualization Based on ECharts [J]. Western Radio and Television, 2022,43 (20): 227-230+234

Comparison the effects of KNN and linear regression models in lung cancer prediction

Yi Zhou

Bishop Allen Academy, Ontario, M8Y 2T3, Canada.

zhouy014@tcdsb.ca

Abstract. Lung cancer has a range of major factors like smoking, yellow gingers, anxiety, etc. now, the problem of this research is that prediction for lung cancer. Prediction for lung cancer is a complex problem that is not suitable for human prediction. This research using a dataset was from Kaggle. There are 16 rows and 309 columns. To determine the k nearest neighbors (KNN) algorithm and linear regression algorithm, which one is better for prediction for lung cancer, and which coefficient will be best effective. This research uses mixed method research. In this work, when the K of the KNN algorithm equals 7 or 2, the effectiveness of the KNN model is best, when the alpha of the linear regression algorithm equals 20, the effectiveness of the linear regression model is best. The KNN model is better than the linear regression model, though the difference is negligible. In the future, more emphasis can be placed on using a wider range of algorithms or using more extensive and generalized dataset, as well as assessing the efficiency of the algorithm on larger datasets.

Keywords: lung cancer prediction, machine learning, KNN.

1. Introduction

Lung cancer, one of the most widespread and hazardous tumours in the world, has attracted attention for its relevance [1,2]. The principal factors linked to the occurrence of lung cancer involve smoking, chronic lung disease, among others [3]. The main treatment modalities for lung cancer include surgery, radiation therapy, chemotherapy, and immunotherapy [4].

Machine learning is the branch of artificial intelligence, it can find the formula and the law in the dataset. it will learn and improve in the large data. And then, they will be used to predict, classify, and make decisions.

Machine learning has a unique advance in in prediction problems in big data. It very easy that to process large amounts of data. For prediction of lung cancer that have utmost large, and many factors, machine learning is suited by. They can be based on large dataset, use quantitative analysis like mathematical statistics. There are more effective than human [5,6]. Human cannot use the large data, and quantitative analysis to predict the likelihood of the lung cancer.

It is important to research it for the health and safety of humans. In order to more batter for predicts the likelihood of one person who has the probability to get lung cancer in the future. Therefore, the research will discuss that two predict model about KNN model and linear regression model. They will be compared with the effect of those two models, and the effect of each different parameter of those models.

2. Method

2.1. Dataset

This research has been using the data of lung cancer in a website Kaggle that was updated by Mysar Ahmad Bhat. The dataset has 15 characteristic variables including gender (M/F), age, fatigue, anxiety, peer pressure, smoking, swallowing difficulty, chest pain, yellow gingers, wheezing, alcohol consuming, coughing, chronic disease, allergy, shortness of breath, and the label that whether has cancer (YES/NO). Then, the data has been pre-processed, by mapping discrete values to a specific number, such as mapping M to 1, F to 2, YES to 1 and NO to 2.

2.2. KNN model

K-Nearest Neighbors Algorithm: An Overview of the Procedure and Its Applications K-Nearest Neighbors is a popular machine learning strategy for classification and regression applications (KNN). It is a non-parametric, lazy learning approach that makes no assumptions about the underlying distribution of the data. [7].

The K-Nearest Neighbor (KNN) approach is a well-known supervised machine learning methodology for classification and regression applications. It is a simplistic algorithm that is easy to use, but it has some limitations, such as high computational cost and the curse of dimensionality [8].

It's more like a classifier that puts a label on an unknown thing. It's easy and easy to understand, but it is a lazy algorithm, which requires a large amount of memory. And it is computationally intensive and has low performance when classifying test samples; by the way it also has Poor interpretability. The results are highly logical and interpretable, but it is difficult to express highly complex data.

2.3. Linear regression model

Goodfellow et al. discuss linear regression as a simple and widely used method for regression tasks. They highlight its ability to a linear connection between the variables in the input and output, as well as its interpretability and ease of implementation. However, they also note that Linear regression is used to presume that input and output variables have a linear relationship, which may not hold in many real-world scenarios. Additionally, linear regression can be sensitive to outliers in the data and may not perform well in situations with high-dimensional input spaces [9].

Kutner et al. describe linear regression as a flexible and powerful tool for modelling relationships between variables. They note its ability to handle both quantitative and qualitative predictors and its simplicity in interpretation. However, they also discuss the potential for overfitting in linear regression models, particularly when the number of predictors is large relative to the sample size. They also note that in linear regression, input and output variables are believed to have a linear relationship, and that violations of this assumption can lead to poor model performance [10].

Using supervised machine learning, the linear regression technique can forecast continuous numerical variables. The algorithm establishes a linear model to describe the relationship between the independent and dependent variables. The advantages of linear regression are strong interpretability and generalization ability, but its disadvantage is that it may perform poorly on non-linear problems.

2.4. Evaluation metrics

This reach has use accuracy, precision, recall, F1 score. They are jointly used to measure the performances of the model. The accuracy represents the proportion of samples that were correctly predicted to all samples. Precision is the correct positive forecasts divided by all positive forecasts. The recall is the proportion of accurately predicted positive samples to all positive samples. The F1 score is the weighted harmonic mean of recall and accuracy.

3. Result

3.1. Comparison between KNN and linear regression

In this reach. The algorithm uses KNN model and linear regression model to predict the probability of lung cancer, as demonstrated in Table 1. Obviously, the effect of linear regression is better than another one.

Table 1. Comparison results between KNN and linear regression.

algorithm	accuracy	precision	Recall	F1_score
KNN	95.16%	98.31%	96.67%	97.48%
L-R	96.77%	98.33%	98.33%	98.33%

The advantage of KNN algorithm is that it is simple to use, flexible in handling outliers, and performs well when the training dataset is large. However, the KNN algorithm has a high computational complexity, leading to poor performance when processing large datasets. Additionally, because the KNN algorithm is an instance-based learning algorithm, it cannot abstract and generalize features, which may result in overfitting.

In comparison, a linear regression algorithm performs better in handling large-scale datasets due to its lower computational complexity. Additionally, the algorithm can abstract and generalize features to avoid overfitting. However, the linear regression algorithm struggles with outliers because it is based on the least squares method, which can be affected by outliers. Moreover, the performance of the algorithm may decrease when there are multiple features in the dataset.

Above all, KNN and linear regression algorithms have their respective downsides and benefits, as well as the selection of algorithm depending on the dataset’s characteristics and the requirements of the problem. In this study, KNN and linear regression algorithms are chosen for prediction because of the relatively small dataset size. For larger datasets, linear regression algorithms may be more suitable.

Next, the impact of k value in KNN algorithm and the regularization coefficient alpha in the linear regression algorithm are further explored.

3.2. Effectiveness of k in KNN models

While using the KNN algorithm, the choice of K has a large effect with the result of prediction. So, there are a range of experiments to research the impact of different k values on algorithm performance and the best one to predict.

There are 11 groups for the K that 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 be used to research what impact of each K value and which one is the best for algorithm performance as demonstrated in Table 2. During evaluation the same metrics are leveraged.

Table 2. Effectiveness of K in KNN.

K	Accuracy	Precision	Recall	F1 Score
1	91.9%	98.2%	93.3%	95.7%
2	96.8%	98.3%	98.3%	98.3%
3	95.2%	98.3%	96.7%	97.5%
4	96.0%	98.3%	98.3%	98.3%
5	95.2%	98.3%	96.7%	97.5%
6	96.8%	96.8%	100%	98.4%
7	96.8%	98.3%	98.3%	98.3%
8	96.8%	96.8%	100%	98.4%
9	96.8%	96.8%	100%	98.4%
10	96.8%	96.8%	100%	98.4%
11	96.8%	96.8%	100%	98.4%

This table shows that when k larger than 7 and equal 6, the recall be 1 is best, but the precision be lowest value. But the K equals 7 and 2, those 4 values are relatively high. So, the best one is 7 or 2.

3.3. *Effectiveness of alpha in linear regression models*

While the linear regression algorithm, the choice of the regularization parameter has a large effect with the result of prediction. So, there are a range of experiments to research the impact of different regularization parameters on algorithm performance and the best one to predict.

Table 3. Effectiveness of alpha in linear regression model.

alpha	Accuracy	Precision	Recall	F1 Score
0 .01	96.77%	98.33%	98.33%	98.33%
0 .02	96.77%	98.33%	98.33%	98.33%
0 .05	96.77%	98.33%	98.33%	98.33%
0 .1	96.77%	98.33%	98.33%	98.33%
0 .5	96.77%	98.33%	98.33%	98.33%
1	96.77%	98.33%	98.33%	98.33%
5	96.77%	98.33%	98.33%	98.33%
10	96.77%	98.33%	98.33%	98.33%
20	98.39%	98.36%	100%	99.17%
50	96.77%	96.77%	100%	98.36%
100	96.77%	96.77%	100%	98.36%

While alpha is less than 20, the accuracy, precision, recall, F1-score is the same. When alpha equals 20, the accuracy, precision, recall, F1-score is the largest. But over the value, those are lower, except recall be 1.so the best value is 20.

4. Conclusion

This research uses mixed method research to determine that the K equal 7 or 2 the KNN model is best effective, the alpha equal 20 the linear regression has best effective, each model's difference is negligible, however the KNN model is a little bit more effective than the linear regression model. This research uses KNN model and linear regression model with L2 regularized to predict the probability of lung cancer. Above all, while the K equals 6 or 2, the performance of the KNN model is best. And while the alpha equals 20, the performance of linear regression is best. But each model's difference was negligible.

In future research, lung cancer prediction can be achieved using other methods such as deep learning and neural network learning. Then test the accuracy and precision for them. Consider the advantages and disadvantages of each algorithm. Identify the most promising algorithms. Additionally, the data can be changed and added, making the dataset more extensive. including lung cancer data from multiple countries to increase the generalizability and applicability of the results. Further investigation, testing, and analysis should be conducted to improve the accuracy of the conclusions. By the way, the KNN algorithm and linear regression algorithm can be researched again in larger dataset, determining that each algorithm whither will be good for prediction, maybe KNN algorithm will be less effective, since the KNN algorithm is bad in too large dataset.

References

[1] Zhou, B., Zang, R., Zhang, M., Song, P., Liu, L., et, al. (2022). Worldwide burden and epidemiological trends of tracheal, bronchus, and lung cancer: A population-based study. EBioMedicine, 78, 103951.

[2] Mathur, P., Sathishkumar, K., Chaturvedi, M., Das, P., Sudarshan, K. L., et, al. (2020). Cancer statistics, 2020: report from national cancer registry programme, India. JCO global oncology, 6, 1063-1075.

[3] Sung, H., Ferlay, J., Siegel, R. L., Laversanne, M., Soerjomataram, I., Jemal, A., & Bray, F. (2021). Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA: a cancer journal for clinicians, 71(3), 209-249.

[4] Siegel, R. L., Miller, K. D., & Jemal, A. (2019). Cancer statistics, 2019. CA: a cancer journal for clinicians, 69(1), 7-34.

[5] Mahesh, B. (2020). Machine learning algorithms-a review. International Journal of Science and Research (IJSR).[Internet], 9, 381-386.

[6] Bell, J. (2022). What is machine learning?. Machine Learning and the City: Applications in Architecture and Urban Design, 207-216.

[7] Badillo, S., Banfai, B., Birzele, F., Davydov, I. I., Hutchinson, L., et, al. (2020). An introduction to machine learning. Clinical pharmacology & therapeutics, 107(4), 871-885.

[8] Chang, C. C., & Lin, C. J. (2011). LIBSVM: a library for support vector machines. ACM transactions on intelligent systems and technology (TIST), 2(3), 1-27.

[9] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444.

[10] Senter, H. F. (2008). Applied Linear Statistical Models . Michael H. Kutner, Christopher J. Nachtsheim, John Neter, and William Li. Journal of the American Statistical Association, 103, 880-880.

Proceedings of the 5th International Conference on Computing and Data Science
DOI: 10.54254/2755-2721/15/20230834

Research on the diagnosis of corona virus disease 2019(COVID-19) based on machine-learning

Fang Du

College of Physics and Information Engineering, Fuzhou University, Fuzhou, China.

031901302@fzu.edu.cn

Abstract. In recent years, the Corona Virus Disease 2019(COVID-19) has brought a huge impact on people's daily life and the normal operation of society, and as machine learning research deepens, this technology could help detect viruses such as the novel coronavirus. According to this, how to accurately, quickly and effectively analyze and classify laboratory results with multiple indicators through the method of machine learning is the object of this paper. In order to explore the classification performance of various machine learning classification algorithms on laboratory results, linear and non-linear methods were used respectively to analyze and classify the laboratory results. In linear analysis, distance discriminant and fisher discriminant were used to explore the classification effect of linear classification on laboratory results. The non-linear analysis mainly used Adaptive Boosting(AdaBoost) and Random Forest algorithm which are widely used to test the classification effect under the influence of multiple indexes. In this paper, python and other tools were used to classify samples of different combinations by using the idea of cross validation. By comparing the running time and detection accuracy, it was found that Adaboost algorithm is applicable in most cases and was a relatively fast and accurate classification method. In addition, Random Forest algorithm had similar accuracy, but it might have better performance on a large data set.

Keywords: machine-learning, fisher discriminant, index discriminant, random forest, COVID-19.

1. Introduction

Machine learning, as a multidisciplinary subject, uses computers as the tools to simulate human learning behavior in real time, and uses knowledge structures to divide the content to effectively improve learning efficiency [1]. With the deepening of people's learning and development in the field of artificial intelligence, machine learning has been applied more deeply in the fields of knowledge-based systems, natural language reasoning, machine vision and pattern recognition. In recent years, as a new strain of coronavirus that has never been found in humans before, Corona Virus Disease 2019(COVID-19) has become a member of coronaviruses. Because of the rapid spread of the virus, diversity of transmission routes and the cause of a wide range of serious illnesses, COVID-19 has caused a major impact on people’s daily lives and work, while also bring the severe challenge to the world-wild health system. For this virus, medical laboratory tests play an important role in assisting doctors to diagnose and clustering and discriminant algorithms in machine learning has provided scientific means of judgement.

In the current research, machine learning has been widely used in drug composition analysis and pathological analysis, such as Hui Xie et al. used Random Forest model, logistic regression model, Adaptive Boosting(AdaBoost), Gaussian Naive Bayes(GaussianNB) and other machine learning methods in the classification model to analyze the proportion of immune cell infiltration in pancreatic cancer [2]. Yixiao Zhai used the Random Forest classification method to classify antioxidant proteins [3]. Bin Tian et al. used a variety of machine learning classification methods such as decision tree, Random Forest, K-Nearest Neighbor(Knn) and support vector machine to analyze the feature recognition of lung images in COVID-19 [4]. Machine learning has a wide range of application and prospects in medicine composition analysis and pathological analysis.

The main task of this paper is to use the discriminant method in machine learning to find the relationship among different factors and judge whether the patient is infected with the COVID-19 according to the data from different cases. By comparing Fisher discriminant, Random Forest and other methods, we can select the best-matched and the highest-speed one and it will help people to judge the unknown virus with a better method.

2. Method

2.1. Problem description

Machine learning is widely used in component analysis and category classification. Among them, the component analysis and result judgement of medical laboratory results are closely related to the current world. In this highly intelligent era, people need to have some better method to classify some unknown things, such as virus and products, to help us find the factors we need and the results of judgment in the huge amounts of data. This paper mainly introduces the principles of clustering, linear discrimination algorithms and their applications in COVID-19 detection. The main analysis systems used are python and matlab.

2.2. Data collection

Since there are few data sets about the medical laboratory test results and detail of COVID-19, the data in this paper comes from a data set about the COVID-19 assay results provided in a Mathematical Contest in Modeling competition. Its main content is the laboratory results of 60 different patients. Each patient's laboratory results contain the patient's medical record number and seven test indicators. The first 30 patients are confirmed cases and the last 30 are healthy. Our goal is to choose the method with the highest accuracy and the fastest computing speed as the model to judge whether a certain person is infected by the COVID-19. The software and platforms used in this project are Pycharm (version 2022.3.2) and MATLAB.

2.3. Data analysis

In order to observe the distribution and characteristics of data in the data set more directly and prepare for subsequent data classification and analysis, using a line chart to display the data set is necessary. It’s not difficult to find from the statistical graph drawn by the seven indicators that the data of the diagnosed and health people fluctuated greatly in chart 1 and chart 7. In chart 2, chart 3, chart 5 and chart 6, there was no significant difference in the detection results between the two groups except for several peaks (Figure 1). In chart 4, there was a significant difference between diagnosed people and healthy people. These chart and analysis may help with the initial setting of some hyperparameters.

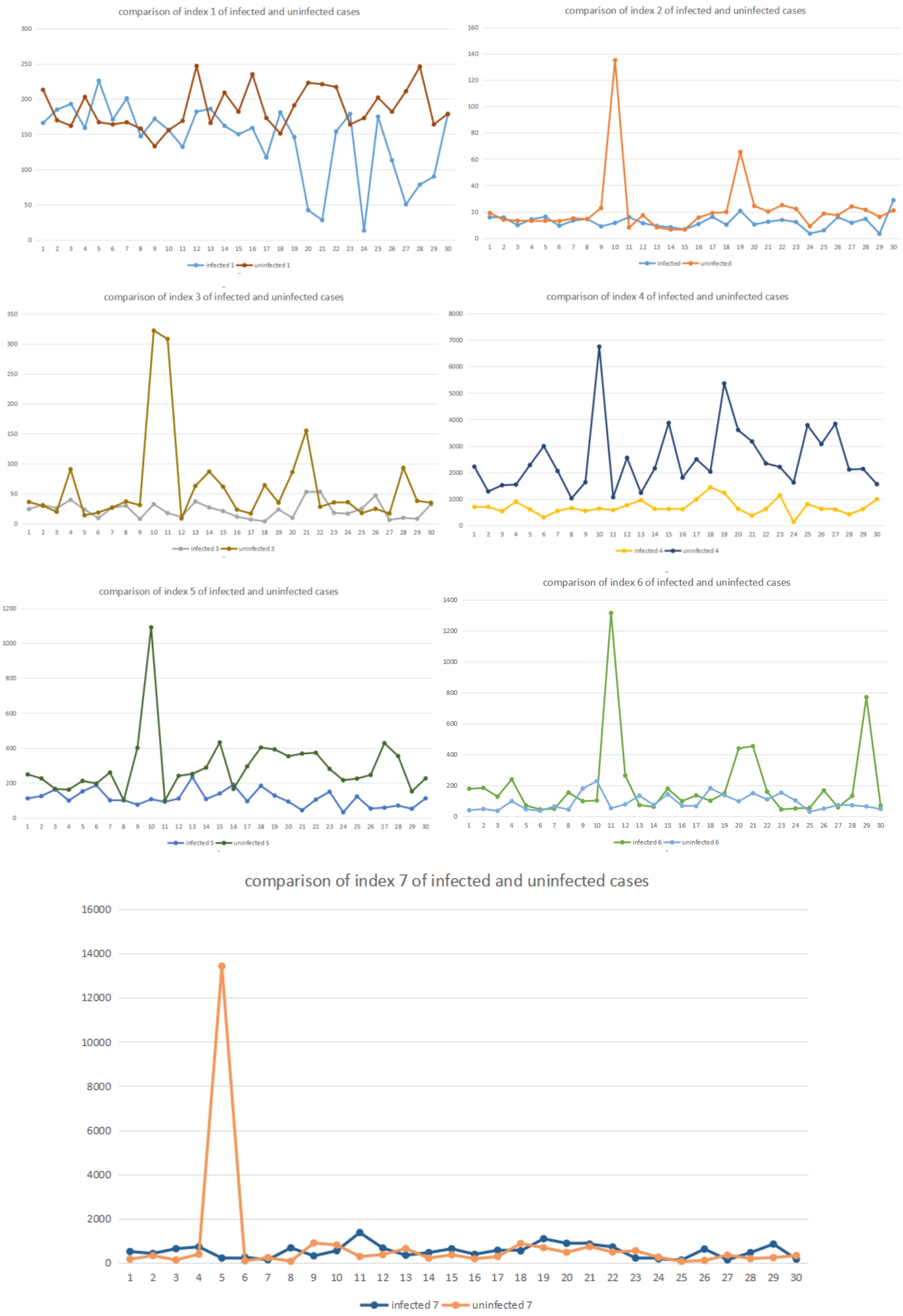


Figure 1. Seven indicators affecting COVID-19 in the data set.

2.4. Technical approach

2.4.1. Distance discriminant.

Distance discriminant is a classification method by calculating the distance between the center of the origin point and the surrounding categories. The basic idea of this algorithm is to calculate the distance between the point to be measured and various categories, and select the category with the shortest distance from the point as the point's classification.

Compared with other algorithms, Distance discriminant is simpler and easier to observe. This method is suitable for discriminating random variables with continuous distribution and is applicable to almost all probability distributions of variables. In Distance discriminant, Euclidean Distance discriminant and Mahalanobis Distance discriminant are usually used.

Euclidean Distance discriminant: If x, y are two points in n-dimensional space, the Euclidean distance between x and y is:

$$\mathrm{d(x,y)} = \| \mathrm{x-y} \|_2 = \sqrt{\mathrm{(x-y)^T(x-y)}} \tag{1}$$

Mahalanobis Distance discriminant: If x and y are two samples taken from a population X with a mean of μ and a covariance matrix of Σ, the Mahalanobis distance between x and y is:

$$\mathrm{d(x,y)} = \sqrt{\mathrm{(x-y)^T\Sigma^{-1}(x-y)}} \tag{2}$$

In this equation, x, y, μ are all vectors. According to the Distance discriminant, if the distance (Euclidean distance or Mahalanobis distance) between the point x and the categories G1 and G2 is d1 and d2 respectively, we can discriminate it with:

$$\begin{cases} x \in G1, d1 < d2 \\ x \in G2, d1 > d2 \end{cases} \tag{3}$$

The algorithm process of Distance discriminant is given in Table 1.

Table 1. Distance discriminant.

Algorithm 1: Distance discriminant
1. Select the sample which is used for test from the sample set
2. Calculate the distance between the sample and each center of categories
3. Choose the category which has the shortest distance from the sample is selected as the classification of the sample

2.4.2. Fisher discriminant.

Fisher discriminant is a classification method of machine learning which select an appropriate projection direction and project the sample in this direction to make the projected sample points be separated as far as possible. This method can effectively reduce the dimension of data and maintain the necessary characteristics of sample set.

Suppose the set of training samples is $\mathrm{X} = \{\mathrm{x_1, x_2, \cdots, x_N}\}$ and each sample is a d-dimensional vector. The sample of class $\mathrm{W_1}$ is $\mathrm{X_1} = \{\mathrm{x_1^1, x_2^1, \cdots, x_{N_1}^1}\}$, the sample of class $\mathrm{W_2}$ is $\mathrm{X_2} = \{\mathrm{x_1^2, x_2^2, \cdots, x_{N_2}^2}\}$. the next step in this algorithm is to find a projection direction:

$$y_i = w^T x_i \, , i = 1,2, \cdots, N \tag{4}$$

By reducing the dimension of the sample data to one dimension and using the Fisher criterion (Rayleigh quotient), the projection direction is discriminated to be the best projection direction:

$$\max \ \mathrm{J_F(w)} = \frac{\widetilde{\mathrm{S_b}}}{\widetilde{\mathrm{S_w}}} = \frac{(\widetilde{\mathrm{m_1}}-\widetilde{\mathrm{m_2}})^2}{\widetilde{\mathrm{S_1}}^2+\widetilde{\mathrm{S_2}}^2} \tag{5}$$

In this equation, $\widetilde{\mathrm{S_b}}$ is the between-class scatter after projecting and $\widetilde{\mathrm{S_w}}$ is the within-class scatter. $\widetilde{\mathrm{m_1}}$ and $\widetilde{\mathrm{m_2}}$ are the mean vectors after projecting. $\widetilde{\mathrm{S_1}}$ and $\widetilde{\mathrm{S_2}}$ are the within-class scatter matrix [5].

The algorithm process of Fisher discriminant is given in Table 2.

Table 2. Fisher discriminant.

Algorithm 2: Fisher discriminant
1. Use the known sample observation matrix to calculate the sample mean vector of each population $\overline{x}^{(i)}$ and total mean vector of each population $\overline{x}$ 2. Calculate the between-class scatter matrix and the within-class scatter matrix respectively 3. Use the within-class scatter matrix and the between-class scatter matrix to search for the projection vector u and minimize the within-class distance and maximum the between-class distance 4. Discriminate the distance between the sample points and the categories on the projection, choose the closest category as the classification

2.4.3. Random Forest algorithm.

Random Forest algorithm is an algorithm that integrates multiple trees through the idea of ensemble learning. This method uses the decision tree as the basic learner and work effectively on classification and regression.

Decision tree is an important classification and regression method in machine learning and data mining. It is a model which use a tree-like structure to represent the predictive analysis. It is constructed in a recursive order from root to leaves and divide the sample into different subsets through selecting the main features. If these subsets can be classified correctly, the leaf nodes are built. Repeat the above steps until each subset is divided into leaves. The decision tree will be constructed at last.

In the process if decision tree construction, information entropy is the most common index to measure the sample set. Suppose that the proportion of class k in the current sample set D is $\mathrm{p_k}$:

$$\mathrm{Ent(D)} = -\sum_{\mathrm{k=1}}^{|\mathrm{y}|} \mathrm{p_k \log_2 p_k} \tag{6}$$

The purity of D increased with decreasing Ent(D) value. after that, Use the information entropy to calculate the information gain of each index:

$$\mathrm{Gain(D, index)} = \mathrm{Ent(D)} - \sum_{\mathrm{v=1}}^{\mathrm{V}} \frac{|\mathrm{D^v}|}{|\mathrm{D}|} \mathrm{Ent(D^v)} \tag{7}$$

By comparing the information gain, the appropriate feature dividing can be selected [6,7].

In addition to information gain, Gini coefficient is often used to divide decision trees in Random Forest. It's because the Gini coefficient reflects the probability that two randomly selected samples of D are inconsistent. The Gini coefficient is calculated by:

$$\mathrm{Gini(D)} = \sum_{\mathrm{k=1}}^{|\mathrm{y}|} \sum_{\mathrm{k' \neq k}} \mathrm{p_k p_k}' = 1 - \sum_{\mathrm{k=1}}^{|\mathrm{y}|} \mathrm{p_k^2} \tag{8}$$

$$\mathrm{Gini\_index(D, index)} = \sum_{\mathrm{v=1}}^{\mathrm{V}} \frac{|\mathrm{D^v}|}{|\mathrm{D}|} \mathrm{Gini(D^v)} \tag{9}$$

$\mathrm{P_k}$ is the proportion of class k in the current sample set D, V is the number of the indexes.

The coefficient used in this paper is the Gini coefficient. The algorithm process of Random Forest algorithm is given in Table 3.

Table 3. Random forest algorithm.

```
Algorithm 3: Random Forest algorithm
    1. Randomly extract m samples from the original data set with the returned samples to generate m training set
    2. Use the training set to train m decision tree models
    3. For the decision tree model in random forest, the best features are selected to partition the data set by comparing the information gain or Gini coefficient.
    4. Use the generated decision trees to construct a random forest and final classification of the test samples is decided by voting according to the multiple trees
Decision tree algorithm:
    1. Calculate the information entropy of the root node
    2. Calculate the information entropy of each index and select the index with maximum information entropy as the dividing index
    3. Reduce over-fitting risk by pruning
    4. Repeat the preceding steps for the divided child nodes until no further dividing is possible
```

2.4.4. AdaBoost algorithm. AdaBoost algorithm is an improvement on Boosting algorithm and the way it is used to train weak learners is to train with all the data in the data set. The training samples will be given a weight again in each iteration and a more effective classifier will be constructed on the basis of the last weak learner. By increasing the weight of misclassification, the model pays more attention to misclassified samples and predicts the final result through voting.

In this method, the method of updating new weight ω′ is [8,9]:

$$\varepsilon = \omega \times (\widehat{y_\imath} == y) \tag{10}$$

$$\alpha_j = 0.5 \times log\frac{1-\varepsilon}{\varepsilon} \tag{11}$$

$$\omega' = \omega \times e^{(-\alpha_j \times \widehat{y_\imath} \times y)} \tag{12}$$

The algorithm process of AdaBoost algorithm is given in Table 4.

Table 4. AdaBoost algorithm.

```
Algorithm 4: AdaBoost Algorithm
    1. Initialize the sample weights that have the same initial value, and apply the constraint that the sum of the sample weights is 1
    2. In the m boosting, do the step 3 to 5 for the j boosting
    3. Train a weak leaner with a weight： $C(j) = train(X, y, \omega)$
    4. Predicted the sample and calculate the error rate of the weight
    5. Calculate the parameter, update and normalize the weights
    6. Complete the final forecast
```

3. Experimental procedure

3.1. Data selection and partitioning

In this work, the data set used had 7 indicators and had divided into two different labels. This study expects training and testing different algorithms by dividing the data set into training and test data sets. In this experiment, Distance discriminant, Fisher discriminant, Random Forest algorithm and AdaBoost algorithm were used to classify the test data set. The evaluation metrics are the time used to process the data and the accuracy of the classification. There are 30 infected and 30 uninfected cases in the total data set. In this experiment, 10 data will be randomly selected form each of the infected and uninfected cases and these 20 data will be used as the test data set. The remaining data will be used as training data set to train the algorithm model.

In experiment, random data set partitioning was repeated 20 times, and the average accuracy of classification and the average running time were used as evaluation metrics for various algorithms.

3.2. Results

The experiment repeated the classification process several times, only the classification results of the first three times in 20 simulations of various classification algorithms are presented here. According to the results, compared with linear algorithm, AdaBoost algorithm and Random Forest algorithm consumed more time but had higher accuracy. In the following results, T means the classification is correct and F means the classification is wrong, as illustrate in Table 5 to table 8.

The results of Distance discriminant are shown in the table 5:

Table 5. Classification results of test data sets by distance discriminant.

Test sample	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
test 1	T	T	T	T	T	T	T	T	T	T	F	F	T	T	T	T	T	T	T	T
test 2	T	T	T	T	T	T	T	T	T	T	T	T	F	T	T	T	F	T	F	T
test 3	T	T	T	T	T	T	T	T	T	T	T	T	T	F	T	T	F	T	T	F

The results of Fisher discriminant are shown in the table 6:

Table 6. Classification results of test data sets by Fisher discriminant.

Test sample	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
test 1	T	T	T	T	T	T	T	T	T	T	T	F	F	T	T	T	T	T	T	T
test 2	T	T	T	T	T	T	T	T	T	T	T	T	F	F	T	T	T	T	F	T
test 3	T	T	T	T	T	T	T	T	T	T	T	T	T	F	T	T	T	F	T	F

The result of Random Forest algorithm is shown in the table 7:

Table 7. Classification results of test data sets by random forest algorithm.

Test sample	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
test 1	T	T	T	T	T	T	T	F	T	T	T	T	T	T	T	T	T	T	T	T
test 2	T	T	T	T	T	T	T	F	T	T	T	T	T	T	T	T	T	T	T	T
test 3	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T

The results of Adaboost algorithm are shown in the table 8:

Table 8. Classification results of test data sets by AdaBoost algorithm.

Test sample	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
test 1	T	F	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T
test 2	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T
test 3	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T

3.3. Result analysis

Finally, the accuracy and running time of the four algorithms are summarized in Table 9 and Table 10, Figure 2 and Figure 3. The results are analyzed through horizontal comparison of the accuracy and running time of different algorithms.

Table 9. Comparison of accuracy of several algorithms.

Algorithm	Distance discriminant	Fisher discriminant	Random Forest	AdaBoost
accuracy rate	0.8525	0.8775	0.9175	0.9300

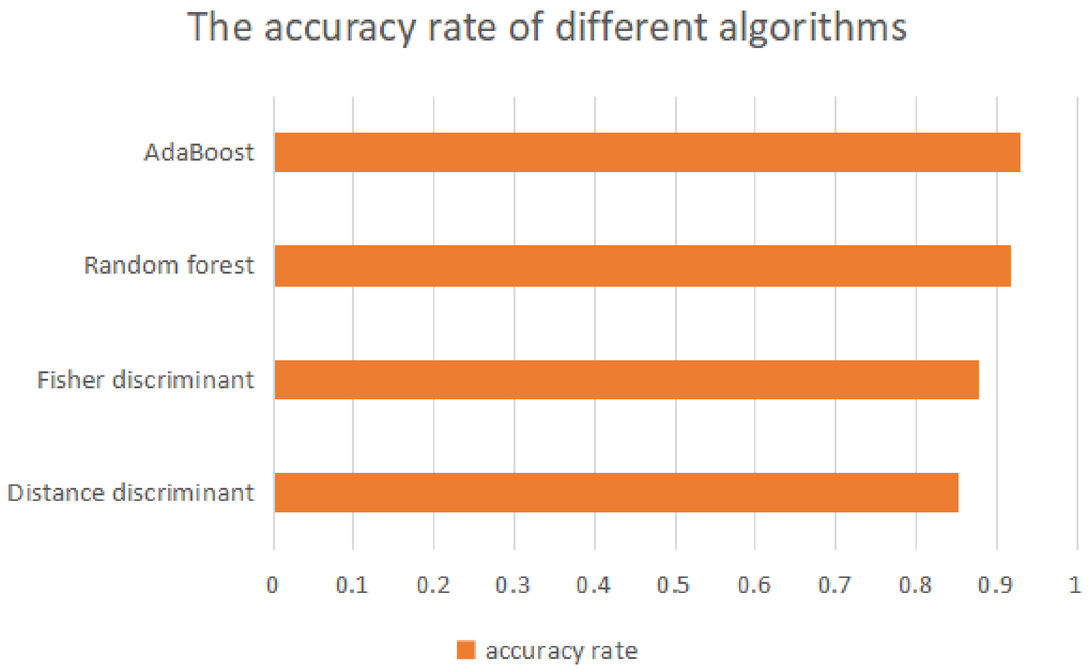


Figure 2. Comparison of accuracy of several algorithms.

By comparing the accuracy of several algorithms, it can be found that among various classification methods, AdaBoost algorithm and Random Forest algorithm are the ones with the highest accuracy. The specific accuracy rate sorting is: AdaBoost > Random Forest > Fisher discriminant > Distance discriminant. According to the above analysis of 7 different indicators, some indicators of infected and uninfected samples overlapped a lot. The overlap in the sample space is also the main reason for the poor classification effect of Distance discriminant and Fisher discriminant on this data set. Compared with AdaBoost algorithm, Random Forest algorithm is more suitable for a large-sample data set. The total amount of data contained in this sample data set is limited, which can not show the advantages of Random Forest algorithm. Therefore AdaBoost learning based on false classification has the best partition accuracy on this data set.

Table 10. Comparison of running time of several methods

Algorithm	Distance discriminant	Fisher discriminant	Random Forest	AdaBoost
running time (s)	0.000302	0.000750	0.012018	0.001594

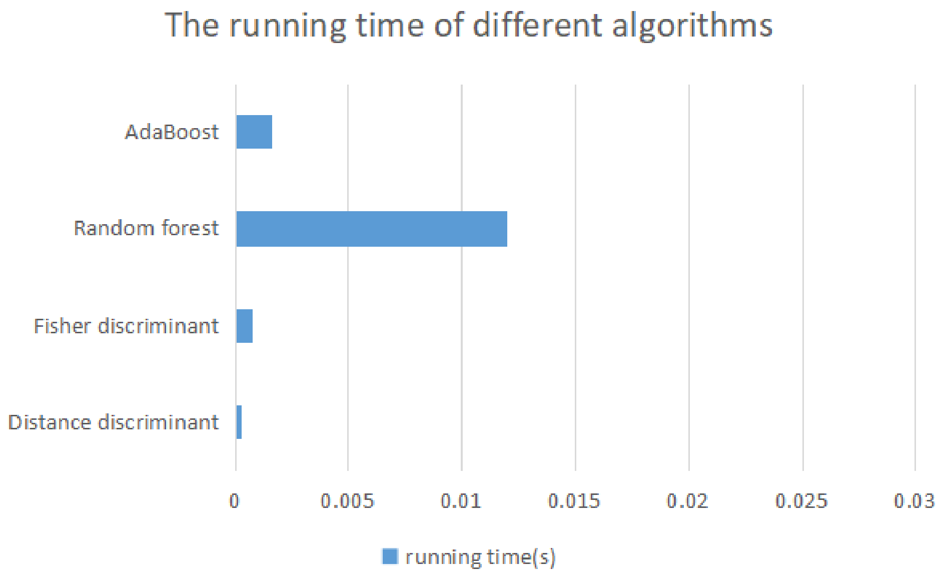


Figure 3. Comparison of running time of several methods.

By comparing the time cost of several algorithms running on the same size sample set, it can be found that the running time is sorted as follows: Random Forest > AdaBoost >Fisher discriminant > Distance discriminant. Random Forest algorithm takes the longest time, because it can accurately run on high_dimensional large data set. However, the space and time occupied by the model training through the training set will increase with the increase of the number of decision trees. In this experiment, the total data set isn't very large, so the time consumed by AdaBoost and other algorithms is relatively small compared with the Random Forest algorithm [10].

In the remaining three algorithms, Distance discriminant is the fastest because of the smallest computation required and it is the same for Fisher discriminant. Generally, AdaBoost costs more time, but in this experiment, it is not obvious due to the small data set.

4. Conclusion

In this paper, linear, non-linear and other machine learning classification algorithm were used in different combinations of COVID-19 test cases, and the idea of cross validation was applied to realize the classification of multi-index results of novel coronavirus test samples. In this study, multiple test indicators of the novel coronavirus test results were analyzed to determine whether a case was infected or not. In the process of testing, this study believes that the AdaBoost model can accurately determine whether a patient is infected with the novel coronavirus through a serious of case indicators in most case, which is a more accurate and rapid classification method. On the other hand, Random Forest algorithm can also achieve similar classification accuracy as AdaBoost, but it cannot work effectively as AdaBoost in data sets with a large amount of data. Therefore, it is more suitable for use in data sets with a large amount of data.

In future studies, algorithms combining various classification algorithms such as AdaBoost and Random Forest will be considered for processing and experiments on data sets with large data and more abundant indicators. In addition, the combination of images, image feature recognition and detection can be applied to the classification and judgment of more medical detection results.

References

[1] Haopeng Li. Intelligent robot exploration based on machine learning method. 2019, Telecom World, **26(4)**:241-242.

[2] Hui Xie, Zuliang Deng. Pancreatic cancer patients survival Value analysis of machine-learn-based immune cell infiltration classification model in predicting survival. 2021 *Journal of Xiangnan University (Medical Sciences)*, **23(04)**:19-27.

[3] Yixiao Zhai. Classification of antioxidant proteins based on machine learning and sequence information. 2022 *Northeast Forestry University*.

[4] Bin Tian,Hui Yu,Jigang Ren et al. Effectiveness of multiple classification models based on machine learning in the differential diagnosis of novel coronavirus pneumonia and community acquired pneumonia. 2021, *Radiologic Practice*, **36(05)**:590-595.

[5] DeeGLMath. Fisher linear discriminant analysis. https://blog.csdn.net/linjing_zyq/ article/details/ 120515566

[6] Mi Tu. Random Forest algorithm. https://blog.csdn.net/m0_46926492/article/details/122798056

[7] VernonJsn. Decision tree algorithm. https://blog.csdn.net/qingxiao__123456789/article/ details/122530376

[8] Xiu lian zhi lu. Detailed introduction about the AdaBoost algorithm https://blog.csdn. net/sinat_29957455/article/details/79810188

[9] Yoav Freund, Robert E Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. 1997, *Journal of Computer and System Sciences*. **55, 1** 119-139.

[10] Jiashilin. Advantages and disadvantages of Random Forests. https://blog.csdn.net /qq_35290785/article/details/100561148

Proceedings of the 5th International Conference on Computing and Data Science
DOI: 10.54254/2755-2721/15/20230835

Homography transform enhance CNN prediction accuracy on image classification

Chang Li

University of Illinois Urbana champaign, Lincoln Hall, 702 S Wright St, Urbana, IL 61801, USA

changli8@illinois.edu

Abstract. Convolutional Neural Network (CNN) image classification is a well-established algorithm that has been implemented in many fields. Benefit from the digitalization process and the exponential increase in the base of smart devices, this algorithm can be applied to even more traditional or casual contexts in the future driven by the trend of Internet of Things. Thus, the needs for optimizing image classification in specific domains may have turned out to be ongoing valuable research direction. This paper focuses on providing optimization under one example context which is using traffic signs as the experimental target. For the randomly selected traffic sign samples in the experiment, the accuracy obtained from the samples treated by the homography transform compared to control group passed all three statistical tests: Two Sample t-test, McNemar's test, and Fisher's exact test. Therefore, research direction has achieved small-scale validation and presents optimism for large-sample experiments and further research in optimization using the introduced strategy in the paper.

Keywords: CNN prediction, homography, image classification, autopilot.

1. Introduction

Machine learning algorithms have been utilized for image classification in various fields [1-4]. For instance, in agriculture, the work of Dingle Robertson and King, who employed the k-Nearest Neighbor (kNN) algorithm to classify various types of agricultural land cover using Landsat-5 TM imagery [5]. Furthermore, Zhang et al. conducted a study that aimed to classify crops in precision agriculture applications using hyperspectral imagery. The study utilized a Convolutional Neural Network (CNN) and achieved high accuracy in crop classification [6]. The authors concluded that CNNs hold potential in the field of crop classification for precision agriculture. This study highlights the effectiveness of deep learning algorithms in image classification tasks, especially in the agricultural sector where accurate classification can have significant economic and environmental impacts. On the other hand, autopilot and smart car have been very popular in the past few years, and the capital markets have given recognition by substantial investing, such as Tesla's stock price surge and the fund investment in Autopilot &smart cars by various large volume & established car manufacturing companies. In fact, engineers were already working on cameras in cars before autonomous driving became popular [7-9]. Tracing back to 1956 there were already applications for back-up cameras on cars [9]. However most rear-facing cameras installed in vehicles only provide perspective projection images, which has the incompetence of providing the distance between the vehicle and obstacles behind it. For example, Lin,

C. C., & Wang, M. S. in their paper have studied on using utilizing homography transform to warp bird's eye view captured by car camera; they called TVTM approach [8].

However, autopilot is still undergoing some obstacles. According to Liu, Y, current driverless cars can only support simple traffic conditions and that it is difficult to make correct judgments in more complex traffic situations, which could lead to accidents [7]. Under this context, the author foresees that car recognition of road conditions may have long-term optimizing needs. Autopilot is designed to collect information in real-time from the surrounding area while driving and turn it into data that matches traffic rules, just like a human driver, so correctly identifying traffic signs is a suitable entry point for the purpose of this paper.

The author finds that the homography transform share some characteristic that may help classic CNN image classification algorithm with its shortfall. Thus, this paper will apply this strategy by feeding CNN with homography transformed image and evaluate if it is a promising optimization related to autopilot technique.

2. Method

This paper mainly studies and explores whether using homography transform can directly improve the accuracy of the original classical CNN image classification.

2.1. Convolutional Neural Network

CNN is the most adopted algorithm on solving image classification problem. The crucial property of CNNs is that it automatically learns and extracts features from raw input data through a series of convolutional layers, pooling layers, and fully connected layers. The two main factors of this process are filters and pooling layers. The filter step for CNN is used to compress the input image matrix:

1) Define a window matrix with size less than the input image matrix, normally 5x5 or 7x7.
2) Fill the window matrix with certain value based on the needs for current filter.
3) Starting from the upper left corner of the input matrix, multiple the original matrix that is under the window matrix with the filter matrix result in one value.
4) Iteratively pivot the window matrix until all possible window cut of original matrix has met.

Based on the mechanism of the filter step, it is noticeable that while window pivots, the data at the edges of the matrix will be used significantly less times than the data in the middle ones. Consider the following sample diagram, black matrix is the 5x5 input image matrix and colored rectangles are the 3x3 window matrix. It is clear the upper left position is only used to calculate the upper left one box for the result by the red upper left window matrix (Figure 1). Any other pivots will not use upper left position.

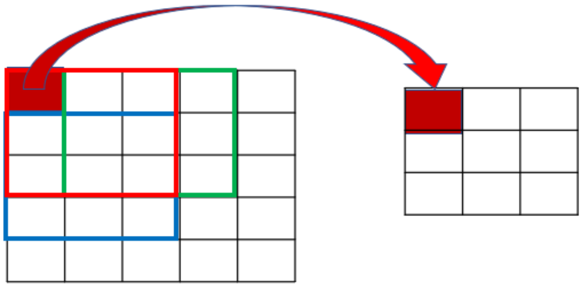


Figure 1. Filter for corner pixel.

On the contrary, the most middle position (center) of the input matrix will join the calculation by every pivot window matrix for this sample diagram. Therefore, every position of the result compressed matrix is related to the original center matrix (Figure 2).

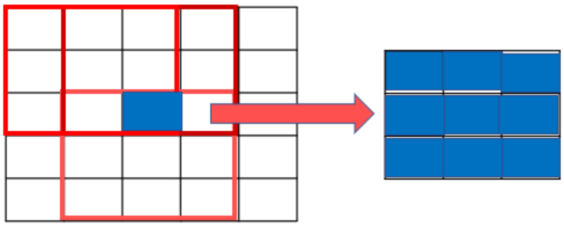


Figure 2. Filter for middle pixel.

From that observation, this paper deduces that after multiple filters when the model stacks up these layers together, the model would contain more information originally from the middle and less surrounding edges area of the input image matrix.

Although padding is an effective solution for this issue, the trade off to apply it may need be considered in the business operation aspect. Padding will expand extra pixels based on the size of filter and window size in order to offset the information loss after filters. However, this approach will increase computational cost and memory requirements of model and slow down the training speed. Even if the company manages to exclude the cost problem, it is also at risk of overfitting when the padding is too large.

Follow by this implication, the author states that it is possible to use homography transform to warp an image from a side angle to a front angle could make the CNN model better recognize the input since the front angle of a target provides more information around the center of the standard target image.

The author adopted a classic version of CNN filter strategy [10].

- The first convolutional layer has 10 filters of size 5x5 with a stride of 1 and a zero-padding of 'same'. The second convolutional layer has 40 filters of size 5x5 with a stride of 1 and a zero-padding of 'same'.
- After each convolutional layer, a ReLU activation function is applied to introduce non-linearity into the model. Then, a max pooling layer of size 2x2 with a stride of 2 is applied to reduce the spatial size of the feature maps.
- The output of the second max pooling layer is then flattened and fed into a fully connected layer with 500 neurons, which is again followed by a ReLU activation function.
- Finally, the output of the fully connected layer is passed through a softmax activation layer with the number of neurons equal to the number of classes in the problem.

2.2. Homography Transform

Homography transformations, also known as planar perspective transformations, is a fundamental concept in computer vision and image processing (Figure 3). This method serves the purpose of modeling the geometric relationships between two images of the same scene, taken from different viewpoints or with different camera configurations. Hartley and Zisserman, in their book Multiview Geometry in Computer Vision (2011), provide a definition of homographic transformations:

A projective transformation that maps corresponding points from one image to another can be described as a 3 by 3 matrix H [3].

$$[x', y' \; 1] = \mathrm{H}[x, y \; 1] \qquad (1)$$

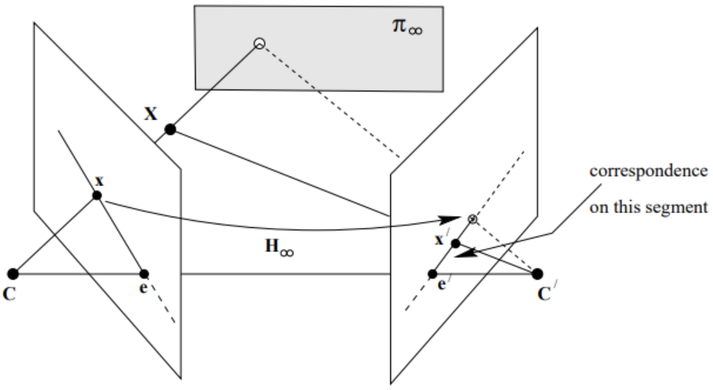


Figure 3. Scene planes and homographies [3].

2.2.1. Match points between image

SIFT: Scale Invariant Feature Transform, which is a feature point detection and matching method with good stability and invariance.

Gaussian kernel function: extract key points.

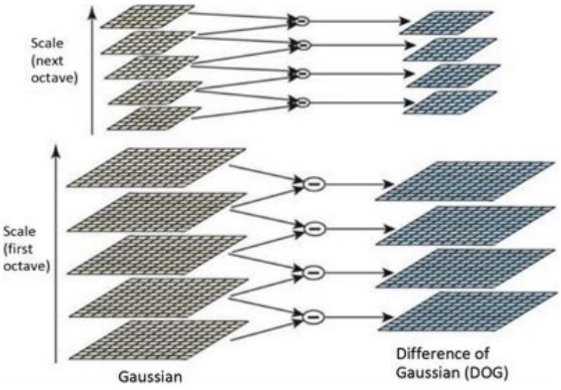


Figure 4. Gaussian pyramid and the dog pyramid.

This paper uses SIFT algorithm to extract feature points from the images, while the Gaussian kernel function is used to calculate the similarity between these points (Figure 4). It is an efficient approach to calculate the homography matrix between two images according to Wang, S who suggests that the SIFT algorithm works well for image matching and recognition, particularly when combined with other techniques such as Fast Library for Approximate Nearest Neighbors (FLANN) and Random sample consensus (RANSAC). The experimental data and analysis presented in the article demonstrate the effectiveness and accuracy of the SIFT algorithm in image points matching [7].

2.2.2. Calculate homography matrix

To find the homography matrix between two target images, just simply apply the Direct Linear Transformation where x and x’ are the homogeneous coordinates of the points in the first and second images, respectively.

$$\boldsymbol{x'} = \boldsymbol{H} * \boldsymbol{x} \tag{2}$$

Solve for the homography matrix H by minimizing the reprojection error:

$$min||A * h - B||^2 \tag{3}$$

Normally it is suggested to use Singular Value Decomposition (SVD) to solve this equation.

3. Result

This section first presents the data set and relevant details, and then gives the analysis of the experimental results.

Figure 5. Traffic sign dataset.

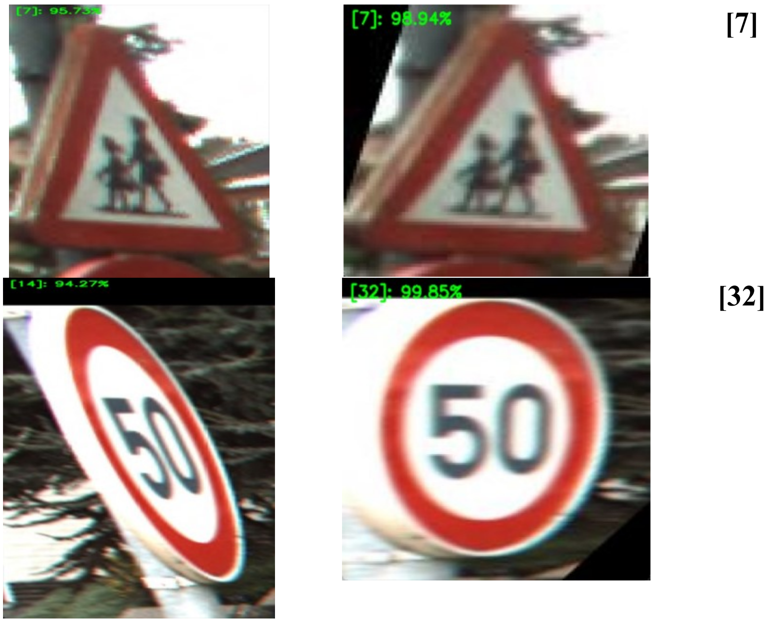
The author adopts a pre-categorized traffic sign dataset having around 1.5k images, the data source is described above where each folder with a unique number contains a certain category of traffic sign (Figure 5). For this experiment, the author inclines to sample images that don't have the front view of the sign which is not relevant with the paper argument and avoid inflation on accuracy. In general conditions, statistical random sampling will take 10 – 20 percent of the population.

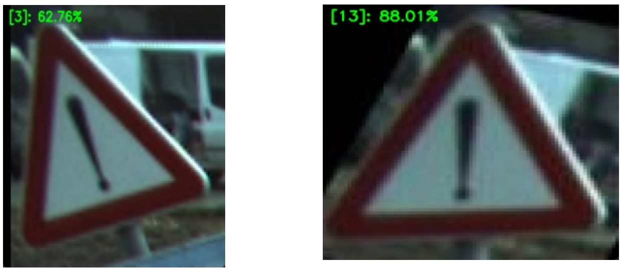
In addition, the author only considers traffic signs that has triangle or circle shape to control unintended interference factors when applying homography transform. After random sampling from each category, the test sample size in this paper is 60.

3.1. Analysis on the result

Images on figure 6 (Left) stand for the control group, and images on 6 (Right) stand for the homography transformed group.

Correct Answer





[13]

Figure 6. Control and homography transformed.

Author finds that the following situation will make the homography transform significantly helpful for classification accuracy:

Input image has an extreme camera angle toward the target, the second image on the left. Since this kind of image distributes useful information in areas with very few areas, it is more likely to mislead the CNN image classification in result of a wrong category. On the other hand, it is obvious that the homography transformed image (2th image) on the right has substantially flattened useful information in the middle, resulting in a correct classification. Input image has small size (less pixel) 3th image also tend to generate wrong classification by the algorithm since less base of pixels will cause low fault tolerance after applying multiple filter layers. Therefore, a small camera angle modification by holography transforms like the third image could correct a wrong classification.

The CNN classification result will generate two values:

Value in the represents the category predicted by the CNN model and the percentage value on the right is the confidence of that predicted by the model. Thus, the following passages will consider the probability of correctness. Observe that sample results by only applying the homography transformation for the target image could increase the certainty for classification when the original prediction is already high and correct the classification when the original prediction is uncertain and wrong.

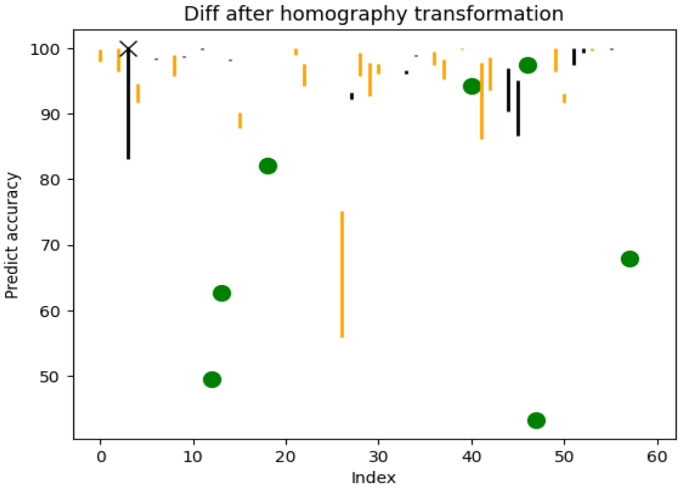


Figure 7. Diff after hormography transformation.

The y-axis is the probability that the CNN model predicts for the given traffic sign image belonging to a type of traffic sign. The x-axis is the index of each sample (Figure 7). For each index, the author draws a vertical line connect the two values for the same index represents the different of prediction accuracy before and after the homography warp of the given image. If the line is colored orange, it represents an increase in accuracy and black means a decrease.

There are two more situations demonstrated in this graph:

① The original prediction on classification is correct and the homography treated one is incorrect

② The original prediction on classification is incorrect and the homography treated one is correct

For context 1, the author marks a black cross sign on the value. For context 2, the author marks a green point.

In this graph, it is visible that majority of the homography transformed images gain reinforcement either on accuracy or even correct the original false prediction on category.

3.2. Statistical examine the result

In order to scientifically determine the effect of this strategy, the author in the following will use statical tests to examine. Since the sample size is > 50, the author will first assume the distribution for two outcomes are normal.

The null hypothesis (H0) is that there is no difference in probability (beta0 = beta1) between the two results, while the alternative hypothesis (H1) is that there is an improvement.

Consider there is prediction error in both groups, the author decides to assign zero accuracy value for incorrect prediction cases.

- Define the test statistic:

Author will use t-statistic:

$$t = \frac{mean_1 - mean_2}{\left(\frac{s}{n^{\frac{1}{2}}}\right)} \tag{4}$$

s is the pooled standard deviation of the two samples, calculated as:

$$s = \sqrt{\left\{\frac{\left((n_1 - 1) * s_1^2 + (n_2 - 1) * s_2^2\right)}{n_1 + n_2 - 2}\right\}} \tag{5}$$

- Set the significance level:

This paper will apply the most widely used 0.05 level of significance for all the statistical test in this paper.

- Compute the p-value:

Result of applying Welch Two Sample t-test is at follow (Table 1).

Table 1. Result of applying welch two sample t-test.

t	-2.0777
mean of control = 85.63583	mean of treated = 94.76383
p-value	0.0206
95 percent confidence interval	-Inf -1.810377

Since the p-value is less than 0.05 level of significance, reject the null hypothesis and adopts the alternative that the homography transform does have an improvement on the prediction accuracy. P-value less than half of the significance level and the 95 percent confidence interval has an upper bound -1.81 which is highly distanced from zero indicate a strong support for the conclusion. The only concern is that the lower bound for 95 percent confidence interval is negative infinite which is a sign of unstable for changes in data. Furthermore, if examine the normality on the sample (Figure 8).

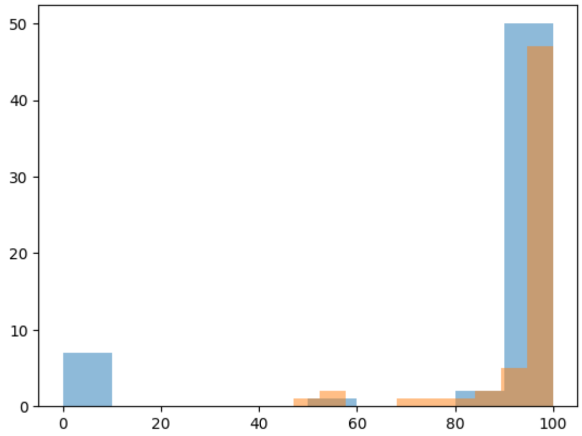


Figure 8. Frequency distribution.

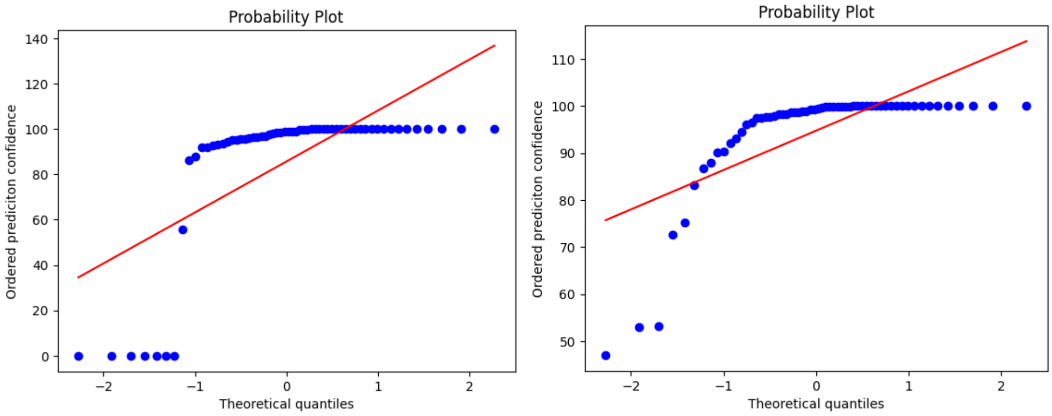


Figure 9. Q-Q plot for control and Q-Q Plot for treated.

Looking at the disparity of the sample, both box plot and Q-Q plot incline to imply that the sample is not likely to be normal distributed, though the homogrpahy prediction has some smooth effect at the lower tail (Figure 9).

Nevertheless, it does not necessarily mean that the sample is less trustworthy. CNN prediction commonly adopts a threshold strategy which makes a decision only when the confidence level passes a certain level. Therefore, it is by mechanism that small prediction confidence at the lower tail will more likely results in wrong prediction which counts to a zero.

This is not to say that the t-test is unreliable, but the author will apply further use additional statistical test to give multiple validation.

3.3. Contingency table test on correction

Since the data is in repeated measure and only two measurement points (raw prediction, homography prediciotn), McNemar's test is applicable which is a classic statistic approach to evaluate machine learning result (Table 2).

Table 2. McNemar's test.

Raw prediction /Homography prediction	Correct	Incorrect
Correct	51	7
Incorrect	1	1

Consider the sample size is not very large and the value in two position is 1, the author decides to apply the adjusted version (constant term 0.5) of McNemar's test. The null hypothesis H0 is that the homography transform is ineffective.

$$X^2 = \frac{(|B-C|-0.5)^2}{B+C} = \frac{(7-1)^2}{8} = 4.5 \tag{6}$$

P-value resulted from this statistic is 0.03389485 which is less than the 0.05 level of significance, therefore reject the null hypothesis and concludes that homography transform does have improvements on the prediction.

Table 3. Fisher's exact test.

Method	Correct	Incorrect
Raw prediction	52	8
Homography prediction	58	2

Author additionally uses Fisher's exact test, Dual Verification, to test proportions for correction are different among raw prediction and Homography prediction (Table 3). Fisher's exact test is favourable in this case because author provides a contingency table and the sample size is not large (Fisher's exact test is more accurate than the chi-square test or G–test of independence when the expected numbers are small.).

The null hypothesis H_0: θ = 1 versus HA: θ < 1, where θ is the odds ratio for raw prediction versus Homography prediction. H0 stands for there is no difference between the two methods and HA stands for Homography prediction is better than raw prediction.

Table 4. P-value test.

p-value	0.04731
95 percent confidence interval	(0.0000000 0.9827049)
sample estimates: odds ratio	0.2266694

According to the test result (Table 4), since p-value is less than the common 0.05 significance level, reject the null hypothesis and favor the alternative hypothesis that Homography prediction is better than raw prediction. It is 95% confident that the true odds ratio is between 0 and 0. 9827049 is less than 1. The sample estimates odds ratio indicates that the odds of being correct for Raw prediction is 0. 2266694 times that for homography prediction which is a significance enhancement for the test case.

Therefore, Fisher's exact test supports that Homography prediction is statistically significantly better than raw prediction on correction. Although the p-value for this test approximates the threshold of 0.05, it is reasonable to keep observe this effect since the more refined and complex the CNN algorithm is, the less significant difference will appear in this test for having small prediction error base.

4. Conclusion

This paper concludes that applying homography transform could have accuracy improvements on CNN image classification. For the sample introduced in the paper, applying homogrpahy transform definitely improves the accuracy of prediction by passing Two Sample t-test, McNemar's test, and Fisher's exact test. For the context of traffic sign classification, further validation could be done by increasing the sample size, testing CNN algorithms that have different layer strategy, or source the sample image from real auto-driving cars' camera. Although the sample size in this paper is not generally large to give a robust conclusion for large scale, it is still reasonable to consider that this method could be plausible under other contexts and worthwhile for further practices.

References

[1] Tanzeel U. Rehmana, Md. Sultan Mahmudb, Young K. Changb, Jian Jina, Jaemyung Shinb. Current and future applications of statistical machine learning algorithms for agricultural machine vision systems, 2018, Computers and Electronics in Agriculture, 156:585-605.

[2] Wang, Y., Yu, M., Jiang, G., Pan, Z., & Lin, J. Image Registration Algorithm Based on Convolutional Neural Network and Local Homography Transformation. 2020 Applied Sciences, 10(3):732.

[3] Hartley, R., & Zisserman, A. Multiple View Geometry in Computer Vision. 2011 Cambridge Core. https://doi.org/10.1017/CBO9780511811685

[4] Liu, Y. Analysis of Key Technical Problems in Internet of Vehicles and Autopilot. 2020, Advances in Intelligent Systems and Computing, 1-10.

[5] Dingle Robertson, L., King, D.J. Comparison of pixel-and object-based classification in land cover change mapping.2011 Int. J. Remote Sens. 32 (6), 1505–1529.

[6] Zhang, S., Wu, X., You, Z., Zhang, L., Leaf image-based cucumber disease recognition using sparse representation classification. 2017, Comput. Electron. Agric. 134, 135–141.

[7] Wang, S., Guo, Z., & Liu, Y. An Image Matching Method Based on SIFT Feature Extraction and FLANN Search Algorithm Improvement. 2021 Journal of Physics: Conference Series. 201-213.

[8] Lin, C. C., & Wang, M. S. A Vision Based Top-View Transformation Model for a Vehicle Parking Assistant. 2012, Sensors, 12(4):4431-4446.

[9] The Car and the Camera. (n.d.). Google Books. https://books.google.com/books/about/The_Car_and_the_Camera.html?hl

[10] Jaemyung Shinb. Spatio-Temporal Anomaly Detection for Industrial Robots through Prediction in Unsupervised Feature Space. (n.d.) 2021 Advances in Intelligent Systems and Computing, 122-134.

Fruit 360 classification based on the convolutional neural network

Dehui Zhang

The school of engineering, Rutgers university, New Brunswick, 08901, United states

dz273@rutgers.edu

Abstract. This research paper focuses on the Fruit360 Classification challenge, a task aimed at developing a fruit classification model capable of accurately identifying various fruits and distinguishing them from each other. In this study, the Fruit360 dataset is used, consisting of 90380 images of 131 fruits and vegetable classes. Prior to training the CNN model, the images are preprocessed by resizing, normalizing, and augmenting them. The authors employ a pre-trained CNN model called ResNet-50 using the PyTorch deep learning framework and add a custom fully connected layer on top to adapt the model to the specific classification task. The authors conclude that the proposed model achieved excellent performance on the Fruit360 dataset. The study highlights the importance of the Fruit360 Classification challenge in advancing the field of computer vision, specifically in the development of deep learning algorithms for image classification tasks. The proposed model has the potential to improve the efficiency and accuracy of fruit classification, which can benefit the fruit industry in terms of enhanced productivity and cost-effectiveness.

Keywords: fruit classification, convolutional neural network, deep learning.

1. Introduction

Fruit360 is a well-known dataset used for fruit classification. It has been widely used by researchers to develop computer vision algorithms for identifying fruits. The dataset contains 90380 images of 131 different fruits. The Fruit360 Classification challenge is an ongoing research topic that involves the classification of different fruits using machine learning algorithms. The primary objective of this research is to develop an efficient classification model that can accurately identify different fruits and distinguish them from one another. The importance of this research lies in its potential to enhance the productivity and efficiency of the fruit industry, which heavily relies on accurate and fast classification of fruits for sorting, packaging, and distribution.

Previous research on image classification has made significant contributions to the development of deep learning algorithms, which have been applied to a wide range of applications, including object detection, speech recognition, and natural language processing. In recent years, the development of deep Convolutional Neural Networks (CNNs) has revolutionized the field of computer vision, leading to significant improvements in image classification accuracy. For example, Z. Zhang et al. used a deep neural network to classify the Fruit360 dataset, achieving an accuracy of 98.56% [1]. Similarly, Y. Chen et al. used a transfer learning technique to classify the Fruit360 dataset, achieving an accuracy of 99.2% [2]. Although these studies have achieved high accuracy in fruit classification, there is still a gap in the current research. The current research gap lies in the performance of fruit classification algorithms on new and unseen fruit categories. The Fruit360 dataset contains only 131 different fruit categories, and it is essential to develop a classification algorithm that can generalize to new and unseen fruit categories, which deserves more attention.

The motivation for this study is to fill the research gap by providing a comprehensive analysis of the Fruit360 Classification challenge and its significance in the field of computer vision. Specifically, this study aims to reflect on the core differences between previous studies and the Fruit360 Classification challenge, and highlight the contribution of this research to the field of computer vision. The Fruit360 Classification challenge presents a unique opportunity to develop and evaluate deep learning models for fruit and vegetable classification, and to identify new approaches to address the complexity and variability of fruit and vegetable images. This paper will provide an overview of the Fruit360 Classification challenge, including its dataset, evaluation metrics, and previous winning solutions. Furthermore, this study will present a detailed analysis of the challenges and opportunities presented by the Fruit360 Classification challenge, and propose new approaches to address the limitations of existing models.

2. Methods

In this section, the materials used in this study will be described, consisting of the procedure of preparing materials, the statistical tests used to analyze the data, and the measurements made during the study.

2.1. Dataset description and preprocessing

The dataset used in this study is the Fruit360 dataset, which consists of 90483 images of 120 fruits and vegetables classes [3]. Some sample images can be found in Figure 1. All images are based on the RGB format and are divided into training and testing sets, with a ratio of 80:20. The classes and corresponding label names can be found in the dataset documentation.

Figure 1. The sample images on the Fruit360 dataset.

Prior to training the CNN model, some preprocessing steps were carried out on the images. First, this study resized all images to 224x224 pixels using the Python imaging library (PIL) to make the computational memory controllable [4]. In addition, the pixel values were also normalized to be between 0 and 1 by dividing each pixel value by 255.

2.2. Employed CNN model

CNNs are a type of deep neural network commonly used for image classification tasks. The main modules of CNN include convolutional layers, pooling layers, and fully connected layers [5].

In this study, a pre-trained CNN model called ResNet-50 using the PyTorch deep learning framework was built [6]. ResNet-50 is a deep CNN architecture that has shown impressive results in various image classification tasks [7, 8]. This pre-trained model as a feature extractor was utilized and this study then added a custom fully connected layer on top to adapt the model to the specific classification task.

2.3. Implementation details

During the training procedure, this study set the learning rate to 0.001 and used the Adam optimizer with default parameters. The cross-entropy loss function was employed to measure the difference between the predicted and ground truth labels [9, 10]. The model was trained for 50 epochs with a batch size of 32. The strategy called early stopping was also considered for preventing overfitting of the model to the training set. In addition, the performance of the model is evaluated based on the accuracy, precision, recall, and F1-score as evaluation metrics. This study also used the confusion matrix to analyze the performance of the model on each class.

In summary, this study used the Fruit360 dataset and preprocessed the images by resizing, normalizing, and augmenting them. The ResNet-50 CNN model using PyTorch was employed and the learning rate, optimizer, loss function, epochs, and batch size for training was also determined. The performance of the model using various evaluation metrics and the confusion matrix is evaluated.

3. Results and discussion

This investigation employed the Fruit 360 dataset to evaluate the effectiveness of a classification model. Multiple parameters, including epoch and batch size, were examined, and the VGG-16 model was incorporated into the trials. The findings demonstrated that the initial accuracy of the model was moderate, but with the progression of each epoch, the accuracy substantially improved, peaking at 90.27% during epoch 30. A confusion matrix illustrating the model's predictions was generated and presented in Figure 2.

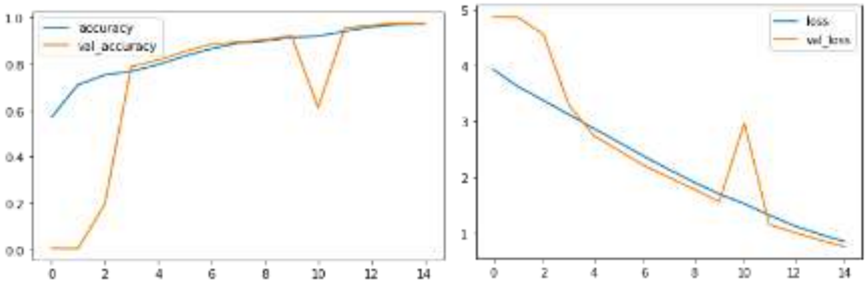


Figure 2. The performance of the proposed model.

The results of this investigation indicate that the accuracy of the classification model applied to the Fruit 360 dataset can be enhanced by adjusting the parameters and incorporating the VGG-16 model. A comparison of the outcomes of this study with previous research reveals that the model performed better than some of the existing models utilized with the same dataset. Additionally, the results indicate a substantial improvement in accuracy following the initial epochs, possibly due to the model's aptitude in recognizing and learning patterns within the dataset. Moreover, the findings suggest that a batch size of 32 was optimal for the model. These adjustments to the parameters and inclusion of the VGG-16 model can achieve a high level of precision in recognizing various types of fruits. Nevertheless, further improvement is still possible, and future investigations could explore more sophisticated models or optimization techniques to enhance the performance of the classification model.

4. Conclusion

The purpose of this investigation was to construct an efficient classification model for the Fruit 360 dataset utilizing machine learning techniques. A convolutional neural network model was developed, which exhibited a remarkable degree of accuracy in identifying distinct fruit types present in the dataset. This research demonstrates the practical applications of machine learning models in fruit classification, particularly in the agriculture industry. Additionally, the classification model's performance surpassed or was comparable to other studies that utilized analogous techniques on the same dataset. Nevertheless, this study has specific limitations, such as the restricted number of fruit types in the dataset. In the future, the research will expand to include a broader range of fruit types and explore the potential of transfer learning to enhance classification performance. Additionally, further study intends to apply the methodology to other analogous datasets to assess its generalizability. In summary, this study highlights the potential of machine learning techniques in fruit classification, providing a useful tool for fruit growers and researchers.

References

[1] Zhang Z Zhang Y and Li J 2020 A novel deep learning framework for fruit recognition based on improved convolutional neural networks IEEE Access vol 8 pp 2308-2321

[2] Chen Y Chen H and Wu Y 2021 Fruit recognition using deep transfer learning in Proc 2021 4th Int Conf on Intelligent and Innovative Computing Applications (ICIIA) p 86-89

[3] Mureșan H and Oltean M 2018 Fruit recognition from images using deep learning Acta Universitatis Sapientiae Informatica 10(1) 26-42

[4] Pillow S 2020 Python Imaging Library Handbook Retrieved from https://pillow.readthedocs.io/en/stable/handbook/index.html

[5] LeCun Y Bengio Y and Hinton G 2015 Deep learning Nature 521(7553) 436-444

[6] He K Zhang X Ren S & Sun J 2016 Deep residual learning for image recognition In Proceedings of the IEEE conference on computer vision and pattern recognition (p 770-778)

[7] Yu Q Chang C S Yan J L et al. 2019 Semantic segmentation of intracranial hemorrhages in head CT scans 2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS) IEEE 112-115

[8] Li B, Lima D 2021 Facial expression recognition via ResNet-50 International Journal of Cognitive Computing in Engineering 2: 57-64

[9] Martinez M Stiefelhagen R 2019 Taming the cross entropy loss Pattern Recognition: 40th German Conference GCPR 2018 Stuttgart, Germany October 9-12 2018 Proceedings 40 Springer International Publishing 628-637

[10] Pang T Xu K Dong Y et al. 2019 Rethinking softmax cross-entropy loss for adversarial robustness arXiv preprint arXiv:1905.10626

Brain tumour MRI detection and classification based on the convolutional neural network

Yihan Xie

The Department of Computer Science, University of Toronto, Toronto, M5S 2E4, Canada

yihan.xie@mail.utoronto.ca

Abstract. Magnetic Resonance Imaging (MRI) has emerged as a widely used diagnostic technique for brain tumour detection. However, the diagnosis of brain tumours poses significant challenges due to their occurrence in diverse locations and various types. Furthermore, MRI generates images that require manual analysis by physicians, which can be laborious and prone to errors. To enhance the efficacy and accuracy of brain tumour detection, recent advances in artificial intelligence have led to the development of machine learning algorithms. In this study, a convolutional neural network (CNN) based method was proposed for brain tumour detection and classification through the preprocessing of raw MRI images. The customized CNN model achieves an accuracy of 98% on a dataset consisting of four types of MRI images, including three types of brain tumours and healthy brain images, with preprocessing applied to all images. The CNN model demonstrates an accuracy of 95% in classifying raw MRI images from the dataset. The CNN model's performance is further improved by training the model with preprocessed images that have been transformed into the same colour space and object area zoomed in. These findings provide a promising avenue for the development of automated and efficient brain tumour detection systems using CNN and MRI.

Keywords: brain tumour, MRI, CNN, machine learning, deep learning.

1. Introduction

Brain Tumor, also known as intracranial tumour, is an abnormal and uncontrollable growth of cells in the brain or near the brain tissue. As the most complex organ of the human body, the brain consists of different units: the forebrain including the cerebrum and tissue below it, the midbrain, and the hindbrain including the brain stem, upper spinal cord, and cerebellum. Brain tumours are not restricted to a fixed location, they can be found in any part of the brain and the skull. They may be classified as primary, originating from brain tissue or nearby, or metastatic, originating from other regions of the body. Brain tumours vary in many types and will cause pain and problem in people’s life. People with Brain tumours are like to experience symptoms such as dizziness, headache, thought disorder, loss of hearing, vision changes, and memory loss. Nervous system cancer including brain tumours is the 10th leading cause of death for humans [1]. Worldwide, approximately 251, 329 deaths are caused by primary malignant nervous system tumours. Estimatively 24810 adults are diagnosed with primary malignant brain tumours in the U.S. in 2023 [2]. Brain tumours can exert pressure on and damage other parts of the brain tissue, leading to brain dysfunction and can also spread to other parts of the body. Magnetic Resonance Imaging

(MRI) is the most popular technology for diagnosing. Determining the specific type of tumour usually requires surgery to test the brain tissue sample taken out from the body.

However, the process of diagnosis and classification of brain tumours is complex, and invasive which is associated with significant labour costs. In addition, MRI and Computerized Tomography (CT) scans do not always provide sufficient resolution to fully characterize the properties of the tumours, which will possibly cause misdiagnosis of the tumour. Delayed diagnosis, invasion, misdiagnosis, and high labour costs are all factors that reduce the recovery rate of brain tumours. Fortunately, recent advances in Artificial Intelligence (AI) have led to the development of novel techniques that hold promise for improving the diagnosis of brain tumours.

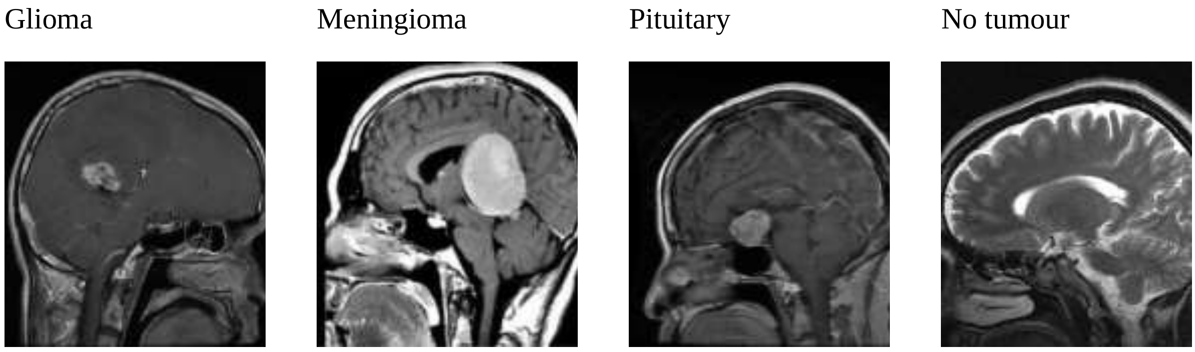
AI is a technique in that machines can work dynamically as if they had human intelligence relying on the AI algorithm. AI techniques include machine learning, deep learning, natural language processing, computer vision, and robotics. When working with a huge amount of data, well-developed AI algorithms can improve accuracy and precision, increase efficiency and productivity and reduce labour costs, freeing up human labour for more complicated work with low substitutability. Now, AI has been applied in various areas including the medical field. AI has shown its ability to improve the accuracy of the diagnosis of brain tumours. For instance, Alrashedy et al. employed the BrainGAN framework to generate and classify brain tumour images and reached an accuracy of 99.09% with ResNet152V2 [1].

In this paper, the convolutional neural network (CNN) will be used in MRI brain tumour classification after applying a preprocessing algorithm to raw data. Data is obtained from consisting of 7, 023 pictures divided into four categories: glioma, meningioma, pituitary and no tumour. For each class, the training and testing data can be accessed [2]. The study trains the models using training raw data and preprocessing data, evaluates the accuracy as well as the loss of the model and compares the performance of the models on the dataset in the two experiments. The result shows that the customized CNN model proposed in this study works well on the dataset without any data preprocessing before the training and reached an accuracy of 95%. With preprocessing algorithm applied to the data before training, the model performs even better and achieves an accuracy of 98%.

2. Method

2.1. Dataset

The dataset used in this study was sourced from [3]. It contains 7, 023 MRI images of human brains in total, categorized into four classes meningioma, glioma, pituitary, and no tumour. The dataset consists of a training set and a testing set for the purpose of training and evaluating the model, respectively. Sample images of these four classes are shown in Figure 1.



Glioma

Meningioma

Pituitary

No tumour

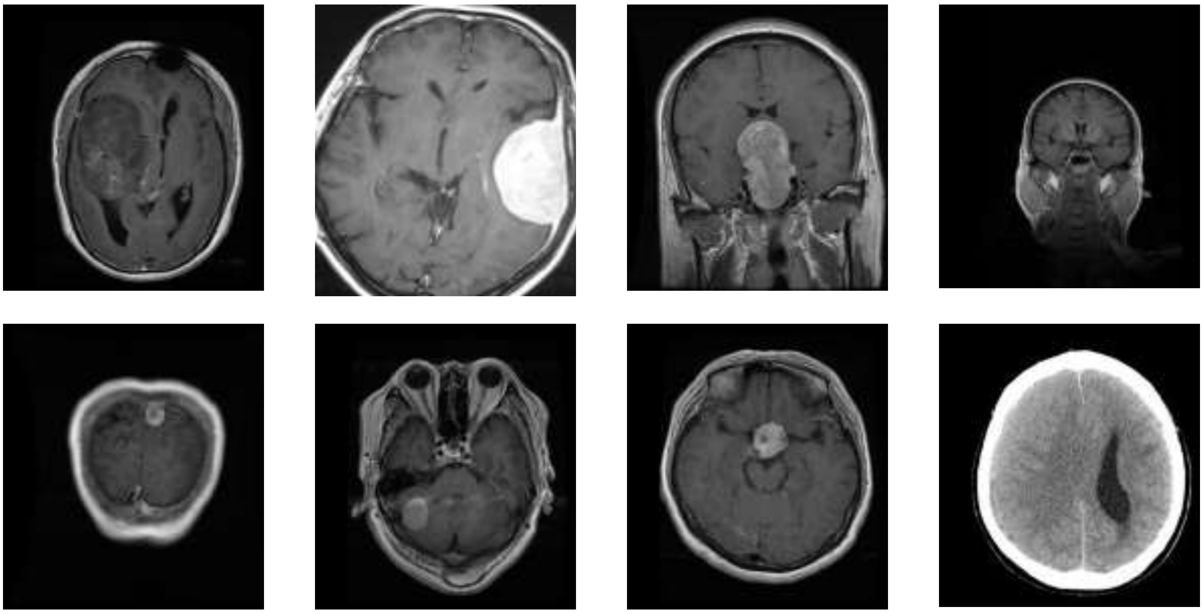


Figure 1. The sample images of the collected dataset.

To ensure the accuracy of the classification related to the grey-scaled MRI images in the dataset, the resizing operation was carried out first. The other preprocessing operation used Python packages called numpy, tqdm, cv2, os and imutils. For each image in the training set and testing set, the extreme points were found and then the rectangle out of them will be cropped. To accomplish this, cv2.cvtColor() method is applied first to convert all the images to cv2.COLOR_BGR2HSV colour space. Then, cv2 GaussianBlur() applies Gaussian smoothing to the converted images with kernel size [3 × 3]. When thresholding the images, cv2.threshold() is applied to the blurred images so that all pixel values above 45 are set to 255. After that comes morphological operations erosion and dilation. Erosion is performed twice on each image by cv2.erode() and dilation is performed twice by applying cv2.dilate() with the number of iterations set to 2 for both. Functions cv2.findContours() and imutils.grab_contours are used to find contours in the thresholded images and then grab the largest. Then the extreme points are found, by which images are cropped. Then, all the images are resized to size (100 × 100). So far, in each image of the dataset, the object area has been increased and the features of images are accentuated.

2.2. CNN model

Convolutional Neural Network (CNN) is a deep learning algorithm which is commonly used for image processing and analysis in different tasks [4-8]. A CNN model can be specified as a sequential model consisting of a series of neural layers. On each layer, computations are performed on the input data of this layer and the output of the layer is provided to the next layer as input [9]. In this way, the input data goes through the layers one after one, sequentially. The CNN model constructed in this study comprises 12 layers of three types, which are the convolutional layer, max pooling layer, and fully-connect layer [10].

The convolutional layer with a kernel of a specific size is the layer where the convolution is performed. The kernel is shifted across the input image. Each time, the kernel is applied to an area of the input, and the dot product between the covered pixels and the kernel is calculated and recorded. In this project, each convolutional layer has a 3 × 3 kernel. Rectified Linear Unit (ReLU) is used as the activation function for each convolutional layer. The max pooling layer is often used after a convolutional layer and is the place where a filter is shifted across the image and keeps only the pixel with the highest value each time. The fully connected layer) learns a set of weights and biases during training and updates these weights and biases. Every neuron in the dense layer is connected to each neuron in the previous layer. The CNN model of this project has the structure shown in Table 1.

Table 1. The strcuture of the proposed CNN.

Layer-1	Convolutional layer of 64 3 × 3 filters
Layer-2	Max pooling layer
Layer-3	Convolutional layer of 64 3 × 3 filters
Layer-4	Max pooling layer
Layer-5	Convolutional layer of 128 3 × 3 filters
Layer-6	Max pooling layer
Layer-7	Convolutional layer of 256 3 × 3 filters
Layer-8	Max pooling layer
Layer-9	Flatten layer
Layer-10	Dense layer (ReLU)
Layer-11	Dropout (0.5)
Layer-12	Dense layer (softmax)

2.3. Implementation details

The CNN model is constructed by using Tensorflow. The model is instantiated by creating an object of TensorFlow.keras.Sequential class. Layers are created by tensorflow.keras.layers and added to the model sequentially from the first to the last. The CNN model is compiled using Adam as the optimizer and sparse_categorical_crossentropy as the loss function. The metric to be evaluated by the model during training and testing is accuracy. In this study, the model is trained by original data that is not preprocessed in 200 epochs and by preprocessed data in 200 epochs.

Data generators for training and testing sets are created using ImageDataGenerator() from TensorFlow.keras, and then the flow_from_dirctory method is used to take the path to a directory and generates batches of augmented data. With the raw data and preprocessed data, the data generators will pass the original data and preprocessed data to train the model respectively. The four classes in the training and testing set are labelled as pituitary -> 4, no tumour -> 3, glioma -> 1, and meningioma -> 2.

3. Result and discussion

The loss to the index of epochs and the accuracy to the index of epochs of the model trained by non-preprocessed data and preprocessed data are plotted in Figure 2.

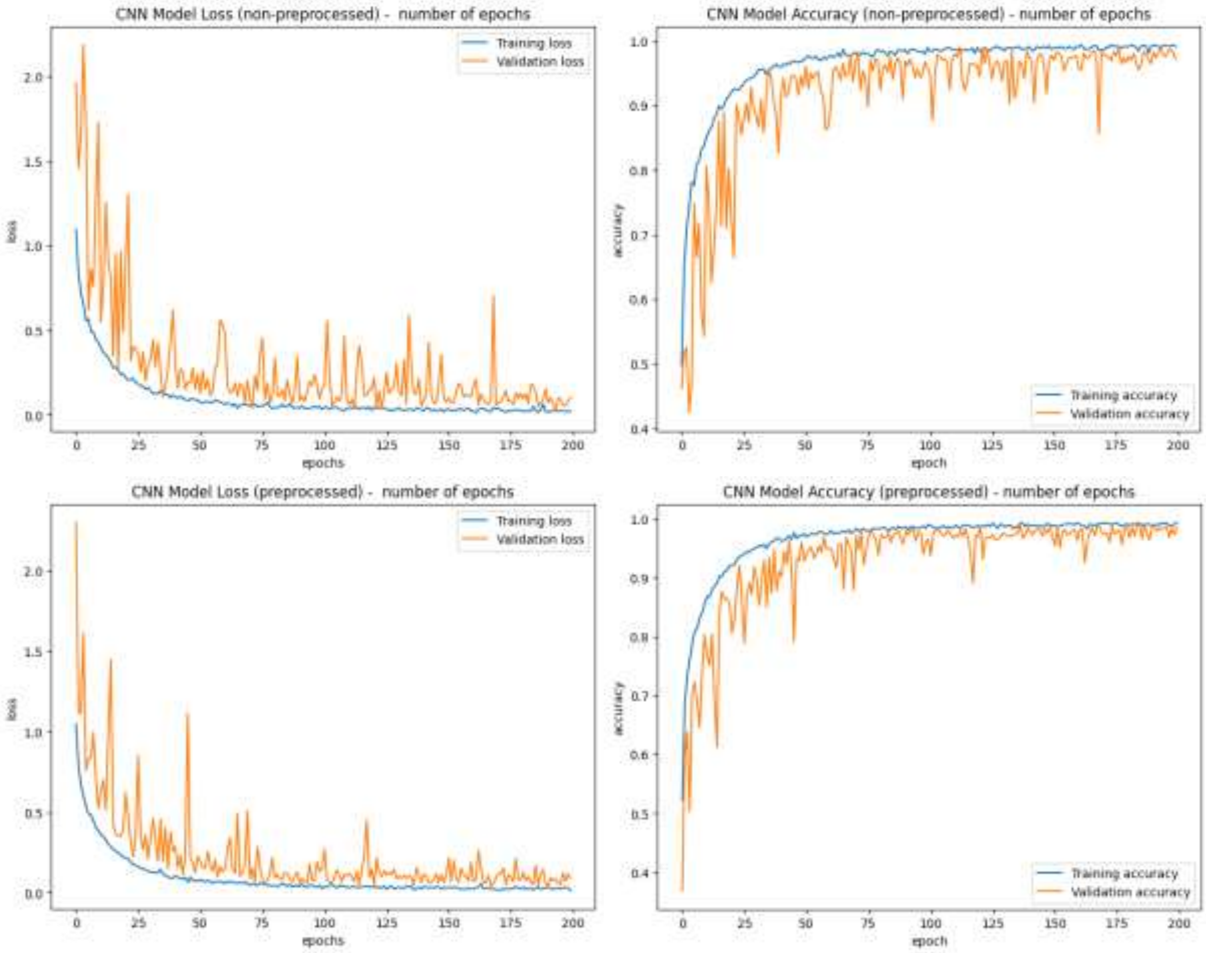


Figure 2. The performance of the model during the training process.

From the graphs, in the first 75 epochs, the training loss and validation loss of the model trained by original data and the model trained by preprocessed data both significantly decreased, the training accuracy and validation accuracy of the model trained by original data and the model trained by preprocessed data both significantly increased. Comparing the two pairs of graphs, it is clear that the accuracy of the model trained by data without preprocessing has a larger fluctuation when growing with a slower rate of growth than the model trained by the preprocessed data. The model trained by original data reached an accuracy of 98% and the accuracy stayed around 94%. However, the loss and accuracy of the model trained by preprocessed data have a smaller fluctuation during the decrease and growth. In addition, the curve of validation loss/accuracy and the curve of training loss/accuracy are more matched by using preprocessed data to train the CNN model. With preprocessed data, the accuracy of the model reached 98% earlier and stayed close to 98%.

The results suggest that preprocessing the data enables the model to attain similar levels of accuracy in fewer epochs and achieve higher accuracy overall. This is possible because the preprocessing operation converts the images into the same colour space, detect the object area and zoom in, and crop the images by extreme points. However, the non-preprocessed images vary in colour space, size, margin, and area of the objects. Thus, the preprocessed data has higher consistency with larger, more centred object areas, and less interference than the original data.

4. Conclusion

In this work, a customized CNN model is proposed to help diagnose and classify brain tumours using brain MRI images from. This study constructed a CNN model, preprocessed data with a preprocessing algorithm, and trained the model with both raw and preprocessed data. Experiments were carried out to evaluate and compare the performance of this CNN model trained by raw and preprocessed data covering 4 types of brain tumour MRI images. Results showed that the customized model trained with the preprocessed data performs better with higher accuracy and lower loss than that trained with raw data. In the future, the further study is looking forward to developing and testing the proposed method on datasets of more types of brain MRI images and then adapting the method to assist doctors with brain tumour diagnosis.

References

[1] Alrashedy H H N Almansour A F Ibrahim D M et al. 2022 BrainGAN: Brain MRI Image Generation and Classification Framework Using GAN Architectures and CNN Models Sensors 22(11): 4297

[2] Cancer.Net 2023 Brain Tumor https://www.cancer.net/cancer-types/brain-tumor/statisticshttps%3A//www.cancer.net/cancer-types/brain-tumor/statistics

[3] Kaggle 2021 Brain tumor mri dataset https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset

[4] Luo X Hu M Song T et al. 2022 Semi-supervised medical image segmentation via cross teaching between cnn and transformer International Conference on Medical Imaging with Deep Learning PMLR 820-833

[5] Tiwari P Pant B Elarabawy M M et al. 2022 Cnn based multiclass brain tumor detection using medical imaging Computational Intelligence and Neuroscience 2022.

[6] Chen S Gamechi Z S Dubost F et al. 2022 An end-to-end approach to segmentation in medical images with CNN and posterior-CRF Medical Image Analysis 76: 102311

[7] Yu Q Wang J Jin Z et al. 2022 Pose-guided matching based on deep learning for assessing quality of action on rehabilitation training Biomedical Signal Processing and Control 72: 103323

[8] Heising L Angelopoulos S 2022 Operationalising fairness in medical AI adoption: detection of early Alzheimer’s disease with 2D CNN BMJ Health & Care Informatics 29(1)

[9] Géron A 2022 Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow O'Reilly Media Inc

[10] Wu J 2017 Introduction to convolutional neural networks National Key Lab for Novel Software Technology Nanjing University China 5(23): 495

Chinese stock price prediction under COVID-19 period based on linear Regression Model

Yijia Zhang[1, †], **Yiran Yang**[2, 3, †]

[1]The department of artificial intelligence, Beijing university of chemical technology, Beijing, 102202, China
[2]The department of computer science, Xiamen university of technology, Xiamen, 361000, China

[3]2010716438@s.xmut.edu.cn
[†]These authors contributed equally.

Abstract. To anticipate and assess the price trend of scientific and technical stocks in a time of high volatility following the breakdown of China's COVID-19 Epidemic Economic Policy at the end of 2022, we plan to utilize a training linear regression model in this work. assisting people and businesses in making better analyses and decisions during this time of high risk, thereby lowering investment risks. This paper uses the stock price data of Alibaba, Tencent, and Xiaomi in Quandl during the new crown epidemic to represent the general trend of Chinese technology stock prices. We preprocessed the data to retain features that better reflect the characteristics of the data and remove features that have less impact on the data analysis. This study chose a linear regression model to model various relationships through the classifier used by Scikit-Learn for regression and estimated the unknown parameters in the linear regression model from the data. The data from the training and test sets are used to train the linear regression model, and the results are visualized as graphs, which can more intuitively convey the overall trend and local volatility of stock prices. In this study, the accuracy of the model can reach more than 85%. The visualized figure shows that although stock values recovered quickly when the regulation was lifted, they are still lower than they were before the outbreak due to the impact of China's past epidemic policies on their long-term fall. The experimental findings demonstrate the great accuracy with which the advanced regression model in this work can forecast the price trend of technology stocks throughout the period of high volatility following the policy's unsealing and the extent to which it may represent price volatility.

Keywords: linear regression, COVID-19, stock prediction, machine learning.

1. Introduction

Stocks represent a unit of ownership in a company and can be acquired through the sale of shares in exchange for monetary compensation. Investors purchase stock in anticipation of a gain in price that will allow them to profit when they sell it in the future [1]. In light of COVID-19, China has implemented a series of restrictive policies which have had significant impacts on the country's trade activities. At the beginning of the COVID-19 epidemic, the growth rate of China's foreign trade dropped sharply over the same period, and both the supply side and the demand side of China's trade have been seriously impacted [2]. As a result, the prices of numerous stocks have been adversely affected.

In 2022, China’s COVID-19lift restrictions, and the prices of many stocks will fluctuate significantly. For example, removing restrictions on entry and exit and foreign trade. As such, it is imperative to accurately predict high-volatility stocks, which will aid individuals, companies, and governments in making informed investment decisions and mitigating risks. The development of the science and technology industry is changing with each passing day, and the competition is fierce. It is a high-growth industry, and its growth rate is far greater than that of general industries. Therefore, this paper selects technology stocks as the data source of the model to predict the price trend of technology stocks after the COVID-19 epidemic in China in 2022. Especially, Artificial Intelligence (AI) was chosen in this study due to its robustness and satisfactory performance [3].

Stocks are initially predicted and analyzed using economic models. Early in the 1960s, Sharpe et al. created the Capital Asset Pricing Model (CAPM) to predict the stock, which offered the first logical framework for connecting the needed return on investment to the investment's risk [4]. Chen et al. put forward a market model based on multiple factors in Economic Forces and the Stock Market, which forecasts the changes in stock prices by considering macroeconomic variables and the characteristics of stocks [5]. However, these economic-based models are not very accurate.

With the development of AI technology, AI has become the main technology for predicting stock price trends due to its high accuracy and robustness. Yoon et al. proposed a neural network-based method that uses historical stock prices, transaction volume, and other factors as input to build a stock price prediction model to predict the rise and fall of stock prices [6]. Lee proposed to use the mixed feature selection method to process the data of the stock market and input the selected feature into the Support Vector Machine (SVM) model as an input variable to predict the rising and falling trend of the stock [7]. Although these models work well, many of them focus on stable or low-volatility stock forecasts. For the prediction of Chinese stocks in 2022, the model that can better predict high-volatility stocks is more critical. Therefore, this paper intends to use the linear regression model in the field of machine learning to analyze and predict the price trend of stocks in the period of high volatility after the lifting of restrictions on China's COVID-19 epidemic policy in 2022.

The data used in this paper are from the data set in Quandl, this dataset collects information on the stocks of representative technology companies’ financial market in China's epidemic period from the end of 2019 to the beginning of 2023. In the selection of technology stocks, this paper selected the stock data of Alibaba, Tencent, and Xiaomi as the representative of technology stock data for analysis and prediction. These three companies’ global market value is at the top of the technology companies with high popularity and high technology level, so their stock data can better represent the overall trend of the China technology industry. We forecast the stock trend by building a linear regression model and training the stock data of three companies in 2022. The experimental findings demonstrate the great accuracy with which the advanced linear regression model in this study can forecast the price trend of technology stocks in a period of high volatility following the policy's unsealing and the extent to which it may represent price volatility.

2. Method

2.1. Dataset description and preprocessing

In this project, we selected the stocks of three companies for analysis, which are Tencent, Xiaomi, and Alibaba. The stock data for these companies was obtained from the data.nasdaq.com platform, which offers a wealth of financial and economic data, encompassing a range of asset classes such as stocks, commodities, derivatives, fixed income, and mutual funds. The platform provides access to real-time, historical, and end-of-day data, sourced from various providers such as Quandl, Sharada, and Brave new coin, among others. In addition, Python also provided the package of quandl to help the researchers get the dataset easily. We chose the period from the end of 2019 to the beginning of 2023, including the period of COVID-19 in China. The datasets for each company comprised 12 volumes and approximately 797 rows, with each row indicating the date within the study period, and each volume representing a specific feature of the company's stock.

For a stock, the main features include high price, low price, close price, and turnover. And combined with the dataset obtained, we selected 'Nominal Price', 'High', 'Low', 'Previous Close', and 'Share Volume (000)' as the features that we used. This study's main purpose is not to study the complex relationship between the various eigenvalues, but only to predict a certain eigenvalue, so we chose the 'Nominal Price' as the label we want to forecast. To improve the utilization of data, we assign an outlier to the missing value, therefore, when a machine learning classifier processes the data, this will just be recognized and treated as an outlier feature. The data is nearly 800 days, and we want to predict the stock price for the next half month, which means the goal is to forecast out 2% of the entire length of the dataset. Based on this we defined the forecasted days variable' forecast_out' and then added a new column 'label' in the dataset to record the predicted value. The newly added column is expressed by moving the data in the 'close' column forward by 2% rows. Next, we defined X (all columns except label) as the feature and y (just 'label') as the label to predict. To speed up processing and improve accuracy, we used preprocessing module in sklearn to preprocess the X. 2% of the data was left when the label column was generated above. These rows do not have label data, so we can use them as input data for prediction. Then we created a new variable 'X_lately' to keep these data.

2.2. *Proposed approach - Linear regression*

In the context of regression problems in stock forecasting, Python provided an effective tool called Scikit-Learn. Scikit-Learn offers several classifiers for regression analysis, and after evaluating the accuracy of each model, the linear regression model was selected for this study. The linear regression algorithm is a frequently used method in solving estimation problems [8]. In linear regression, the relationships are modeled using linear predictor functions whose unknown model parameters are estimated from the data. Usually, we have a dataset D, x as a feature vector of dimension d, t as the target value (sometimes denoted by y), and the objective is to get a good mapping y through the linear regression model.

$$D = \{(x_i, t_i)\}_{i=1}^{N} \tag{1}$$

Parameterize y with w:

$$y = y(x, w) \tag{2}$$

For input vector, the prediction is:

$$y_n = y(x_n, w) \tag{3}$$

The model is defined as:

$$y(X, W) = \omega_0 + \omega_1 x_1 + \cdots + \omega_D x_D \tag{4}$$

Then, extending the formula by using the matrix to represent a set of fixed non-linear functions of the input vector:

$$y(X, W) = \sum_{j=0}^{M-1} \omega_j \phi_j(X) = W^T \phi(X) \tag{5}$$

The above formula is aimed to find the appropriate (1⩽i⩽D) parameter for the linear regression model. The actual class values $(y_1, y_2, \dots y_n)$ will be approximately equal to the predicted y_n values. Linear regression algorithm has strong data processing speed, it doesn't require complex calculations, so it is easy to understand and could run fast even with large amounts of data. What's more, when some new data was added, it was easy to update the model.

After importing the sklearn, we used the Linear Regression classifier which provided training and testing data. The dataset was divided into two parts, one part is used as the training data and accounts for 80% of the entire dataset, and the other is used as testing data and accounts for 20%. We trained the machine learning classifier by training data and got the accuracy of the model by testing data.

3. Result and discussion

This section examines the forecasting ability of Regression Models for stock data. Then, we will illustrate the prediction results of the selected machine learning model based on the Hong Kong Stock

Exchange. The following Table 1, Table 2 and Table 3 demonstrate the accuracy and the forecast for the stock's Nominal price of these three companies for the next sixteen days. In addition, the stock price trend of these companies is also shown in Figure 1, Figure 2, Figure 3 and Figure 4. From the table, the accuracy of the model can reach more than 85%. The folding line chart shows the stock price trend from the beginning of 2020 to 2023. It is a period that included the outbreak of COVID-19, the implementation of epidemic prevention and control policies, and the lifting of the control policy. In this chart, the red line represents the past stock price fluctuations, and the blue one represents the future stock price forecast.

Table 1. Stock price prediction for Alibaba.

Company	Predicted results				Accuracy
Alibaba	83.113	80.948	87.004	82.987	0.912
	87.829	83.407	83.725	84.368	
	99.642	105.993	102.931	115.844	
	110.544	118.839	113.713	112.415	

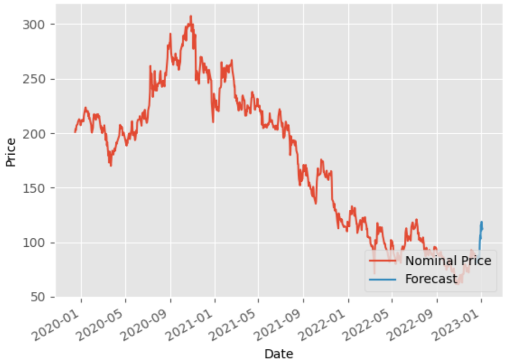


Figure 1. 2020.01-2023.01 Alibaba stock price trend.

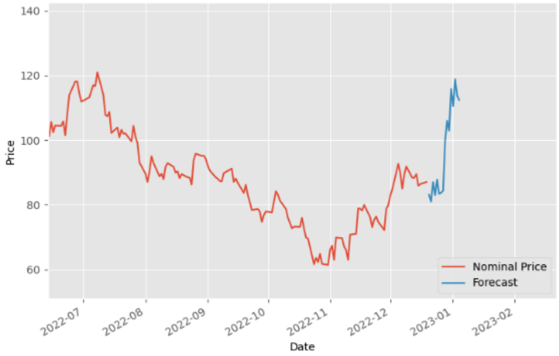


Figure 2. 2022.07-2023.01 Alibaba stock price trend.

Table 2. Stock price prediction for Tencent.

Company	Predicted results				Accuracy
Tencent	317.570	317.158	331.954	324.483	0.893
	335.034	340.295	343.000	346.762	
	369.789	361.420	358.326	367.628	
	365.678	383.966	370.260	368.247	

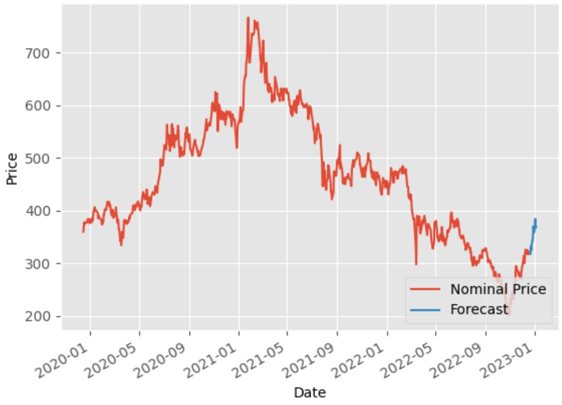


Figure 3. 2020.01-2023.01 Tencent stock price trend.

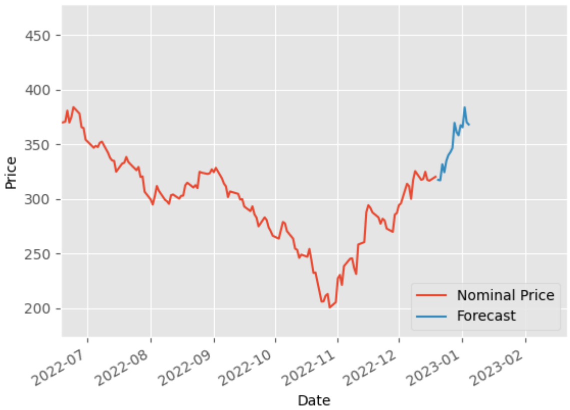


Figure 4. 2022.07-2023.01 Tencent stock price trend.

Table 3. Stock price prediction for Xiaomi.

Company	Predicted results				Accuracy
Xiaomi	10.670	10.647	11.178	10.881	0.876
	11.454	10.987	11.217	10.926	
	11.667	11.954	11.480	12.179	
	12.158	11.980	11.797	11.913	

The overall trend for these three companies revealed a fluctuating pattern with initial growth and subsequent decline in the stock prices of these companies. Notably, the peak period for all three companies was observed between the end of 2022 and the beginning of 2021. Both Tencent and Xiaomi stock prices have peaked January in 2021 but the Alibaba’s peak occurred November in 2020. It is worth noting that, in comparison to the other two companies, Alibaba's stock price exhibited no significant growth, and after reaching the peak, it demonstrated a downward trend. It drops to the lowest point at the end of 2022, at this point the stock price is just 1/4 of it was at the beginning of 2020. After that, it began to rise again, and the predicted outcome indicates an upward trajectory. Regarding Tencent, it showed a trend with first steady fluctuations and then growth before reaching the highest point of the price. The price at the peak was twice that of the beginning of 2020. After that, it began to fall until the end of 2022. But there is some difference in the period of the second half of 2021, its stock price was relatively stable, it showed some fluctuations, but the average value remains steady. Since November 2022, the price had been increasing including the prediction result. For Xiaomi, in the early period, it revealed dramatic growth from the beginning of 2020 to January 2021, during this period it occurred a little peak in September 2020. And the highest price was up to 2.5 times as the beginning. Subsequently, it began to decline, but the price also recovered in the middle of 2021. For the forecasted prices, it shows an upward trend. In addition, for the lowest point of the price, both Alibaba and Tencent dropped to a level lower than the price at the beginning, but Xiaomi dropped at the same as the beginning.

The coronavirus outbreak in early 2020 was recognized as a "public health emergency" by the WHO and posed a serious challenge to the country's economy, with most industries being strongly affected by the outbreak [9]. Due to the epidemic policy, it can be observed that the epidemic policy has led to an overall downward trend in the stock prices. The transmission mechanism of the COVID-19 epidemic to the stock market was the contagion effect of negative investor sentiment, which led to a resonance in the share prices of companies in different regions, resulting in a general decline in the stock market [10]. However, as China lifted the measure imposed to control the epidemic at the end of 2022, there has been a noticeable increase in the stock prices of all three companies. Therefore, it can be concluded that for the coming period, the stock prices of all three companies show a volatile increase, which is a very good thing for the recovery and development of the Chinese economy. Table 4 presents the general trends of the three companies.

Table 4. The general trends of the three companies.

Company Trend	Alibaba	Tencent	Xiaomi
Past	Small growth and significant decline	Volatility growth volatility decline	First, grow and then fall back to the original level
Future	Volatility rises	Volatility rises	Volatility rises

4. Conclusion

In this work, we intend to use a training linear regression model to predict and analyze the price trend of scientific and technological stocks in a period of high volatility after the unpacking of China's COVID-19 Epidemic Economic Policy at the end of 2022. We trained a linear regression model, trained the dataset after data preprocessing, and performed visualization and accuracy calculations. The experimental results show that the advanced linear regression model in this paper can predict the price trend of technology stocks in a period of high volatility after the policy is unsealed with high accuracy and can reflect the price volatility to a certain extent. From the visualized image, it can be seen that due to the impact of China's previous epidemic policies on the continuous decline of stock prices, the policy lifting has enabled stocks to recover in a short period. In the future, we plan to use more complex and sophisticated machine learning models for forecasting, so that the prediction results can reflect more accurate fluctuations.

References

[1] Ibbotson R G and Sinquefield R A 2010 Stocks, Bonds, Bills, and Inflation: Year-by-Year Historical Returns (1926-2008) (Chicago: Morningstar)

[2] Qiang Z 2020 Impact of the global spread of the new crown pneumonia epidemic on China's trade development and countermeasures vol 15 (Chinese: Business Economics Research) p 4

[3] Al-Blooshi L and Nobanee H and Nobanee H 2020 Applications of Artificial Intelligence in Financial Management Decisions: A Mini-Review

[4] Perold A F 2004 The Capital Asset Pricing Model (Journal of Economic Perspectives) vol18 pp 3-24

[5] Naifu C and Richard R and Stephen A R 1986 Economic Forces and the Stock Market vol 59 (The Journal of Business) chapter 3 pp 383-403

[6] Yoon Y and Swales G 1991 Predicting Stock Price Performance: A Neural Network Approach// Twenty-fourth Hawaii International Conference on System Sciences (IEEE)

[7] Lee M C 2009 Using support vector machine with a hybrid feature selection method to the stock trend prediction vol 36 (Expert Systems with Applications) chapter 8 pp 10896-10904

[8] Şahın D Ö and Akleylek S and Kiliç E 2022 LinRegDroid: Detection of Android Malware Using Multiple Linear Regression Models-Based Classifiers vol 10 (IEEE) pp 14246-14259

[9] DuanYou Y 2020 The impact of the COVID-19 epidemic on China's stock market - an empirical analysis based on the pharmaceutical industry vol 18(China: China Business News) p 3

[10] Hong X and Hongxia P 2021 The Impact of the COVID-19 Epidemic on the Chinese Stock Market - A Study Based on the Event Research Method vol 7 (China: Financial Forum) p 11

Proceedings of the 5th International Conference on Computing and Data Science
DOI: 10.54254/2755-2721/15/20230842

Prediction of carbon dioxide emissions based on machine learning algorithms

Zhihuang Chen

The department of Computer Science, California State University, Fullerton, Fullerton, CA 92831-3599, the United States

zhchen@csu.fullerton.edu

Abstract. The escalating emergence of environmental issues, including the greenhouse effect and coral bleaching, has raised global awareness of the significance of sustainable development and preservation of the earth's resources. Although reducing emissions is essential to mitigate the adverse effects of greenhouse gases, it remains challenging to detect without specialized equipment. This constraint is particularly burdensome for small organizations and individual groups due to the high associated costs. Therefore, this study proposes using machine learning algorithms and common vehicle attributes to predict greenhouse gas emissions accurately. Specifically, the research employs the random forest algorithm, incorporating vehicle power parameters to predict carbon dioxide emissions. The study employs Mean Square Error (MSE), Root Mean Square Error (RMSE), and R-squared metrics to analyze the model's effectiveness, accuracy, and feasibility in predicting greenhouse gas emissions. This approach will enable small groups to participate in environmental protection efforts, democratizing the process for all who desire to safeguard the environment.

Keywords: machine learning, random forest, prediction of carbon dioxide emissions.

1. Introduction

With climate change caused by environmental problems worldwide, individuals are increasingly aware of the need for environmental protection. A 2020 survey from Ipsos found that 71% of people globally believe the world is facing a climate emergency, up from 58% in 2019 [1]. Among them, carbon dioxide is one of the main gases causing environmental problems. From the Issues and Local Programs page on the Washington State Department of Ecology's official website, Carbon dioxide emitted by vehicles is the most common anthropogenic greenhouse gas, according to an article by the Washington State Department of Ecology advocating for environmental protection [1]. In other words, humans can effectively reduce corresponding environmental problems by reducing the carbon emissions of vehicles. Today, relevant departments and automakers in many countries have taken up the challenge of reducing vehicle carbon emissions by analyzing data from sensors and other sources combined with machine learning to predict and optimize a vehicle's carbon emissions.

Machine learning, a subfield of artificial intelligence that uses algorithms and statistical models that allow computer systems to learn and improve from experience, has already significantly reduced carbon emissions from vehicles. For example, machine learning can optimize the vehicle's engine parameters to reduce carbon emissions by analyzing the patterns and correlations among factors such as vehicle performance, weather, traffic conditions, and driving habits. This data can help vehicle manufacturers develop more economical and environmentally friendly vehicles. Machine learning has undeniable potential to help reduce vehicle carbon emissions. By analyzing data from sensors and other sources, machine learning algorithms can optimize vehicle performance, reduce fuel consumption, and minimize carbon emissions. As the automotive industry continues to adopt machine learning, it's not hard to imagine a significant reduction in carbon emissions from cars in the coming years. At the same time, many companies have also launched projects and services that optimize vehicles through machine learning algorithms, such as Tesla [2], Ford [3], Toyota [4], BMW [5], and Volkswagen [6]. Despite the benefits of this method, a limitation exists in that the prediction results heavily depend on the accuracy and availability of sensor data. Obtaining and predicting carbon emission data is expensive for some small businesses, organizations, and the public without professional equipment. Protecting the earth's ecological environment is crucial to the sustainable development of human beings. It requires the joint efforts of all human beings, and the cost and hardware requirements make it more difficult for most people to participate in protecting the earth's ecological environment. Therefore, developing low-cost and accurate methods for predicting vehicle carbon emissions without professional equipment is an important research topic to promote broader participation in environmental protection.

This study aims to study the feasibility of predicting vehicle carbon emissions without relying on sensors based on vehicle data recorded by government departments and authoritative organizations combined with machine learning algorithms. In order to prevent unnecessary misleading to the users of the research results and maintain the objectivity of the research data, this research will not include the brand, model, vehicle manufacturing process, and technology of the vehicle. However, the selection of the characteristics of the project will take the dynamics of the vehicle and its basic characteristics as the main parameters of the study. Considering that the actual carbon emissions of vehicles may vary depending on the influence of uncertain factors such as vehicle parameters, driving environment, and driving habits, this research's machine learning algorithm model will use ensemble learning. The main advantages of ensemble learning over traditional single-model approaches are its flexibility and scalability. It can help reduce the impact of uncertainty and variability in data in situations where there is a high degree of uncertainty or variability in the data. In addition, it will make the results of this research more informative.

2. Method

2.1. Dataset preparation

In this study, the scikit-learn (sklearn) toolkit was utilized to process, analyze and evaluate the data. In terms of data collection, in order to ensure the authenticity and accuracy of the data. The data used in this study includes vehicle parameters and carbon emission data released by government departments and authoritative institutions. The sample data size is 12, 988, including common vehicles from 2019 to 2023 [7]. Among them, 11,589 were for conventional power vehicles (i.e. gasoline and diesel), 712 were for electric vehicles, 46 were for hydrogen-powered vehicles, 424 were for gasoline-electric hybrid vehicles, 424 were for biofuels, and 217 were for electric hybrid vehicles (i.e. fuel ethanol and electricity). Due to the different fuels the vehicles use, carbon emissions, and various attributes vary widely and go unrecorded. For example, the carbon emissions of pure electric vehicles are always zero, and the engine parameters of biofuel vehicles are not well documented. Considering that these factors may have adverse effects on the results, and this study aims to accurately predict the feasibility of vehicle carbon emissions by analyzing data without professional sensors, this study will only use conventionally powered vehicles as Training data, including gasoline and diesel vehicles. Therefore, the actual usage data is 11, 589.

Furthermore, in order to identify meaningful parameters in the data. Linear regression was used to analyze the relationship between vehicle parameters and CO2 emissions. In the obtained results, it is found that there is a certain linear relationship between the carbon dioxide emission of the vehicle and the power and fuel consumption of the vehicle. Therefore, in the feature selection of the model, these data with a linear relationship are used as the main imported data. For some missing data, this study uses a module in the scikit-learn library to perform mean imputation on missing data. Consider that the ordering of the data in the original data is random. Therefore, using the mean value of the data to replace missing values can keep the overall distribution characteristics of the data unchanged, thereby minimizing the error caused by missing data. For data type conversion, based on the algorithm features used in this research, the data type is uniformly converted to real numbers (i.e. float).

2.2. Machine learning models

This study aims to predict vehicle carbon emissions through vehicle parameters. Given that the cited data contains missing and uncertain values, the algorithm of choice for this study is the random forest algorithm within the supervised learning machine learning framework [8-10]. Random forest is an ensemble learning method based on decision trees and is widely used in machine learning due to its ability to enhance model accuracy and stability by combining multiple decision trees. Also, random forests are less susceptible to noisy data. It goes through a decision forest consisting of multiple decision trees, each consisting of randomly selected features and samples. At the same time, the random forest algorithm can deal with missing data and effectively deal with missing values in the data set. The working principle of the random forest is that its training set is generated by bootstrap sampling, and features are randomly selected in the new training set to build a decision tree. Using each decision tree to predict new sample data after building a certain number of decision trees. The results from each tree are then aggregated using voting or averaging methods to arrive at a final prediction. Random forest reduces the variance of the decision tree by randomly selecting features and samples, thereby improving the generalization ability of the overall model. Therefore, the advantage of using the random forest algorithm in this study is that it can effectively reduce the negative impact of a series of uncertain factors, such as different technologies of automobile manufacturers, thereby improving the accuracy of the research results.

3. Result and discussion

Through random forest algorithm model analysis, the mean square error obtained in this study is 40.4801, the root mean square error is 6.3624, and the R-squared is 0.9961 (Figure 1). The possible values of the target variable (Comb CO2) ranged from 151 to 979, with a mean of 407.2 (Figure 2). A Mean Square Error (MSE) of 40.48 indicates that the model's predictions are off by about 6.36 units (average square root of 40.48). This value appears relatively small compared to the range of possible values for the target variable.

Second, the Root Mean Square Error (RMSE) of 6.36 is also relatively small compared to the mean and range of possible values of the target variable, which indicates the model’s performance. Also, an R-squared value of 0.996 indicates that the model can explain most of the variance in the target variable. In other words, the model's predictions are highly correlated with the actual value of the target variable. It can be seen from Figure 2 that the direct relationship between the predicted value and the actual value is linear, and the image is close to a straight line. This indicates that the model's predictions are similar or the same as the actual values. Furthermore, most of the deviations in the predicted values of the model are concentrated around zero, as illustrated in Figure 3. This finding indicates that the model's predicted results closely match the actual values, underscoring its suitability for reference in this study's data analysis.

Overall, a MSE of 40.48 is deemed acceptable for this study, indicating that the model's predictions are usually accurate. Secondly, the RMSE of 6.36 is also within the acceptable range, the deviation is small, and the model has an excellent R-squared value (0.996). Based on these metrics and the context, the random forest model performs remarkably well on this dataset. Therefore, predicting the vehicle’s carbon emissions through the random forest algorithm model is feasible.

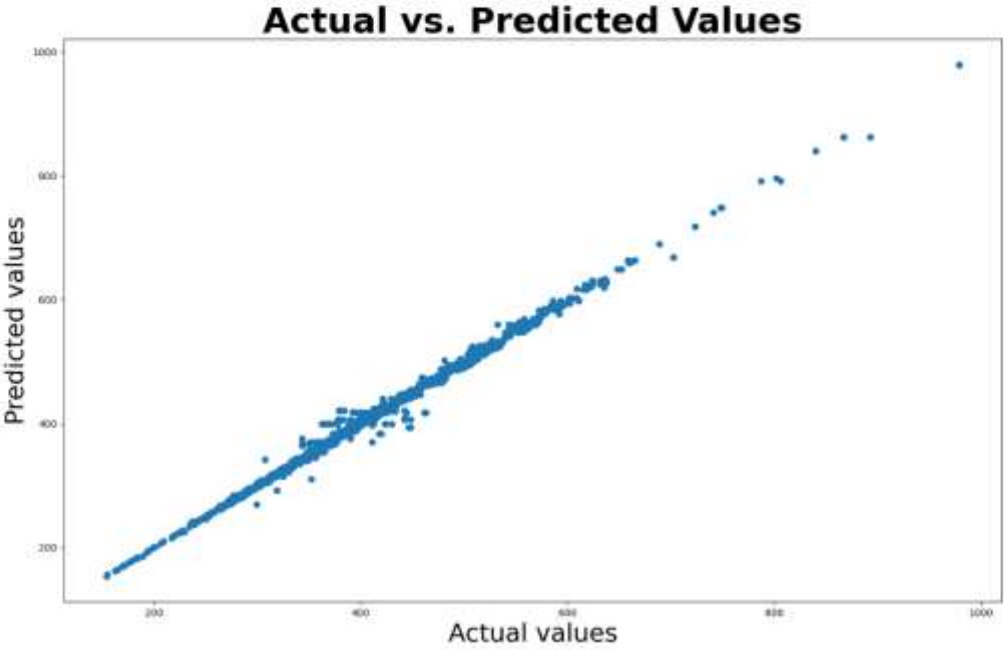


Figure 1. The relationship between actual and predicted CO2 emissions.

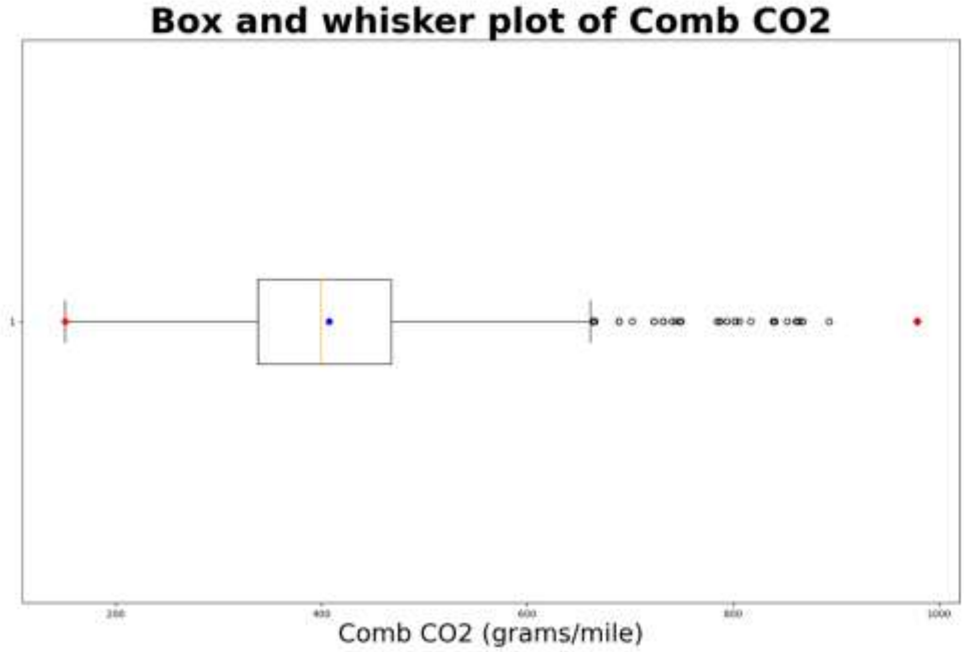


Figure 2. The statistics for the sample data.

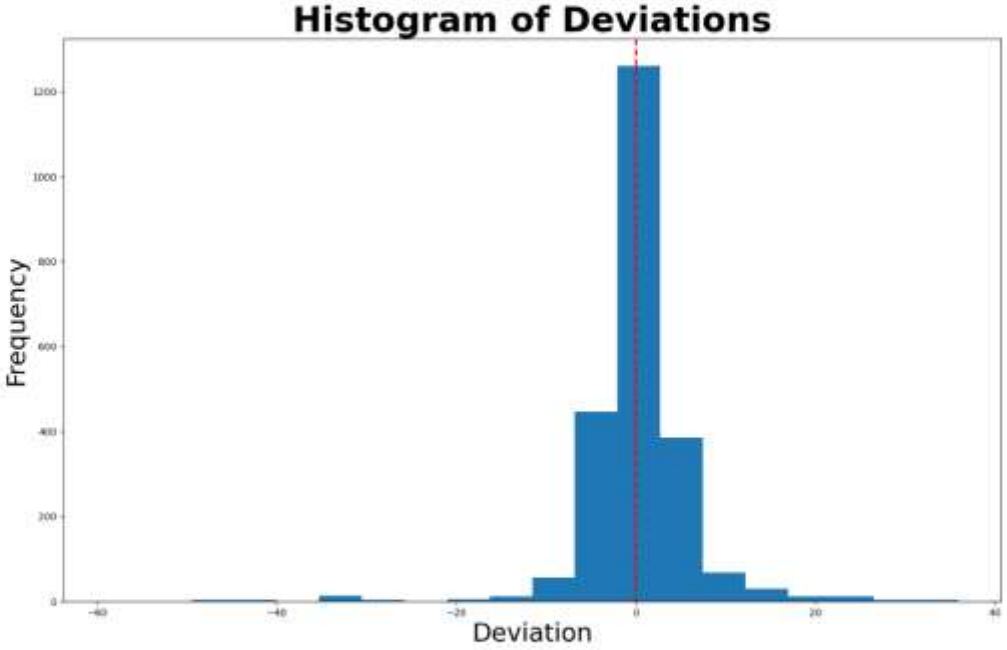


Figure 3. The deviation value of the predicted data.

4. Conclusion

In this study, the purpose of the research is to test the hypothesis that there is a way to help all those who want to participate in environmental protection to reduce the limitations and costs. In the experiment, carbon dioxide, one of the common greenhouse gases, was used as the predicted target for research and analysis. The results of this experiment show that it is feasible to predict carbon dioxide emissions by combining common vehicle attributes with the random forest algorithm. Furthermore, this study's MSE, RMSE, and R average prove that the sample data fit the algorithm model well. It also means that the method has the potential to predict other greenhouse gases. For subsequent improvements, it will be considered to apply the algorithm to the prediction of other greenhouse gases and add some factors that may affect the prediction results, such as the year of the vehicle, road conditions, and the type of vehicle fuel.

References

[1] Reducing air pollution from cars 2023 Reducing car pollution - Washington State Department of Ecology Retrieved April 5 2023 from https://ecology.wa.gov/Issues-and-local-projects/Education-training/What-you-can-do/Reducing-car-pollution

[2] AI &amp; Robotics 2023 Tesla. Retrieved April 5, 2023, from https://www.tesla.com/AI

[3] Ford Establishes Latitude AI to Develop Future Automated Driving Technology 2023 Ford Media Center. Retrieved April 5, 2023, from https://media.ford.com/content/fordmedia/fna/us/en/news/2023/03/02/ford-establishes-latitude-ai-to-develop-future-automated-driving.html

[4] Toyota automated driving. 2023 Retrieved April 5, 2023, from https://amrd.toyota.com/app/uploads/2022/02/ATwhitepaper.pdf

[5] BMWK - Federal Ministry for Economics Affairs and Climate Action. 2023 AI-based solution for optimizing the energy efficiency and consumption of electric vehicles BMWK Retrieved

April 5, 2023, from https://www.bmwk.de/Redaktion/EN/Artikel/Digital-World/GAIA-X-Use-Cases/77-gaia-x-decentralized-in-vehicle-mlaas-to-ev-energy-efficiency/use-case.html

[6] Volkswagen autonomous driving in Hamburg 2022 Volkswagen Group Retrieved April 5, 2023, from https://www.volkswagenag.com/en/news/stories/2019/04/laser-radar-ultrasound-autonomous-driving-in-hamburg.html

[7] Energy 2023 Fuel Economy Data https://www.fueleconomy.gov/feg/download.shtml

[8] Biau G Scornet E 2016 A random forest guided tour Test 25: 197-227

[9] Rigatti S J 2017 Random forest Journal of Insurance Medicine 47(1): 31-39

[10] Oshiro T M Perez P S Baranauskas J A 2012 How many trees in a random forest? Machine Learning and Data Mining in Pattern Recognition: 8th International Conference MLDM 2012 Berlin Germany July 13-20 Proceedings 8 Springer Berlin Heidelberg 154-168

Proceedings of the 5th International Conference on Computing and Data Science
DOI: 10.54254/2755-2721/15/20230843

Cross-platform spam messages classification based on the multiple machine learning algorithms

Mengliang Tan

The Department of Applied Mathematic, University of California, San Diego, America

metan@ucsd.edu

Abstract. The proliferation of spam messages has had a detrimental impact on users' experience of emails and social media. Consequently, it is imperative to implement effective spam filtering mechanisms to enhance online experiences. Internet companies have leveraged machine learning algorithms to detect and thwart spam messages. Given the multitude of popular social media platforms, it is critical to evaluate the efficacy of prevalent machine learning algorithms across diverse online platforms. This study seeks to assess the performance of Support Vector Machine, Linear Regression, and Random Forest on social media. To this end, datasets containing spam and non-spam messages sourced from YouTube comment sections and Twitter will be procured. The text data will be transformed using a vectorizer to enable interpretation by machine learning models. Three models employing SVM, Linear Regression, and Random Forest will be trained and deployed to test their effectiveness. The models will be applied to detect spam messages in the test dataset, YouTube comment set, and Tweet set. The performance of the models will be evaluated based on accuracy, F1 score, and precision score. The findings indicate that the models' performance on various social media datasets is not satisfactory, as there is a significant reduction in accuracy.

Keywords: brain tumour, MRI, CNN, machine learning, deep learning.

1. Introduction

Spam emails, which are unwanted emails sent in bulk to users' mailboxes, pose significant challenges to users' email experience. These emails typically arrive in large quantities and occupy considerable space in the users' inboxes, often disrupting their ability to access important messages. It has been easier for spammers to get their hands on personal information such as email addresses. It can be leaked to spammers through computer viruses, information leaks from big internet companies, some websites that force you to submit your personal information to them, etc. That is why, in recent years, the volume of spam emails has been continuously increasing. Spam messages are responsible for 45.1% of the world’s email traffic by March 2021 [1]. Hardware efficiency can also be prevented from being fully utilized by having a large inflow of spam emails taking up storage space and CPU. Some spam messages may contain Trojan downloaders in the mail, which will be disguised as the file of bills, then it will implement viruses in the users’ computers after being downloaded [2]. These spam messages can also be used as a way for scammers to conduct fraudulent practices such as cheating individuals into sharing sensitive personal information like passwords, Bank Verification Numbers, and credit card numbers, which can cause actual financial loss [3].

In contemporary times, machine learning has emerged as a dynamic and diverse field. Numerous techniques have been put into use by major inter companies such as Google and Yahoo which are two of the biggest email services providers. The role played by Machine Learning algorithms in this context is to generate rules for spam filters to recognize spam and non-spam messages. The way to achieve this is that algorithms will be “fed” with training samples, and from these samples, they will learn the pattern of text data and recognize the rules of classification. After years of development and progress in the field, the power of spam filtering keeps increasing. For example, the machine learning model deployed by Google can reach 99% accuracy in filtering out spam emails [3].

In 1995, Support Vector Machine (SVM) created by Vapnik has later been widely applied and performed greatly in many areas such as function approximation, modeling, optimization control, and binary classification which is essential for spam filtering [4]. The formulation of SVM utilizes the Structural Risk Minimization principle, which minimizes an upper bound on the expected risk and has been proven to be better than the traditional Empirical Risk Minimization principle which works by minimizing the error of the training data [5]. Logistic Regression is a very commonly used machine learning algorithm with very low time complexity [6]. For its capability in data analysis, a lot of software companies have a Logistic regression model as their products’ innate feature [7]. Random forest is also a very popular machine-learning algorithm. The Random Forest algorithm was created by Breiman and Cutler, and it was extremely well at handling massive data sets even under the circumstance when sets got missing data [8].

Despite the fact model trained by these machine learning algorithms has shown excellent performance, researchers have paid less attention to the accuracy and efficiency of machine learning’s prediction of spam comments across different groups of data sets. With the proliferation of social platforms and the increasing amount of text data being transmitted online, the need for machine learning models that can recognize unwanted messages across different platforms has grown. This research will mainly focus on using Support Vector Machine, Logistic regression, and Random forest as the algorithm to train machine learning models with various data sets of text messages from different social platforms such as Twitter and YouTube comment sections and evaluate their performance.

2. Method

2.1. Dataset Preparation

In this study, six data sets provided by Kaggle were utilized [9-13]. All data sets were in the form of comma-separated values files. The content of these files included messages that were labeled as spam and non-spam from Twitter and YouTube comment sections. There were two Twitter sets with one having 11787 messages and another one having 5169 messages. For YouTube comments, there were three sets. Two of them were comments from videos related to Kate Perry and Eminem. Kate Perry set had 348 messages. Eminem set had 446 messages. The third one had 4673 messages. The remaining data set would be the spam training set which would be used to train the machine learning model. It had 5169 messages. It is noteworthy that all of the data sets were binary.

In terms of the processing part of this study, all six csv. Files would be read into data frames by using pd.readcsv(). 4470 messages of the spam training set would be used as the training set, and the rest was for testing the model. Then, it was needed to merge two Tweets sets and three YouTube comments sets. In YouTube comments message sets, columns of comment IDs, dates, and authors were not needed. Hence, they would be first removed by using pandas.drop() and had columns of class and content swapped for further integration. After this, pandas.merge() method was used here to merge all three datasets. A similar process would be applied to two other two data sets of Tweets. Columns not related to the work would be removed and then two sets would be merged. After the integration, drop_duplicates() was utilized to remove repeating messages if they existed. Now there were three sets: the spam training set, the integrated YouTube comment set, and Tweet set.

The machine clearly could not directly interpret the text data of the files. To solve this, CountVectorizer() would be applied here to create a vectorizer to transform text data into numerical data, thereby the machine could count the number of presences of different words in the file. For the step of vectorization, the fit_transform() method was utilized to fit the vectorizer to the training data and also transform text data. This vectorizer would be saved.

2.2. Machine learning model

Support Vector Machine is a machine learning method that is good at dealing with classification and regression problems, and it uses hypothesis space of a linear function in a high dimensional feature space, trained with a learning algorithm from optimization theory that implements a learning bias derived from statistical learning theory [5]. SVM functions in the classification process by seeking a hyperplane to distinguish between points belonging to the first and second classes. Furthermore, it also maximizes the distance between points of different classes and the hyperplane. Random forest performs binary classifications by constructing decision trees for class prediction and then the output would be determined by choosing the class that gotten chosen by the most number of trees [14]. Logistic regression analyzes the relationship between several existing independent variables and produces a binary variable based on the interactions between the different variables [15].

Following the preprocessing part, the spam training set was divided into a training part and a testing part. For further implementations, scikit-learn, a machine learning library for python, would be imported.

After this, a model would be created utilizing the Support Vector Machine algorithm. To improve the accuracy of the algorithm, the hyperparameters for the SVM model were tuned to optimize the machine learning process this study just created. RandomizedSearchCV would be implemented here. There were many hyperparameters, and not all of them would be adjusted. RandomizedSearhCV was a fast way to perform hyperparameters tuning. C-parameters and gamma parameters would be tuned to find the most optimized decision boundary which could separate data into two classes as accurately as it could while preventing the model from being overfitting. Upon completing hyperparameter tuning, the training data set was used to train the model using fit(). After training was completed, the accuracy score, precision score, and F1 score, of the model would be first evaluated on the testing part of the spam training set, then on YouTube's comment set and Tweet set.

RandomForestClassifier() would be used to create a Random forest model. N_estimators would be set to 100 and random_state set to 42. Since this process was done in a different file, the previously saved vectorizer would be loaded, and then directly transform the text data of the training part. LogisticRegression() was used to create the Logistic regression model. These two models would then be tested like how it was done on the SVM model previously.

3. Result and discussion

After applying Support Vector Machine, Random Forest, and Logistic regression algorithms on three data sets, the experiment have shown metrics that reflect the performance of different machine learning algorithms on data from different online social platforms. In Table 2, The Random Forest model managed to reach 100% precision. As shown in Tables 1, Table 2, and Table 3, SVM had the highest accuracy and F-1 score on all three sets.

Table 1. Performance of the SVM model on different data sets.

Metrics of Performance	Spam-Training set	YouTube's comment set	Tweet set
Accuracy	0.985	0.577	0.650
Precision	0.978	0.895	0.684
F1-score	0.937	0.219	0.246

Table 2. Performance of the Random Forest model on different data sets.

Metrics of Performance	Spam-Training set	YouTube's comment set	Tweet set
Accuracy	0.974	0.541	0.646
Precision	1.000	0.807	0.705
F1-score	0.881	0.083	0.241

Table 3. Performance of the Logistic regression model on different data sets.

Metrics of Performance	Spam-Training set	YouTube's comment set	Tweet set
Accuracy	0.983	0.563	0.646
Precision	0.978	0.875	0.759
F1-score	0.930	0.170	0.178

The study's results, as presented in Tables 1, Table 2, and Table 3, indicated that the three models performed exceptionally well in terms of their accuracy, precision, and F1 score, with minimal disparities between their metrics. However, when handling datasets from Twitter and YouTube comment sections, all three models experienced a significant decline in their performance metrics, particularly in F1 score and accuracy. The F1 score, which combines the precision and recall scores to evaluate the models' ability to identify true positives while minimizing false positives and false negatives, revealed a marked discrepancy between the precision score and F1 score for all three models, with the former being significantly higher. The Random Forest model exhibited the largest drop in F1 score when applied to the YouTube comments dataset, indicating potential instability when encountering various datasets. While the other two models also displayed low F1 scores, they remained relatively consistent. Interestingly, precision exhibited the smallest decline compared to the other metrics for all three models, implying that the models' ability to avoid false positives was less affected during cross-platform examinations.

Comparing all three models, it is indicated that cross-platform examinations can have a huge influence on their performance. None of them produced an optimistic performance, and none of them has a huge advantage over the other two models. The SVM model is slightly better than the other two since its performance is relatively consistent on different datasets and has a higher F1 score which means more balanced in avoiding false negatives and false positives.

4. Conclusion

This study investigated the performance of three machine learning models, namely SVM, Logistic regression, and Random forest, on various social media platforms. The models were tested on different datasets obtained from social media platforms, and various metrics of binary classification were evaluated to assess their accuracy and efficacy. The proposed method was validated through a series of experiments, and the findings revealed that the performance of machine learning models on social media platforms significantly dropped when the models were not previously trained using datasets from the corresponding social platform. The results indicated no significant differences between the performance of the three models, with SVM exhibiting a slightly superior performance. However, the study could have benefited from a more diverse range of social media datasets beyond Twitter and YouTube. Future research will focus on improving the accuracy of machine learning algorithms when handling data from other platforms.

References

[1] Kudupudi N NAIR S 2021 Spam message detection using logistic regression International Journal of Advanced Computer Science and Applications 9(9): 815-818

[2] Spam Report 2016 https://media.kasperskycontenthub.com/wp-content/uploads/sites/43/2016/08/07185333/Spam-report_Q2-2016_final_ENG.pdf

[3] Dada E G Bassi J S Chiroma H et al. 2019 Machine learning for email spam filtering: review, approaches and open research problems Heliyon 5(6): e01802

[4] Hsu W C Yu T Y 2010 E-mail Spam Filtering Based on Support Vector Machines with Taguchi Method for Parameter Selection J. Convergence Inf. Technol 5(8): 78-88.

[5] Jakkula V 2006 Tutorial on support vector machine (svm) School of EECS, Washington State University 37(2.5): 3

[6] Mrisho Z K Ndibwile J D Sam A E 2021 Low Time Complexity Model for Email Spam Detection using Logistic Regression International Journal of Advanced Computer Science and Applications 12(12)

[7] Berrou B K Al Kalbani K Antonijevic M et al. 2023 Training a Logistic Regression Machine Learning Model for Spam Email Detection Using the Teaching-Learning-Based-Optimization Algorithm Proceedings of the 1st International Conference on Innovation in Information Technology and Business (ICIITB 2022). Springer Nature 104: 306

[8] Reddy K N Kakulapati V 2021 Classification of Spam Messages using Random Forest Algorithm Resesearchgate

[9] Kaggle 2018 https://www.kaggle.com/datasets/goneee/youtube-spam-classifiedcomments?select=Youtube02-KatyPerry.csv

[10] Kaggle 2017 https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset

[11] Kaggle 2023 https://www.kaggle.com/datasets/greyhatboy/twitter-spam-dataset

[12] Kaggle 2021 https://www.kaggle.com/datasets/fahmisulthoni/tweet-spam

[13] Kaggle 2022 https://www.kaggle.com/datasets/madhuragl/5000-youtube-spamnot-spam-dataset

[14] Kontsewaya Y Antonov E Artamonov A 2021 Evaluating the effectiveness of machine learning methods for spam detection Procedia Computer Science 190: 479-486

[15] Lawton G et al. 2022 What Is Logistic Regression? - Definition from Searchbusinessanalytics Business Analytics, TechTarget, 20 Jan. https://www.techtarget.com/searchbusinessanalytics/definition/logistic-regression.

Spam filter based on naive bayes algorithm

Mengyuan Han

The department of International Software Engineering, Dalian University of Technology, Dalian, China

983822294@mail.dlut.edu.cn

Abstract. The widespread use of Electronic Mail (E-mail) has led to a significant increase in spam, which has severely impeded the growth and well-being of the Internet. To mitigate this issue, the implementation of email filtering techniques has become necessary, requiring the use of specific technological tools. Presently, the K-Nearest Neighbor (KNN), Support Vector Machine (SVM), and Naive Bayes (NB) algorithms are commonly used in probability statistical classification methods for email filtering. Among these, the NB algorithm is the most classical, with its rich mathematical theory as the basis, high classification efficiency, and straightforward algorithmic approach. However, the algorithm relies on the conditional independence assumption, making the accuracy susceptible to the correlation between attributes. This study focuses on email filtering techniques based on the NB algorithm, conducting experiments to evaluate the classification accuracy and proposing feasible improvements to weaken the independence assumption. The experimental results demonstrated the effectiveness of the employed method.

Keywords: spam filter, naive bayes algorithm, machine learning.

1. Introduction

With the continuous advancement of the Internet, E-mail has gradually become one of the common means of communication in people's daily life. However, the resulting spam has occupied a large number of network resources occupied and disrupted the normal order of E-mail communication, which has gradually become a significant problem in the work of Internet governance.

Spam can be strictly defined as any unwanted mail that does not adhere to the recipient's acceptance preferences, encompassing advertisements, unsolicited electronic publications, promotional materials, and other forms of communication that the recipient has not requested before, as well as mails that cannot be rejected, those that conceal the sender's identity or address, and mails containing false information about the source, sender, or route. Spam imposes many harms, including occupying a large number of network resources, seriously affecting the normal mail service; Dealing with them is time-consuming, inefficient and frustrating; Be used by hackers, causing the attacked website network paralysis; Spreading harmful information, causing harm to the real society, etc.

According to the report issued by Symantec, an international authoritative network security company [1]. It shows that the global spam accounted for 87.4% of the total number of emails in 2009. Since 2007, spam has risen by an average of 15%. At present, China has become one of the most seriously harmed countries in the world by spam. According to a survey by the ITU, in 2003 alone, the cost of dealing with junk mail reached 4.8 billion yuan, and in 2006, the annual loss caused by junk mail to the national economy has exceeded 10 billion yuan.

Common spam prevention measures include IP blacklist and whitelist, Real-time Blacklist List (FBL) and mail filtering. Filtering technology works by following an algorithm or rule to determine whether an email is spam or not. The original mail filtering technology used pattern matching algorithm rules, such as the search of keywords to find spam. Furthermore, regular expression is used to realize fuzzy matching. With the development of technology, people begin to use spam recognition algorithms, such as Bayesian algorithm, which mainly originates from information classification. It is an algorithm for classification by calculating a posterior probability. Because Bayesian filtering operates purely according to statistical rules, and because tags are created solely by users, spammers have no way of guessing how their filters are configured to effectively block all types of spam.

This study will deeply explore the spam filtering technology based on the Park Subayes algorithm, and calculate the accuracy of classification through experiments, as well as explore the possible improvement.

2. Bayesian classification technique

The problem of spam filtering is actually a binary classification problem. At present, the most widely used is Bayesian classification technology, which is also a good spam filtering technology.

2.1. Bayes' theorem

Bayes' theorem is A theorem about the conditional probability of random events A and B [1]. It was proposed by the famous British mathematician Bayes in the 18th century. Its fundamental idea is to find the probability of another event when the probability of one event is known. In mathematics, the probability of event A occurring in the presence of another event B is not necessarily the same as the probability of event B occurring in the presence of another event A.

In situations where the occurrence probability of an event cannot be determined, it is possible to calculate the occurrence probability of an event related to it and, through the application of attribute correlation theory, infer the occurrence probability of the event itself. According to this thought description, Bayes theorem can be widely applied to text classification.

The formula of conditional probability is:

$$\mathrm{P(A|B) = P(A) * \frac{P(B|A)}{P(B)}} \tag{1}$$

It can be expressed as

posterior probability = prior probability * adjustment factor

According to the deformation, the formula of Bayes' theorem can be obtained as follows:

$$\mathrm{P(B_i|A) = \frac{P(B_i)P(A|B_i)}{\sum_{j=1}^{n} P(B_j)P(A|B_j)}} \tag{2}$$

2.2. Naive Bayes classifier

Bayesian classifier is a classifier based on Bayesian decision theory [2, 3], which is widely used in text classification at present. Text classification is to judge which category a text belongs to. When making classification, Bayes classifier calculates the probability of the posterior probability of the test text under different categories according to the samples of known text categories through Bayes' theorem, and compares the size of the posterior probability and selects the category of the value with a larger posterior probability as the category of the text. Bayesian reasoning model diagram can be found in Figure 1.

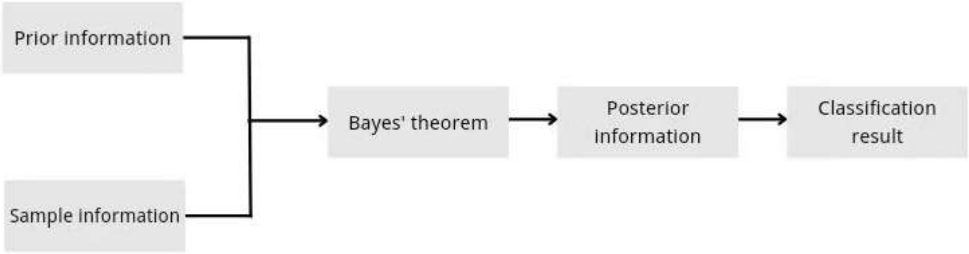


Figure 1. The procedure of the Bayes model.

In the field of classification, naive Bayes model is commonly used [4-5]. Naive Bayes algorithm is a certain improvement on the basis of Bayes algorithm, it is based on the conditional independence assumption between eigenvalues, the algorithm is very easy to understand, and has a high accuracy.

The difference between naive Bayes and Bayes is that it assumes that the attributes are independent of each other, which can be expressed by the following formula:

$$P(X|Y = c_k) = \prod_{j=1}^{n} P(x_j|y = c_k) \tag{3}$$

When determining the category, the following optimization formula is usually used:

$$y = \operatorname{argmax} P(Y = c_k) \prod_{k=1}^{n} P(X_k|Y = c_k) \tag{4}$$

3. The experiment

3.1. Training procedure

First, the collected data set is labeled with ham or spam, where ham represents normal mail and spam represents spam. Then the data in each mail is preprocessed, the basic process is: identify and cut the words in the mail -- > the words out of the cut lowercase processing -- > the length of less than 3 words and stop words out -- > count the number of remaining words. The flow chart can be found in Figure 2.

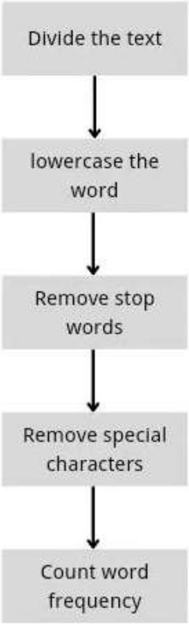
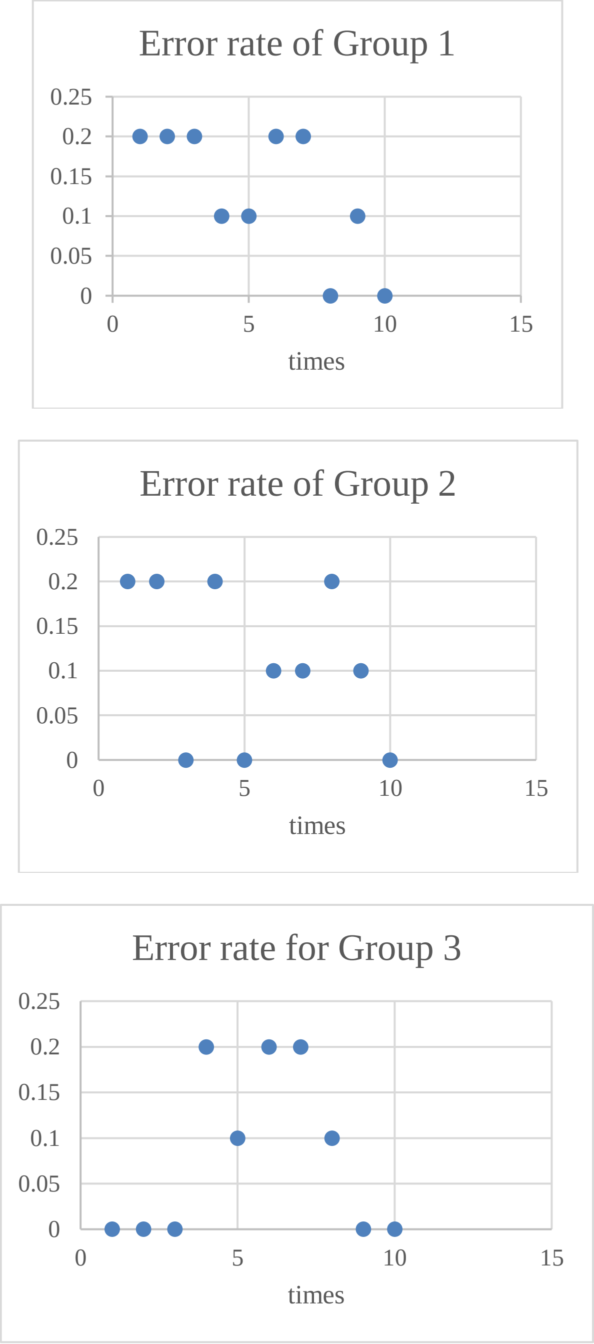


Figure 2. The procedure of the training process.

The classification is determined by naive Bayes algorithm. Calculate and compare the probability that the mail is classified as ham and spam. The higher probability is the category to which the mail belongs.

3.2. Test model

First prepare the data set, which contains 150 emails. Then divide the data set into 3 groups, with each group including 25 ham and 25 spam. From the 50 emails in each group, randomly select 40 for training and 10 as test set. Each group is tested for ten times. The results can be found in Figure 3 and Table 1.



Error rate of Group 1

Error rate of Group 2

Error rate for Group 3

Figure 3. The error rate of three groups.

Table 1. The performance of the model.

Group Number	Group 1	Group 2	Group 3
Average error rate	0.13	0.11	0.18

The average error rate in each group is between 0 and 20%(0.13,0.11,0.18 respectively), which means Naive Bayes classifier has quite high accuracy.

4. Algorithm analysis

4.1. Advantage and disadvantage

Because of its rich mathematical theory, naive Bayes algorithm has high classification efficiency and classification performance, especially based on the assumption of feature independence, which simplifies the form of Bayes classifier. In addition, naive Bayes algorithm can solve multiple classification problems, the algorithm is simple, suitable for incremental training.

However, the corresponding disadvantage is that it depends on the assumption of conditional independence among features, which makes the influence of each attribute on the global must be independent of each other. But obviously this assumption is not in line with reality and is difficult to satisfy. Therefore, when there are too many attributes or the attributes are correlated, the efficiency of naive Bayes classifier will be reduced. In contrast, when the number of attributes is small or the correlation is not strong, the performance of classification will reach the best.

4.2. Possible improvement: Naive bayes classifier based on random forest

Random forest is a group of tree structure classifier, the basic unit is decision tree. Since random forest is the idea of integrated learning, it makes up for the shortcomings of overfitting and low precision caused by a single tree [6]. As a combination classifier that integrates a variety of weak classifiers to form a strong one, the advantages of random forest are high accuracy, not easy to overfit, simple implementation, etc.

Bagging algorithm belongs to a classical ensemble learning algorithm, which is generally used to solve regression and classification problems. The weight of Bagging algorithm is the same, and data is extracted in the way of sampling with retractions. The method is to randomly extract n samples from the sample set for each training session. If s training sessions are performed, s training sets will be generated. The prediction result will be decided by the vote of s classifiers obtained by training.

Random forest algorithm uses Bagging method to carry out the selected samples with retractions and generate random samples of training sets for each tree [7-10]. Since each new training set is constructed on the original training set, there are differences among training sample sets. In the construction process of decision tree, the optimal segmentation on the selected random features is used to segment nodes, and the grown tree does not need to be pruned, so that the training error rate of random forest is low, and the anti-noise ability is strong.

5. Conclusion

In the present age of swift advancement of Internet technology, email has become an integral component of individuals' daily lives. The pressing need for anti-spam measures to purify the network environment necessitates the application of email filtering technology, which has been identified as one of the most efficacious approaches. This study primarily focuses on the widely used Naive Bayes filtering algorithm and experimentally verifies its high accuracy in email classification. Additionally, it proposes that the independence assumption can be weakened by incorporating the Random Forest algorithm. Despite an ever-increasing interest in spam mails, the constantly evolving email filtering technology can effectively mitigate the inundation of unwanted mails.

References

[1] Malakoff D A 1999 Brief Guide to Bayes Theorem Science 286(5444) 1461-1461

[2] Zhang K Chen X Song Y et al. 2019 improved method for TAN method Computing technology and Automation (in Chinese) (01):55-61

[3] Lin S Tian F 2000 Research on Bayesian classifier for data mining Computer Science (in Chinese) 27(10) 73-76

[4] Lu L Wang J Wang C 2017 Research on data mining Classification Algorithm for cloud computing Microcomputers and applications (in Chinese) 36(06):7-9

[5] Ma G 2018 The improvement and application of Naive Bayes Algorithm (in Chinese) Hefei: Anhui University

[6] Wang L 2020 Research on spam filtering based on Bayesian (in Chinese) Shanghai: Shanghai University of Engineering Science

[7] Liu Y Xing Y 2019 Research and application of text classification based on improved random forest algorithm (in Chinese) Computer system application 28(05):222-227

[8] Breiman L 2001 Random Forests Machine Learning 45(01):5-32

[9] Wang Y Xia S 2018 A survey of random forest algorithms for ensemble learning Information and Communication Technology (in Chinese) 12(01):49-55

[10] Biau G Scornet E 2016 A random forest guided tour Test 25: 197-227

Automated classification of brain tumors based on the convolutional neural network

Zhekai Jin

New York Institute of Technology, Anhui University, Hefei, 230000, China.

R32114010@stu.ahu.edu.cn

Abstract. The treatment of brain tumors, utilizing conventional methods such as surgery, radiotherapy, and chemotherapy, is limited in terms of accuracy and effectiveness. Furthermore, there exists a possibility of missing the diagnosis for small lesions and certain benign tumors with comparable density to normal tissue. To improve the precision and efficiency of brain tumor diagnosis, recent developments in artificial intelligence have been explored, including the use of Convolutional Neural Networks (CNNs). This research investigates the potential of a four-class CNN-based deep learning algorithm for the diagnosis of brain tumors. A dataset of MRI images, including various forms of brain tumors, underwent preprocessing and cleansing, and was subsequently classified into four categories. The CNN model trained to identify and diagnose MRI images achieved an 85.4% accuracy on the validation set. This study underscores the potential of CNNs to enhance the detection and precision of brain tumors, in addition to improving the consistency and dependability of diagnosis, thereby providing new leads for the discovery of novel therapies and medications. However, the study recognizes that limitations and areas of improvement exist in terms of dataset size, model architecture, and evaluation metrics.

Keywords: brain tumors prediction, Convolutional Neural Network (CNN), MRI images.

1. Introduction

Brain malignancies and intracranial tumors are new organisms that are developing in the cerebral cavity. They may originate directly from an organ, such as the brain, meninges, or nerves, or they may spread from other bodily parts into the brain by metastasis. The majority of these tumors can result in localized symptoms such headaches and intracranial pressure [1]. Brain tumors vary greatly in size, with some being detected early due to noticeable symptoms, while others grow undetected until they become quite large. Brain tumors affect roughly 1.9 to 5.4 people per 100,000 people annually, accounting for 1% to 3% of all body tumor types. In order to run simulations and forecast results, recent developments in artificial intelligence (AI) offer the chance to integrate and synthesis ever-increasing volumes of multidimensional data. This will improve shared decision-making for patients and physicians.

As the three pillars of tumor treatment, surgery, radiotherapy, and chemotherapy play an important role. However, when encountering tumors located in hollow organs, particularly in the gastrointestinal system, the diagnosis is often challenging due to the thin intestinal wall and the presence of gas, digestive juice, and food residues that may impede accurate imaging. In addition, for small lesions and some benign tumors with the same density as normal tissue, due to the partial volume effect, it is easy to miss the diagnosis. In summary, manual errors in early diagnosis methods may occur. The most recent artificial intelligence technology must be used to improve tumor diagnosis and prediction in order to increase the recognition and accuracy of brain tumors [2].

Convolutional Neural Network (CNN) technology is among the most cutting-edge forms of artificial intelligence and has a significant impact on a variety of industries, including image recognition, speech recognition, natural language processing, and others [3-7]. CNN can automatically learn abstract and high-level features in brain tumor images without manual feature selection and extraction. This can improve diagnostic accuracy and reduce the need for manual intervention. Additionally, CNNs were able to identify and classify images of brain tumors with a high degree of accuracy. Compared with traditional image processing methods, CNN can better distinguish different types of tumors and can detect tiny abnormal areas, thus improving the accuracy of diagnosis. More importantly, CNN is highly reproducible and can produce similar results across different experimental conditions and different datasets. This ensures the reliability and consistency of diagnosis, thereby improving the reliability of clinical application, helping medical researchers better understand the morphology and characteristics of brain tumors, thereby providing more clues and ideas for discovering new treatments and drugs. Therefore, this paper presents a CNN-based approach for automated classification and prediction of brain tumors.

This study aims to investigate the use of a deep learning algorithm built on a four-class convolutional neural network for brain tumor identification. This study used a dataset of MRI images containing different types of brain tumors for experiments. The dataset is separated into four categories after preprocessing and cleaning. Then, a trained four-class model was utilized to classify and diagnose MRI images. Finally, this study evaluated and validated the models, and analyzed and compared the diagnostic results.

2. Method

In this project, a series of brain tumor's MRI picture dataset provided by a dataset on Kaggle was obtained [8]. The dataset was pre-processed and organized into separate Training and Testing folders, with each folder containing four subfolders that correspond to different tumor classes. These folders have MRIs of respective tumor classes, which respectively concludes 2, 870 and 394 images. The images were grayscale and measured 512×512 pixels in size. Figure 1 depicts an exemplar image in its original form.

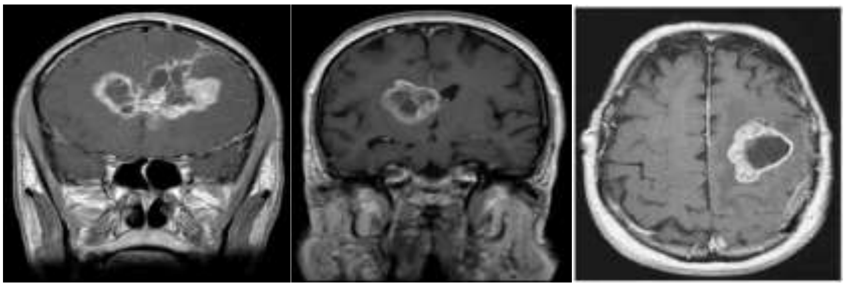


Figure1. The example images of brain tumor.

The data preprocessing step comprises three essential components. First, train_datagen object is created with the following parameter: rescale=1./255. This normalizes the pixel values of the images by dividing each pixel value by 255. Second, the test_datagen object is also created with the same parameter to ensure consistency in data preprocessing between the training and validation datasets. By applying multiple picture transformations, the ImageDataGenerator function also does data augmentation on the training dataset. By doing so, overfitting is decreased and the training dataset's diversity is increased. Third, the set of the data was resized into 256×256 and the batch size is 32, the class mode is categorical.

Several layers make up the CNN model, each of which is intended to extract and learn pertinent characteristics from the input images. A convolutional layer is created by combining several filters, and it pulls features from the input image. Each filter reacts to a certain characteristic or pattern in the image. The output of the convolutional layer is a set of feature maps indicating the presence of distinct features in the input image. Pooling Layer reduces the spatial dimensions of the feature maps through downsampling. The most common sort of pooling layer is the MaxPooling layer, which selects the greatest value within a window of pixels and ignores the remainder. The activation layer gives the model additional non-linearity by applying a non-linear activation function to the output of the layer before it. ReLU and sigmoid are two often used activation functions [9]. The fully connected layer is similar to the layers in a traditional neural network and is used to classify the entrance image according to the characteristics learned. One or more fully connected layers receive the flattened and inserted exit of the convolutional and pooling layers. The probability distribution for the various classes is generated via a softmax activation feature in the final fully connected layer. In order to prevent overflow, drop-out layers randomly remove a portion of the neurons from the preceding layer during training.

Four Conv2D layers with progressively larger filter widths of 32, 64, 128 and 256 make up his pattern architecture. The spatial dimensions of the characteristic maps are decreased by adding a MaxPooling2D layer with a pool size of (2, 2) after each Conv2D layer. The Flatten layer reduces the characteristic maps to a one-dimensional vector, which is then transferred to two thick layers with 128 units each that are entirely interconnected. Meningioma, glioma, pituitary tumor, and no tumor are the four tumor classifications for which probability distributions are produced by the first dense layer using ReLU activation and the second dense layer using a softmax activation function [10]. After the first Dense layer, the Dropout layer with a rate of 0.5 is introduced to prevent overflowing by randomly removing 50% of the neurons during training.

In this code, the GPU is commented out, which means that the model is trained on the GPU. Training a deep learning model on a GPU can be several times faster than training on a CPU, especially for large datasets and complex models. The optimizer used in this code is Adam. The CNN model's loss function, which calculates the difference between the predicted and actual probability distributions, and loss, which calculates the proportion of correctly categorized images, both use categorical cross-entropy.

3. Results and discussion

Using a collection of brain MRI pictures divided into four classes—meningioma, glioma, pituitary tumor, and no tumor—the CNN model was trained. The model's accuracy on the validation set, which is displayed in Table 1, was 85.4% after being trained for 10 epochs with a batch size of 32.

Table 1. The performance processed on the brain tumor dataset.

Model	Scale
Training Loss	0.4726
Training Accuracy	0.7321
Testing Loss	0.6231
Testing Accuracy	0.8543

The presented model has achieved a noteworthy level of accuracy on the validation set, indicating its efficacy in accurately classifying brain MRI images into the four distinct tumor categories. However, there are some limitations and potential areas for improvement. Firstly, the dataset used in this code only contains a limited number of images (3064 training images and 535 validation images). A larger dataset with more diverse images could potentially improve the model's performance.

Secondly, the model architecture employed in this code is comparatively simple, involving only four convolutional layers and one fully connected layer. More complex architectures, such as those using residual connections or attention mechanisms, could potentially improve the model's performance. Lastly, it is important to note that the accuracy metric alone may not be sufficient to evaluate the model's performance, especially in medical image analysis tasks where false negatives and false positives can have serious consequences. Other metrics such as sensitivity, specificity, and positive predictive value should also be taken into account.

4. Conclusion

In conclusion, this study explored the application of a four-class convolutional neural network in the diagnosis of brain tumors using MRI images. The dataset was pre-processed and organized into separate training and testing folders, with each folder containing four subfolders that correspond to different tumor classes. The trained model correctly classified the four different tumor types in the validation set with an accuracy of 85.4%, demonstrating its effectiveness in identifying brain MRI images. However, there are potential areas for improvement, such as using a larger and more diverse dataset and employing more complex model architectures. Additionally, it is important to consider metrics beyond accuracy, such as sensitivity and specificity, in evaluating the model's performance in medical image analysis tasks. Overall, the use of CNN-based approaches in the diagnosis and prediction of brain tumors has great potential to improve diagnostic accuracy, reduce the need for manual intervention, and provide more clues for discovering new treatments and drugs.

References

[1] Roumen K et al. 2020 Advances in 3D Image and Graphics Representation Analysis Computing and Information Technology (Springer Science and Business Media LLC)

[2] Rui M et al. 2021 Early Gesture Recognition with Reliable Accuracy Based on High-Resolution IoT Radar Sensors (IEEE Internet of Things Journal)

[3] Hou Z et al. 2022 Chex: channel exploration for CNN model compression Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 12287-12298.

[4] Rhanoui M et al. 2019 A CNN-BiLSTM model for document-level sentiment analysis Machine Learning and Knowledge Extraction 1(3): 832-847

[5] Palanisamy K et al. 2020 Rethinking CNN models for audio classification arXiv preprint arXiv:2007.11154

[6] Lin T Y et al. 2015 Bilinear CNN models for fine-grained visual recognition Proceedings of the IEEE international conference on computer vision 1449-1457

[7] Yu Q et al. 2020 Improved denoising autoencoder for maritime image denoising and semantic segmentation of USV China Communications 17(3): 46-57

[8] Sartaj B et al. 2022 Brain Tumor Classification (MRI) (CC0: Public Domain)

[9] Limeng P et al. 2019 DeepDrug3D: Classification of ligand-binding pockets in proteins with a convolutional neural network (PLOS Computational Biology)

[10] Eman S et al. 2019 Automated Grading for Handwritten Answer Sheets using Convolutional Neural Networks (International Conference on new Trends in Computing Sciences)

Proceedings of the 5th International Conference on Computing and Data Science
DOI: 10.54254/2755-2721/15/20230846

The stock price forecast under the failure of silicon valley bank based on the ARIMA model

Dengyi Gu

The department of Statistics, The University of California, Santa Barbara, 93117, The United State

dengyi@ucsb.edu

Abstract. The collapse of Silicon Valley Bank on March 10, 2023, had a profound impact on the stock prices of many companies in the United States. This study aims to examine the response of other banks in the US to this event by utilizing the Autoregressive Integrated Moving Average (ARIMA) model to forecast their stock prices. The research demonstrates that the ARIMA model effectively predicts the general trend of these banks' stock prices, with Root Mean Squared Error (RMSE) values below 1 for four out of six major US banks. These findings indicate that the proposed method is a promising tool for managing sudden fluctuations in stock prices, outperforming traditional linear regression models. Consequently, this research provides valuable insights for investors and financial institutions in managing and mitigating risks associated with abrupt market changes. Additionally, the study contributes to a greater understanding of the effects of bank collapses on the stock market. Overall, the research highlights the significance of incorporating advanced forecasting methods, such as ARIMA, in analyzing and predicting stock price movements in volatile market conditions.

Keywords: machine learning, ARIMA, stock price forecast.

1. Introduction

Silicon Valley Bank's long-term Treasury bond portfolio suffered losses as the Federal Reserve raised interest rates to combat inflation. Customers withdrew large amounts of their funds, prompting the bank to sell $21 billion in securities, borrow $15 billion, and sell Treasury stock. However, due to warnings from prominent investors, customers withdrew $42 billion the next day, leading to a bank run. On March 10, 2023, the 16th largest bank in the U.S., Silicon Valley Bank, collapsed, making it the second-largest bank failure in U.S. history [1-3]. As a result of the Silicon Valley Bank crisis, the financial sector was hit hard. Large technology stocks, such as Nvidia, Apple, Microsoft, and Amazon shares have suffered varying degrees of decline. In addition, the stocks of the six major U.S. banks also suffered significant declines. Goldman Sachs Bank shares even fell 4.22% [2]. In this case, the price volatility of the stock is remarkebly high. If market participants can accurately predict the direction of stock prices, this will allow them to consistently earn higher risk-adjusted returns than the market [4]. Therefore, it is crucial to make accurate predictions for such highly volatile stocks.

The term "machine learning" was first introduced by Arthur Samuel, a prominent figure in the field of computer games and artificial intelligence, who was an employee of IBM. [5, 6]. It is a field within computer science and artificial intelligence that aims to imitate human learning processes and progressively improve accuracy through the use of data and algorithms [7]. Machine learning encompasses a variety of disciplines, including statistics, probability theory, convex analysis, computational complexity theory and approximation theory. Machine learning algorithms are designed to automatically analyze data, detect patterns, and make predictions about previously unknown data. It has found widespread applications in diverse areas, such as data mining, natural language processing, and stock price prediction [8]. Many different machine learning algorithms have been proposed and utilized in the field of stock price prediction, such as Linear Regression Model (LR), Random Forest Model (RF), and Neural Network Model (NN) [4]. For instance, Kim et al. (2020) conducted a study to evaluate the effectiveness of incorporating effective transfer entropy (ETE) with popular machine learning techniques like LR, Multilayer Perceptron (MLP), RF, Extreme Gradient Boosting (XGBoost), and Long Short-Term Memory (LSTM) for predicting the direction of the S&P 500 index. [4]. The majority of previous studies have relied on statistical time series methods based on historical data to predict stock prices and returns. Some of the widely used techniques in stock price prediction include the Autoregressive Conditional Heteroskedasticity (ARCH) model, Autoregressive Integrated Moving Average (ARIMA) models, Moving Average (MA) models, Autoregressive Moving Average (ARMA) models, Kalman filter, and Exponential Smoothing methods [4]. However, there are no models in previous studies that predict the stock prices of other banks for the sudden event of Silicon Valley Bank's collapse.

To address this research gap, the present study employs models to predict the stock prices of various banks during the Silicon Valley Bank collapse and conducts a thorough analysis of the experimental results. To more accurately predict sudden stock price fluctuations in a short period, this paper will use the ARIMA model to predict the stock prices of other banks. In summary, the findings and recommendations provided in this paper can be useful for investors and financial analysts in making informed decisions and managing risks. Additionally, this paper also provides further recommendations based on the research findings to better prepare for any future occurrences of similar situations.

2. Method

2.1. Dataset description and preprocessing

This study used the closing prices of the six major US banks collected from the historical data section of Yahoo Finance, as a proxy for all U.S. banks to conduct the experiment. These were Citigroup Inc., Wells Fargo & Company, JPMorgan Chase & Co., Morgan Stanley, The Goldman Sachs Group, Inc., and Bank of America Corporation, all of which play significant roles in the US financial market [9-14]. The closing price data for each bank has been recorded from February 21, 2023 through April 3, 2023, including March 10, 2023, when Silicon Valley Bank went bankrupt. Notably, the data collection process excluded weekends, when the stock market is closed weekly. Therefore, there are exactly 30 days of data available. The valid data were recorded in an Excel table and stored in CSV format, with "Date" and "Close" representing the date and closing price of the day, respectively. To ensure accuracy and consistency, six different CSV files were created for the six banks. The recorded data served as the basis for subsequent analysis and modeling aimed at predicting sudden stock price fluctuations.

2.2. Proposed approach

This study used the ARIMA model to predict the stock prices of other banks in the United States during the special period of the Silicon Valley Bank's collapse. The ARIMA model, an acronym for Autoregressive Integrated Moving Average Model, is a time series-based prediction model utilized for analyzing and forecasting data [15]. The ARIMA model assumes that future time series values are composed of past time series values and random error terms, and predicts future time series values based on this historical data. The ARIMA model is comprised of three main components: autoregression (AR), differencing (I), and moving average (MA). $AR(p)$ is a regression model that represents the linear relationship between current values and past values, where p is the number of lagged observations in the model. $I(d)$ represents the differencing process applied 'd' times to the original time series data in order to achieve stationarity. $MA(q)$ is used to capture the effect of random errors on the prediction results, where "q" refers to the linear combination of the current residual value and the residual values of the past "q" time points in the ARIMA model. Combining all three types of models yields the $ARIMA(p, d, q)$ model, where "p", "d", and "q" are the three parameters of the ARIMA model, which can be combined to construct different ARIMA models. Therefore, the best ARIMA model was obtained by selecting the parameters.

The data was analyzed using Python's pmdarima library, specifically the auto_arima function, which automatically selects the optimal ARIMA model based on criteria like the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC) [16]. The model was then trained and tested using the previous data, with 90% of the data used as training data and the other 10% as testing data. The Root Mean Square Error (RMSE) method was used to calculate and help determine the magnitude of the error between the predicted and true values in the test data. A linear regression model, which is also suitable for short-term prediction but weak for non-linear data, was also used as a control model, and the root mean square error method was used to calculate the error and compare it with this model. The ARIMA model is also used to forecast the short-term stock price for the next 5 days in order to provide some references and help for future investment.

3. Result and discussion

After training and testing the data using the ARIMA model, the results were obtained for the stock price forecasts of the six largest U.S. banks. Among them, Bank of America, Goldman Sachs and Morgan Stanley chose to use ARIMA(0,1,0) after selecting the parameters of the model using the auto_arima function. The other three banks have different parameters, but all of them choose "1" for parameter "d". Figures 1 through 6 show the model's predictions for the stock prices of the six largest U.S. banks, respectively. The orange line is the test price, while the green line is the model's predicted price, and the error is calculated using the RMSE method. The red line shows the subsequent 5 days' forecasts based on these data. As can be seen from the graph, the ARIMA model can predict the general direction of the subsequent stock prices very well. The RMSEs of Citibank, Wells Fargo, JP Morgan, and Bank of America are all below 1. For comparison, the RMSE of the linear regression model for Citibank's stock price prediction is above 1.5, as shown in Figure 7.

The experimental results reveal that the ARIMA model is effective in predicting the volatile stock prices that exhibit non-stationary behavior, and it can provide a rough trend direction. Conversely, linear regression models are unsuitable for predicting stock prices during periods of abrupt fluctuations, such as the instance of Silicon Valley Bank's collapse. This is because such times are typically characterized by non-linear patterns in stock prices, which are not accommodated by the linear regression's assumption of a linear relationship between data points. The ARIMA model, in contrast, can capture the inherent non-linearity in the data, rendering it a more appropriate method for predicting sudden and significant changes in stock prices. Hence, the research suggests that the ARIMA model outperforms linear regression models in these circumstances.

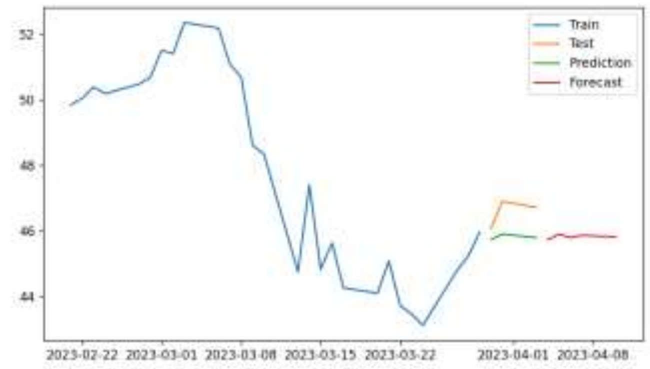


Figure 1. Figure with The Citigroup's Stock Price Forecast Chart in the ARIMA model.

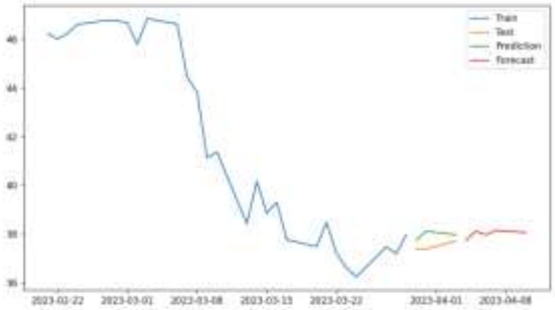


Figure 2. Figure with The Wells Fargo & Company’s Stock Price Forecast Chart in the ARIMA model.

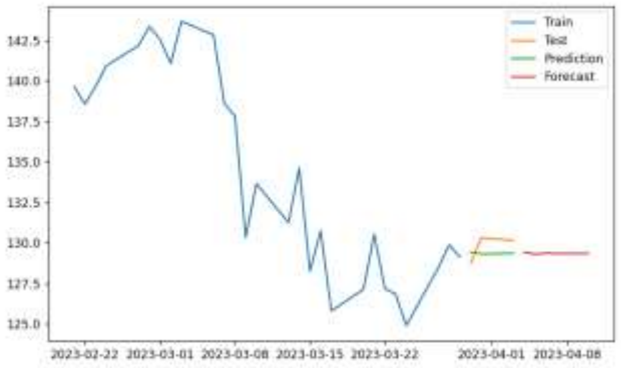


Figure 3. Figure with The JPMorgan Chase's Stock Price Forecast Chart in the ARIMA model.

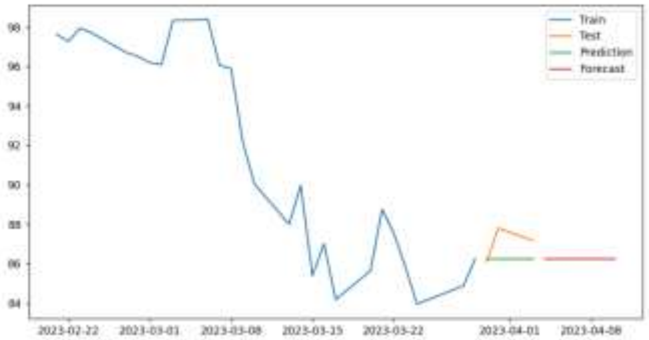


Figure 4. Figure with The Morgan Stanley's Stock Price Forecast Chart in the ARIMA model.

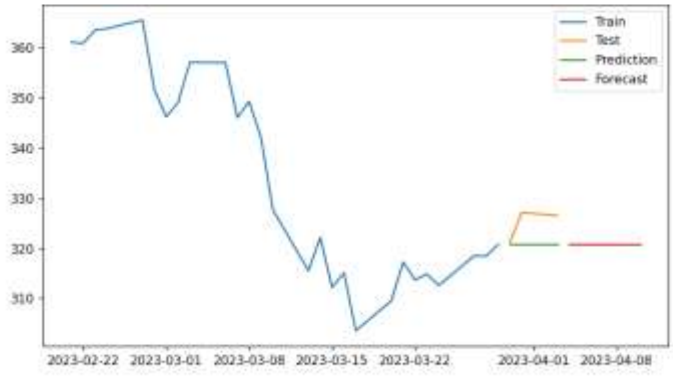


Figure 5. Figure with The Goldman Sachs Group's Stock Price Forecast Chart in the ARIMA model.

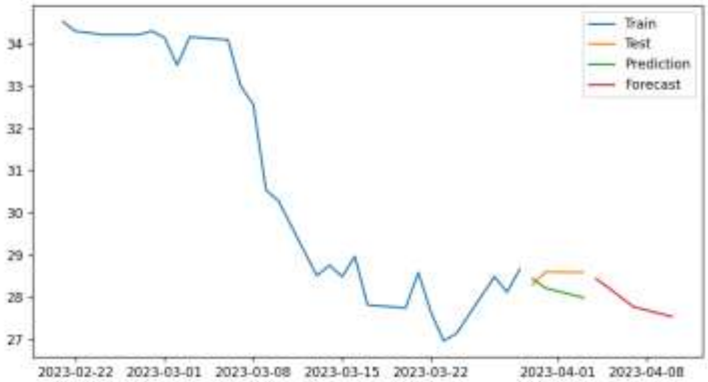


Figure 6. Figure with The Bank of America Corporation's Stock Price Forecast Chart in the ARIMA model.

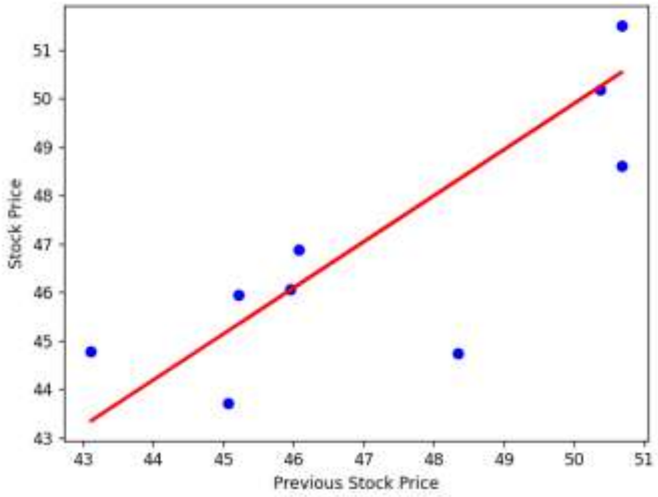


Figure 7. Figure with The Citigroup's Stock Price Forecast Chart in the Linear regression model.

4. Conclusion

This paper presents a methodology for predicting the stock prices of banks in the United States in the event of a potential failure of a bank in Silicon Valley. The study utilizes an ARIMA model that has been trained and tested using historical stock price data to make predictions. Additionally, a linear regression model is used as a comparison to demonstrate the advantages of the ARIMA model in this specific scenario. The experimental results indicate that the ARIMA model performs well in predicting the general direction of stock prices with various parameter settings. In the future, further adjustments will be made to the proposed method to reduce model errors and expand its applicability to other cases of sudden stock price changes beyond just the Silicon Valley Bank failure. Overall, this research provides insights into the potential of ARIMA models for stock price prediction in the banking sector and highlights areas for future improvement and expansion of the methodology.

References

[1] The guardian 2023 Why did Silicon Valley Bank fail? https://www.theguardian.com/us-news/2023/mar/10/silicon-valley-bank-collapse-explainer

[2] Finance Sina 2023 https://finance.sina.com.cn/wm/2023-03-11/doc-imyknara9466032.shtml

[3] Finance Sina 2023 https://finance.yahoo.com/news/silicon-valley-bank-committed-one-135059554.html

[4] Kumbure M M Lohrmann C Luukka P et al. 2022 Machine learning techniques and data for stock market forecasting: A literature review Expert Systems with Applications 116659

[5] Kohavi R and Provost F 1998 Glossary of terms Machine Learning3 vol. 30 no. 2–3 pp. 271–274

[6] Samuel A L 2000 Some studies in machine learning using the game of checkers IBM Journal of research and development 44(1.2): 206-226

[7] Github 2018 https://kangcai.github.io/2018/10/24/ml-overall-1/

[8] El Naqa I Murphy M J 2015 What is machine learning? Springer International Publishing

[9] Finance Yahoo 2023 ohttps://finance.yahoo.com/quote/C/history?p=C

[10] Finance Yahoo 2023 https://finance.yahoo.com/quote/WFC/history?p=WFC

[11] Finance Yahoo 2023 https://finance.yahoo.com/quote/JPM/history?p=JPM

[12] Finance Yahoo 2023 https://finance.yahoo.com/quote/MS/history?p=MS

[13] Finance Yahoo 2023 https://finance.yahoo.com/quote/GS/history?p=GS

[14] Finance Yahoo 2023 https://finance.yahoo.com/quote/BAC/history?p=BAC

[15] Capitalone 2021 https://www.capitalone.com/tech/machine-learning/understanding-arima-models/

[16] Pmdarima 2.0.3 2023 https://pypi.org/project/pmdarima/

Cross-data recognition on sign language letters based on convolutional neural networks

Zhixi Zhu

The Affiliated High School of South China Normal University, Guangzhou, China

zhuzx.cicely2021@gdhfi.com

Abstract. With an estimated 1.5 billion hearing-impaired people globally, sign language is a vital means of communication among them. Meanwhile, the complexity and diversity of sign languages across different regions bring challenges to researchers. As existing studies in this field usually lack the implementation of recognition models with multiple datasets, which limits their practical value in real-life scenarios, this study intends to get around this constraint. This work constructs a baseline American Sign Language Letter Recognition model using Convolutional Neural Network (CNN) and then optimizes it to enhance its ability. Finally, cross-data recognition is carried out. In order to train and evaluate the model, this study gathers data from several sources, including the MNIST sign language set and real-life photos. It also investigates how data augmentation affects recognition ability. Consequently, the CNN model is able to recognize hand gestures for different alphabetic letters in solid backgrounds. Its accurate rate of it reaches about 99.83%. For the extra real-life dataset, which contains 96 images, the model could still detect the majority of the images with sign language equivalents, despite some accuracy loss in more crowded backgrounds. This work, in general, concentrates on the potential of CNNs for sign language identification and highlights the significance of cross-data recognition in creating useful recognition models.

Keywords: computer vision, convolutional neural networks, sign language recognition.

1. Introduction

Sign language is a form of communication that utilizes hand gestures, having practically the same properties and functions just as spoken languages. It is widely used within the hearing-impaired community. Globally, there are currently more than 1.5 billion people suffering from hearing loss, which is predicted to reach 2.5 billion in 2050 [1]. Among all those individuals, 430 million are confronted with hearing disabilities, according to the World Health Organization [1]. Given these statistics, it is plausible to assert that there is presently a substantial population of hearing-impaired persons, who are likely to encounter a multitude of communication barriers in their daily lives since individuals who hear typically do not acquire the ability to comprehend sign languages. To overcome the obstacles in communication between deaf and hearing people, sign language recognition, with the help of technologies, has become a research focus among researchers. Due to an increasing need for convenience in real-time communication, explorations into this field can be of high practical value on various occasions, especially for sign language users to convey their messages more efficiently.

Studies in this field are important for bridging the communication gap by helping hearing people understand sign language, whereas challenging due to sign languages' complexity and diversity [2]. Similar to spoken languages, sign languages differ completely across countries, and even among distinct regions within the same nation. Moreover, many sign language categories, like Chinese sign language, involve dynamic gesture expressions, which make recognition tasks harder. As a result, to develop practical and accurate methods to recognize and translate sign languages, researchers need not only to collect a large amount of data but also to design and keep optimizing algorithms.

In terms of different methods used, there are currently two main categories of sign language recognition: data-glove based, and computer vision based. As the latter better simulates daily scenarios in which people are not wearing special gloves for communication, a computer vision-based data source was chosen for this study. Computer vision-based recognition is based on video or image data collected by input devices such as cameras. Then the recognition model is constructed mainly using deep learning, which has experienced a surge in recent years due to its prominent capabilities in handling complicated data from different sources [3]. Convolutional Neural Networks (CNNs) were deemed suitable for constructing a sign language recognition model due to their robust feature extraction capabilities. Other algorithms, like 3-Dimensional Convolutional Neural Networks, Recurrent Neural Networks, and YOLO are also introduced later by many researchers to carry out recognition tasks [4]. However, it is worth noting that sign language recognition has not received as much attention as other deep learning subjects. Moreover, existing studies usually lack the implementation of recognition models with multiple datasets, which means researchers often focus on the accuracy of their models in predicting the same dataset; the pictures in that dataset are likely to be taken under rather ideal circumstances: a neat background and high resolution. To enable practical sign language recognition, it may eventually be necessary to use data from other sources and test the model in a more realistic setting, while the main limitation of existing studies is their failure to do cross-data recognition.

To tackle the problem mentioned above, this study uses Convolutional Neural Network to construct a baseline American Sign Language Letter Recognition model and then carries out optimization to enhance its ability to carry out cross-data recognition. Data from different sources are collected and used to train and test the model. This study also explores the effects of data augmentation in the optimization process. Consequently, the improved CNN model can recognize the hand gestures for different alphabetic under ideal conditions like solid background colors. In daily contexts, when the backgrounds are much more cluttered, the model loses some accuracy when tested but can still correctly recognize most of the pictures given, which contain sign language representations.

2. Method

2.1. Dataset preparation

In this section, the composition of datasets and the pre-processing methods will be introduced.

The training dataset and the major testing dataset are obtained from Kaggle [5], including 21,455 cases used for training and 7,172 cases used for testing, representing 24 classes of alphabetic language letter (J and Z are excluded because of dynamic expressions). All the data is stored in CSV files, marked with labels. Each case is a 28*28-pixel grayscale image. Figure 1 presents the first 5 training cases.

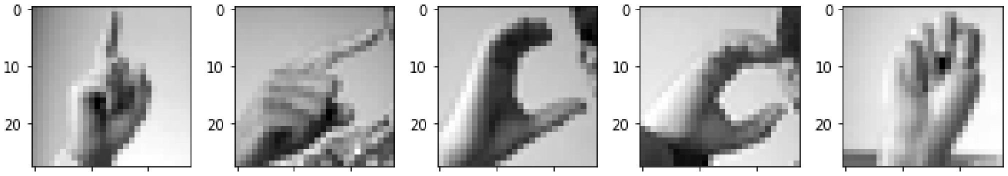


Figure 1. Sample training data.

It is worth mentioning that, in this study, both the original testing dataset corresponding to this training set and an extra testing dataset are utilized to evaluate the ability of this CNN model. The extra testing data set is composed of 96 images, 4 for each alphabetic letter, which are collected in real-life situations. This is because the implementation of sign language recognition tool in daily life is rather likely to be accompanied with cluttered backgrounds instead of ideal solid colors. By doing this, this study further examines the proposed CNN model's performance in practice.

To prepare for training and testing, all data, especially the real-life images, is firstly normalized: resized and converted to 28*28-pixel grayscale images, which are consistent with the format of the provided training dataset on Kaggle. Then the training data is augmented by rotating, zooming, and shifting a little bit, to increase the amount of data and avoid overfitting, hence better training the model and improving its capability to do recognition.

2.2. The proposed CNN model

A Convolutional Neural Network (CNN) is a type of neural network that takes advantage of convolutional structures to do feature extraction [6]. CNNs were considered suitable for constructing image recognition models due to their robust feature extraction capabilities. Many computer-vision-based recognition tasks, like face recognition, have been achieved with CNNs [6]. Typically, besides input and output, a CNN model is composed of three major types of layers: convolutional layers, pooling layers, and fully connected layers. In a convolutional layer, a filter, which is spatially smaller than the original image, shifts across the image and carries out operations, creating feature maps. A pooling layer helps reduce spatial information, hence rendering the model less complex and reducing the risk of overfitting. Finally, a fully connected layer builds links between previous layers and the output.

In this work, a CNN model with three convolutional layers is built after accepting grayscale images with a resolution of 28*28 pixels, and the filter size for each layer is set to be 3*3. A max-pooling layer of size 2*2 follows each convolutional layer. After accepting the data from the previous convolutional layer, it reduces the dimensions of feature maps. To avoid overfitting, a dropout layer with a rate of 0.2 is placed next to the pooling layer. The multidimensional output from the preceding layers is then flattened and entered into the fully linked layer after these. In the end, the CNN model can recognize and output the corresponding alphabetic letter for each sign language gesture in the input data.

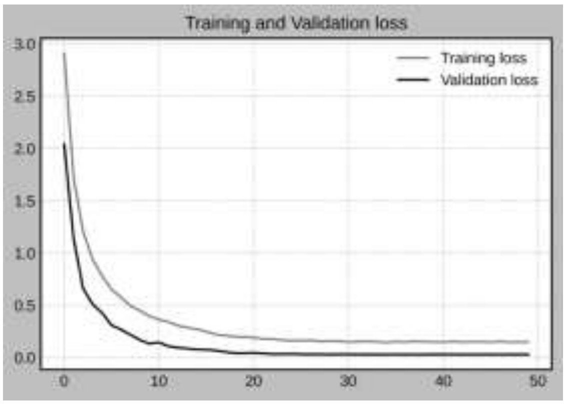
2.3. Implementation details

This part describes some relevant details in this study.

TensorFlow and Keras are used to build this recognition model since their convenience has been proved by many studies [7-9]. The model is trained across 50 epochs, with each batch with a size of 128. The patience parameter for learning rate reduction is set to 2, which implies that if the validation accuracy does not increase after two epochs, the learning rate will be automatically reduced. The Adam optimizer, which is a computationally efficient method for gradient-based optimization, is used in this model [10]. And the categorical cross-entropy is selected as the loss function. Furthermore, to optimize the model and enhance its performance on the test datasets, several changes are made to the model's parameters. All loss and accuracy data are recorded in the model's history during the training phase. Finally, the provided dataset and additionally collected photos are used to test the model.

3. Results and discussion

Data augmentation is done on the original training dataset to avoid overfitting and increase the amount of data. The Figure 2 and Figure 3 below show the training and validation loss and accuracy during model training. The loss diagram illustrates that there is a rapid decrease in training and validation loss at the beginning, then these figures both approaches to zero. And in the accuracy graph, both training and validation accuracy rise to approximately one at the end of training. The trends shown in these two graphs imply the parameter settings are appropriate.



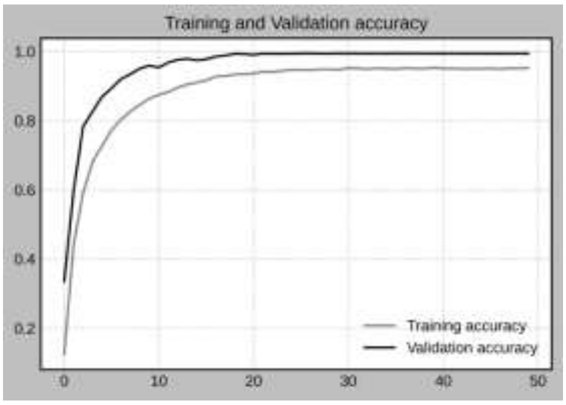


Figure 2. Loss diagram.

Figure 3. Accuracy diagram.

After testing the model using the provided datasets on Kaggle, the results reveal that the accurate rate of this CNN model in recognizing fingerspelling letters from the corresponding test cases on Kaggle reaches approximately 99.83%. From this, it can be speculated that the model's ability in recognizing fingerspelling gestures under ideal conditions is strong.

Then the model is tested with 96 real-life photographs, it recognizes 81 of them correctly, which means the precision rate of it drops to around 84.38%. This reveals that the model can still accurately recognize the majority of testing photographs, but the cluttered backgrounds negatively affects its performance, to some extent.

The Table 1 below compares its performance on different datasets.

Table 1. Results comparison.

	Provided testing dataset	Collected real-life photos
Recognition accuracy	99.83%	84.38%

The testing result suggests that the model is highly acute when performing recognition tasks under ideal conditions—tiny backgrounds and clearly presented gestures. Nevertheless, when faced with real-life photos with more complex backgrounds, the accuracy rate experiences an acceptable drop. This also points out that future studies may need to concentrate on improving the recognition methods' practical value.

Comparatively, this study discusses both the model's performance on the original dataset, which is rather high, and in real-life scenarios, which is relatively lower. It tackles the constraints in previous studies which did not consider the practical utility of the recognition model.

4. Conclusion

In this work, CNN is selected as the algorithm to implement American sign language recognition based on computer vision. Datasets are collected from online resources and real life, to overcome the limitations that many previous researchers did not take the practical utility of the model into account. The results show that the recognition capability of this model in solid color backgrounds is high, while it loses some accuracy when recognizing real-life photographs taken in more cluttered backgrounds. This suggests that the model needs further adjustments and optimizations to truly put into daily-life use. To improve this model, future researchers may need to consider designing better models by combining the baseline CNN algorithm with other methods like LSTM, or adopting other algorithms like 3D-CNN, and YOLO. Furthermore, as the situations of daily communication are likely to be much more complex than training data contained in existing sign language datasets, collecting a wider range of data in real-life conditions may hone the trained model's recognition capability in practice.

References

[1] World Health Organization 2023 Deafness and Hearing Loss

[2] Suharjito and Gunawan H and Thiracitta N and Nugroho A 2018 Indonesian Association for Pattern Recognition International Conference pp 1–5

[3] Voulodimos A and Doulamis N and Doulamis A and Protopapadakis E 2018 Computational Intelligence and Neuroscience pp 1-13

[4] Li T and Yan Y and Du W 2022 IEEE International Conference on Artificial Intelligence and Computer Applications pp 927-31

[5] Tecperson 2017 Sign Language MNIST (Kaggle)

[6] Li Z Liu F Yang W et al. 2021 A survey of convolutional neural networks: analysis, applications, and prospects IEEE transactions on neural networks and learning systems

[7] Yu Q Wang J Jin Z et al. 2022 Pose-guided matching based on deep learning for assessing quality of action on rehabilitation training Biomedical Signal Processing and Control 72: 103323

[8] Grattarola D Alippi C 2021 Graph neural networks in tensorflow and keras with spektral [application notes IEEE Computational Intelligence Magazine 16(1): 99-106

[9] Ramasubramanian K Singh A Ramasubramanian K et al. 2019 Deep learning using keras and tensorflow Machine Learning Using R: With Time Series and Industry-Based Use Cases in R, 667-688

[10] Kingma D P and Ba J 2015 the 3rd International Conference for Learning Representations (San Diego)