Proceedings of the 2023 International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/44/20230676

Comparative analysis of VGG, ResNet, and GoogLeNet architectures evaluating performance, computational efficiency, and convergence rates

Xiao Zhang[1,4], Ningning Han[2], Jiaming Zhang[3]

[1]Shanghai Jiaotong University, Shanghai, 200240, China
[2]East China Normal University, Shanghai, 200062, China
[3]Cambridge international school LITAI COLLEGE, Shanghai, 200080, China

[4]ZhangXiao97@alumni.sjtu.edu.cn

Abstract. This paper conducts an in-depth comparative analysis of three foundational machine learning architectures: VGG, ResNet, and GoogLeNet. The focus of the evaluation is their performance metrics on the CIFAR-100 dataset, a widely adopted benchmark in the field. Employing a comprehensive set of evaluation metrics, this investigation assesses not only testing accuracy but also the rate of training convergence and computational efficiency, providing a holistic perspective on the architectures' capabilities. Through rigorous experimentation, we elucidate the inherent advantages and drawbacks associated with each of these architectures. For instance, our findings delve into the nuances of how different architectures fare in terms of computational resources, which is vital for deployment in resource-constrained environments. Additionally, this study extends the analysis to explore the effect of hyperparameter settings, particularly learning rates, and the utility of data augmentation techniques in modulating the overall performance of each architecture. The ultimate objective is to furnish empirical insights that will assist researchers and practitioners in making well-informed choices when selecting a machine learning architecture for their specific application requirements.

Keywords: Machine Learning, Computer Vision, VGG Architecture, ResNet Architecture, GoogLeNet Architecture.

1. Introduction

Most of the sensory message in our daily life comes from visual information received by our eyes. Computer Vision (CV), as ‘eyes’ of computer, has developed rapidly because of fast growth of internet since the 21st century. It has a lot of applications in reality such as facial-recognition, picture and video recognition and editing, aided driving etc. Work efficiency can be improved largely with the assistance of CV, as it can transform pictures and videos to digital data which can be handle by computers. This can help firms to process massive amount of data quickly, reducing time and labour costs. Other advantages such low error rate or high precision can also be achieved because of the nature of computer. They will not get tired or distracted while they can observe those tiny changes and differences that human eye can not recognize. These leads to increasing economic and monetary value for relative industries and companies. However, the technological development is much more difficult

DOI: 10.54254/2755-2721/44/20230676

than expected. There are several tasks that CV needs to do, including classification, localization and segmentation. Unlike textual data, image data is much more complex with many dimensions of features. Therefore, dealing with this kind of data has more uncertainty. For example, data quality might affect the accuracy and reliability of the result. Even the small changes from light can affect the performance of the model. Rate of convergence also varies due to the different situations and the architecture chosen.

Computer vision plays an increasingly important role in today's world by giving computer systems the ability of "vision", enabling machines to perceive, understand and interpret image and video data. Traditional non-artificial intelligence (AI) methods and heuristics have shown a number of shortcomings when implementing computer vision tasks. These methods often face limitations in accuracy, generalisation ability and efficiency when dealing with complex scenes, significantly changing environments, and large-scale datasets. The emergence of AI models fills these shortcomings and brings new hope to computer vision tasks. In the field of target detection, breakthroughs have been achieved with the development of Deep Neural Networks (DNNs) and Convolutional Neural Networks (CNNs.) The introduction of DNNs and CNNs not only improves the accuracy of target detection, but also enables the models to extract a higher level of semantic information from the images, providing a solid foundation for the successful application of computer vision tasks. This thesis will focus on three important neural network models, GoogLeNet, ResNet and VGG16, which have a wide range of applications in computer vision. GoogLeNet is well known for its deep network structure and the innovation of the "Inception" module, ResNet effectively solves the problem of gradient vanishing in deep neural network training by introducing residual connections, and VGG16 excels in image classification tasks with its simple but deep convolutional layer structure. In this paper, we will introduce the architectures and working principles of these three models in detail and explore their performance in different computer vision tasks through the analysis of experimental results. By studying and analysing these models, we aim to gain an in-depth understanding of their strengths and limitations in computer vision tasks and provide valuable references for future research and applications. This study will contribute an important reference for the development and advancement of the computer vision field, helping to better cope with complex vision tasks and challenges.

2. Related work

In the realm of computer vision, propelled by the integration of deep learning techniques, substantial progress has been witnessed across various domains, heralding transformative strides in object detection, semantic and instance segmentation, Generative Adversarial Networks (GANs) for image synthesis, and real-world contextual implementations spanning healthcare, agriculture, manufacturing, and surveillance. Noteworthy contributions within this landscape include the pioneering work by Sanghyun Kim et al., which introduces the Multi-layered Relation Embedding Network (MUREN) for the detection of Human-Object Interactions (HOIs)[1]. This innovation encompasses a dual-pronged approach through the employment of a multiplex relation embedding module and an attentive fusion module, effectively engendering contextually enriched information exchange to amplify detection efficacy. Concurrently, Zhang et al., elucidate a comprehensive methodology for multi-modal medical image synthesis, aimed at enhancing medical image modality completion [2]. By synergistically amalgamating the Commonality- and Discrepancy-Sensitive Encoder, Dynamic Feature Unification Module, and generative adversarial framework, this research affirms its supremacy via demonstrable excellence on multi-modal MRI datasets. Moreover, through an intricate dissection of component effects, the study effectively underscores the role of each constituent element. In the domain of medical image analysis, the prospects of self-supervised deep learning (DL) emerge prominently. Nielsen et al., delineate the development of DINO, a paradigm that excels in medical image classification using minimal labeled data [3]. The pronounced proficiency exhibited across three distinct medical datasets accentuates the viability of DINO as an apt solution for scenarios characterized by paucity of annotated samples. Further diversifying the application spectrum, Jungo et al., introduce MahaAD, an innovative unsupervised framework catering to safer retinal microsurgery

DOI: 10.54254/2755-2721/44/20230676

[4]. By harnessing the Mahalanobis distance metric, this method identifies unsuitable images emanating from iiOCT probes, thus augmenting downstream task performance. Importantly, this work extends its ambit to encompass marker discovery, COVID-19 lung lesion segmentation, and anomalous detection in robotic surgeries, underscoring its versatile implications. In summation, these recent advancements underscore the continuous evolution of deep learning methodologies within computer vision, casting a ripple effect across myriad applications and domains, thereby charting an illuminating trajectory for future explorations.

The evolution and fusion of diverse models have ushered in a new era where emergent deep learning paradigms are finding application in an array of contexts spanning various domains. Agus Eko Minarno et al., investigated VGG-16 model's effectiveness in classifying glioma brain tumors from MRI images, achieving up to 100% accuracy, and suggests data preprocessing's variable impact on model performance. investigates VGG-16 model's effectiveness in classifying glioma brain tumors from MRI images, achieving up to 100% accuracy, and suggests data preprocessing's variable impact on model performance [5]. A Korean research introduced an enhanced object detection approach [6], merging VGG and ResNet networks to boost accuracy through high-level feature extraction, training loss reduction, and network modifications, resulting in a 85.8 average mean average precision for detecting objects of diverse sizes. Another work investigated GoogLeNet's application in food image classification, attaining notable accuracy across datasets, while highlighting challenges and proposing future enhancements [7]. Ahmet Çınar and Seda Arslan Tuncer suggested a combined Alexnet and Googlenet hybrid model with SVM for classifying various white blood cell types, outperforming standalone models, and attaining 99.73% accuracy and 0.99 F1-score across Kaggle and LISC databases [8]. Collectively, numerous studies showcase the diverse applications and advancements in computer vision models across varied domains, emphasizing accuracy improvements, innovative methodologies, and potential avenues for further research.

3. Methodology

3.1. VGG

The VGG (Visual Geometry Group) model, developed by the University of Oxford, has stood the test of time as a benchmark in simplicity and performance [9]. While a trend towards more intricate network architectures was prevalent, the VGG model brought forth the potency of stacking simple 3x3 convolutional layers. This design choice enabled the network to learn complex features while maintaining computational efficiency, demonstrating that high performance could be achieved without the need for architectural complexity. The VGG model, specifically the VGG16 variant, comprises an input layer, followed by five convolutional blocks, each consisting of multiple 3x3 convolutional layers with ReLU activation functions, and 2x2 max-pooling layers. Subsequently, the model employs two fully-connected layers with 4096 nodes each and culminates in an output layer consisting of 1000 nodes, corresponding to the 1000 categories in the ImageNet dataset. This layered approach allows for a hierarchical, increasingly abstract representation of input features, which contributes to its superior performance. One of the enduring aspects of the VGG architecture is its adaptability for various computer vision tasks. Its feature extraction capabilities make it an ideal candidate for transfer learning applications, extending its utility beyond mere image classification. The advantage of VGG16 is the simplicity, which means it is easy to understand and realize while the disadvantage is that it has relatively larger number of parameters, causing high storage and computational expense.

DOI: 10.54254/2755-2721/44/20230676

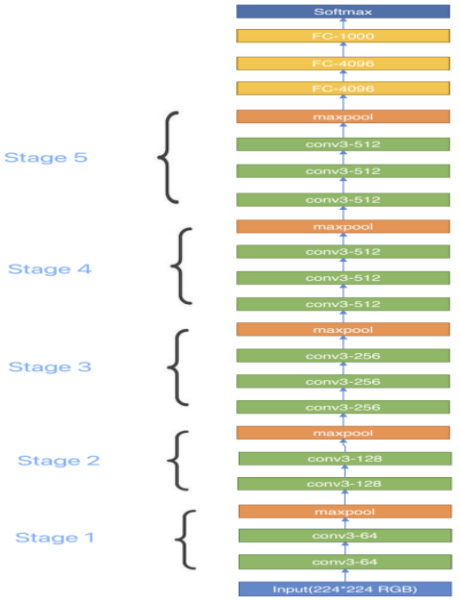

Figure 1. Structure of VGG.

3.2. ResNet

The ResNet network is a network structure based on the VGG19 network, with the introduction of residual units through short connection mechanism [10]. It solves the degradation problem of the decreasing of accuracy rate with the increasing network layers by identity mapping (which refers to the curved lines) and residual mapping (the remaining part of the model). ResNet follows the complete 3*3 convolutional layer design of VGG and uses a Batch Normalization layer and a ReLU activation function, in addition to introducing an extra 1*1 convolutional layer to transform the input into the desired shape, which is directly summed with the residual function result. As what we can see from the model, a very essential design principle of ResNet is that the amount of feature map doubles when the size of feature map is reduced by half. The principle helps maintaining the complexity of network layers. ResNet creates residual learning by adding the short connection mechanism between every two layers.The main differences between them are mainly in the fact that ResNet directly uses the stride=2 convolution for down sampling and replaces the fully connected layer with a global average pool layer. With the strategy it takes, the application of zero-padding increases the dimensions and also no more arguments will appear during the network. Or we could adopt new mapping, the projection shortcut which generally uses a 1x1 convolution, which increases the parameters and also increases the amount of computation.

DOI: 10.54254/2755-2721/44/20230676

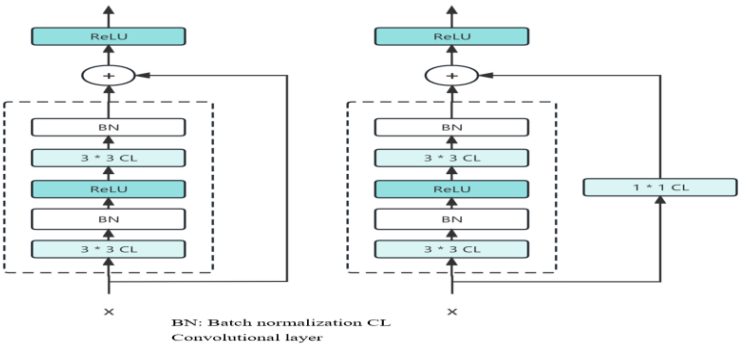

Figure 2. Key Structure of ResNet.

3.3. GoogLeNet

GoogleNet, also referred to as Inception v1, stands as a seminal model, chiefly due to its introduction of the Inception module[11]. This component enables the network to adaptively acquire spatial hierarchies of features. The initial layer of the GoogleNet architecture, termed the "Stem Layer," incorporates a conventional convolutional layer succeeded by pooling operations. This structure serves to compress the spatial dimensions of the input image while augmenting the depth dimension. At the crux of the GoogleNet architecture lies the Inception Module, which employs parallelized convolutional filters of varying dimensions (1*1, 3*3, and 5*5) alongside max-pooling operations. This design facilitates the network's ability to capture spatial feature hierarchies at disparate scales. To optimize computational efficiency, 1*1 convolutional operations are deployed for dimensionality reduction. In contrast to preceding architectures such as AlexNet and VGG, which incorporated fully connected layers, GoogleNet opts for an average pooling layer antecedent to the final softmax layer. This choice yields a substantial diminution in the total number of parameters. During the training phase, the architecture is further augmented with two auxiliary classifiers, appended to intermediate layers. These classifiers furnish auxiliary gradients, thereby enhancing the efficacy of backpropagation. It should be noted that these auxiliary classifiers are omitted during the inference phase for computational expediency.

DOI: 10.54254/2755-2721/44/20230676

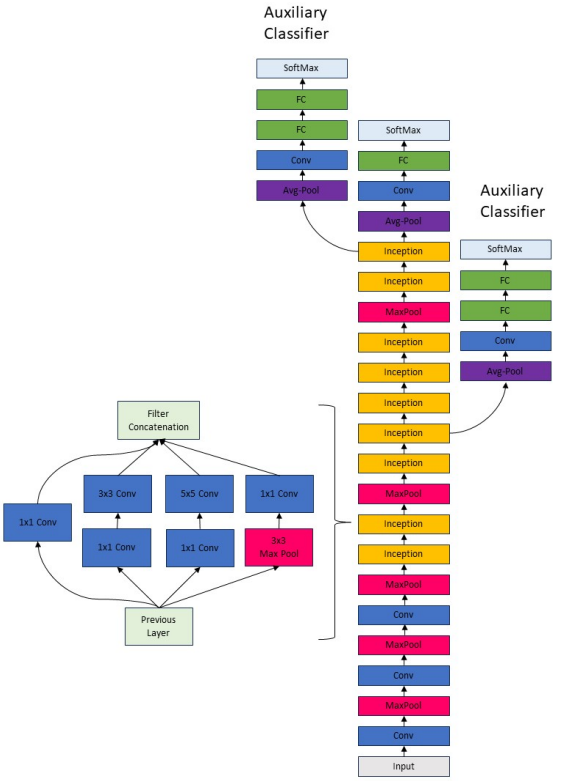

Figure 3. Structure of the GoogLeNet and the Inception Module.

4. Experiments

In our work, we apply three models on the Cifar-100 data set. The CIFAR-100 dataset, an extension of the simpler CIFAR-10 dataset, consists of 60,000 32*32 color images categorized into 100 classes, each containing 600 instances. These 100 classes are further grouped into 20 superclasses, and each image is annotated with both a fine-grained label indicating its class and a coarse-grained label indicating its superclass. The dataset is partitioned into a training set comprising 50,000 images and a test set comprising 10,000 images. Developed by Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton, this dataset serves as a standard benchmark for evaluating machine learning algorithms in the domain of image classification. Its manageable size and complexity make it a popular choice for rapid experimentation and proof-of-concept development, particularly in the application of convolutional neural networks (CNNs).

4.1. VGG-16 Performance

During the experiment, CIFAR-100 dataset was used to test VGG16 architecture. The figure 4 gives the accuracy and the loss information of the model. As it shows, the overall trend of the line is increasing, representing the accuracy is getting higher and higher. The first stage of 0-60 epochs have relatively larger oscillating behaviour, which has been improved in the later stages, with obvious higher accuracy in 61st epoch. The model becomes stable after the convergence in the epoch of 65 while it has further accuracy in 121st epoch with the better learning rate. The final accuracy is 0.7212.

DOI: 10.54254/2755-2721/44/20230676

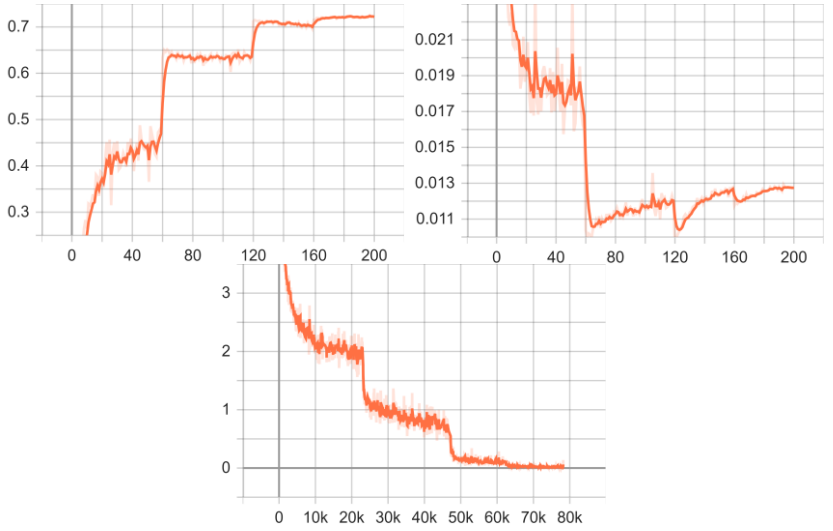

Figure 4. Test Accuracy (Left), Test Average Loss (Middle) and Train Loss of VGG-16 Model.

In the average loss of the test graph, similar to the accuracy graph, x-axis denotes the training epoch and the y-axis denotes the loss value. Significantly, the loss value in the training process decreases quickly between the step 57 and step 65, which might be the result of the refined learning rate. However, the loss value started to climb up again in the later process, which is possibly because of overfitting. The final loss value is 0.01202.

In the train loss graph, the x-axis represents the number of update steps while y-axis represents the loss value. Each time the learning rate refresh brings a remarkable loss reduction and oscillation is observed during the process. The final training loss is about 0.1375.

4.2. ResNet-101 Performance

In our experimentation with ResNet-101 on the Cifar-100 dataset, the first plot of interest is the accuracy plot. The x-axis of this plot represents the epoch of the training, while the y-axis denotes the corresponding accuracy. It is evident from the graph that the model's accuracy oscillates in small increments during the initial sixty epochs. This suggests that the features learned by the model in its early training stages are relatively stable, albeit not yet optimal. After the 61st epoch, there is a considerable increase in accuracy, reaching a value of 0.74, before reverting back to approximately 0.7 shortly thereafter.

The second plot we examine is the average loss plot. Coinciding with the fluctuation in accuracy after the 61st epoch, the average loss begins to increase again after initially showing a significant decline. Furthermore, after another 60 epochs, precisely at the 121st epoch, the average loss experiences a sharp decrease and ceases to vary drastically. This period also coincides with another notable increase in model accuracy.

The final plot represents training loss. Throughout the training process, the training loss ultimately stabilizes, serving as an indicator of the model reaching a steady state. This stabilization is largely attributable to the architecture's residual learning and skip connections, which effectively combat the degradation problem, thereby enhancing both the model's accuracy and efficiency. By the end of the training period, the model achieves a steady-state with an accuracy of 0.7907 and a loss value of 0.0072465.

DOI: 10.54254/2755-2721/44/20230676

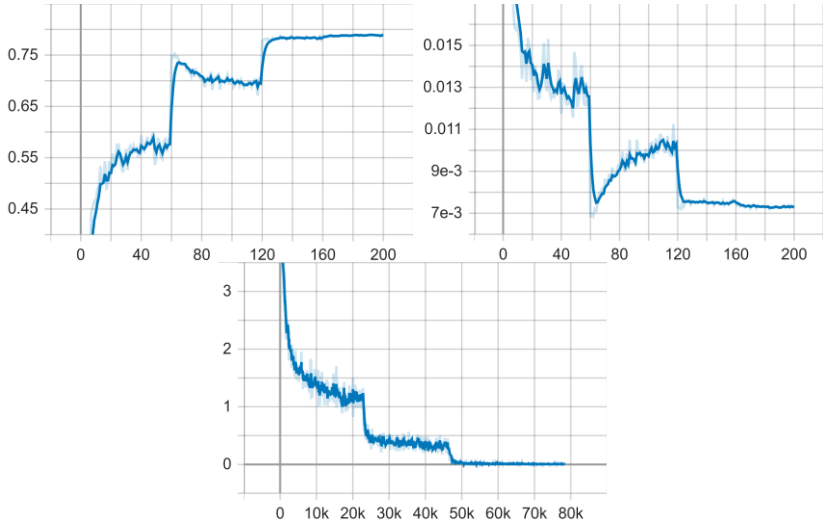

Figure 5. Test Accuracy (Left), Test Average Loss (Middle) and Train Loss of Resnet-101 Model.

4.3. GoogLeNet Performance

In the experimental evaluation, the GoogLeNet architecture was subjected to testing using the CIFAR-100 dataset. In the training loss curve graph, the x-axis represents the number of epochs for training, while the y-axis indicates the loss value. The curve, on the whole, exhibits a declining trend, decreasing from the initial loss value to near zero. This suggests that the model parameters eventually converge to a near-zero loss on the training dataset. Notably, every third of the total number of steps, the model's loss rate experiences a sharp decline followed by convergence. This behavior is attributed to our experimental design, wherein the learning rate is reduced at the 60th, 120th, and 160th epochs.

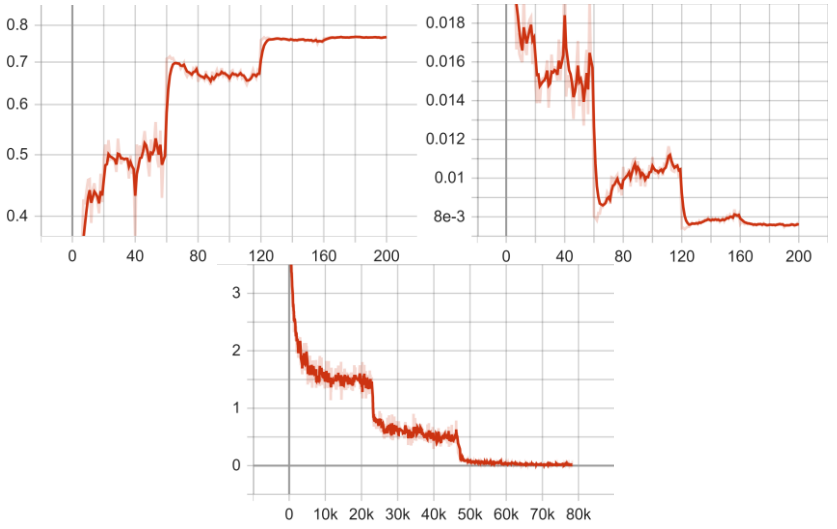

Figure 6. Test Accuracy (Left), Test Average Loss (Middle) and Train Loss of GoogLeNet Model.

In the graph depicting the average loss on the test set, the x-axis again represents the epochs, and the y-axis denotes the average loss value. The curve also generally displays a declining tendency, decreasing from the initial loss value to 0.007598. Due to changes in the learning rate, the model's loss rate on the test set also experiences three rapid declines. However, after the second decline, the loss value reaches near its optimal point but subsequently increases due to overfitting. It finally converges to the optimal value following the last decline.

DOI: 10.54254/2755-2721/44/20230676

In the graph illustrating the accuracy curve on the test set, the x-axis likewise represents the number of steps, while the y-axis signifies the prediction accuracy rate. The curve predominantly follows an ascending trend, increasing from near zero to the ultimate optimal rate of 76.63%. After the first ascent, the accuracy curve oscillates around 50%, which is caused by an excessively high learning rate. After reducing the learning rate, the model gradually converges to its optimal state.

5. Conclusion

In the realm of computer vision, distinct approaches have been materialized through seminal architectures such as VGG, ResNet, and GoogLeNet we focus on in our work. The VGG architecture, characterized by its uniform layer configuration, provides ease of implementation at the expense of computational intensity. ResNet innovatively employs skip connections to alleviate the vanishing gradient problem, thereby enabling the training of significantly deeper networks with enhanced computational efficiency. Conversely, GoogLeNet integrates Inception modules, optimized for the extraction of multi-scale features, and offers memory efficiency. These architectures commonly leverage pre-trained models to facilitate transfer learning, thereby mitigating data and temporal constraints. Additionally, data augmentation and regularization methods are ubiquitously employed to bolster generalization while curtailing overfitting. In our work, a comparative assessment was executed to evaluate the architectures' performance on the CIFAR-100 dataset. Our findings reveal that the VGG architecture yielded the quickest training convergence, completing the process within 1 hour and 31 minutes. GoogLeNet followed, requiring 2 hours and 38 minutes, while ResNet-101 necessitated a duration of 8 hours and 35 minutes (It is, however, very fast because when we did the extended experiments, we found that the inception v4 model consumed more than 20 hours to train which have only half of the layers of resnet-101). Variability in testing accuracy was observed: ResNet led with 78.93%, trailed by GoogLeNet at 76.63%, and VGG at 72.21%. Each architecture bears inherent advantages and is optimally suited for specific applications. VGG's structural simplicity renders it highly effective in fine-grained image classification and as a versatile feature extractor for various tasks, making it an ideal choice for problems necessitating intricate texture and shape recognition. ResNet's utilization of shortcut connections offers an elegant solution to the vanishing gradient problem and makes it a preferred choice in object detection paradigms such as Faster R-CNN, as well as real-time applications. The architecture also exhibits commendable generalization capabilities across diverse computer vision tasks. GoogLeNet, distinguished by its Inception modules, operates with fewer parameters while maintaining robust performance. This feature renders it particularly beneficial in resource-constrained environments and in tasks that demand multi-scale object recognition. Its inclusion of auxiliary classifiers during training further mitigates the vanishing gradient challenge.

References

[1] Kim S, Jung D, Cho M. Relational Context Learning for Human-Object Interaction Detection[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 2925-2934.

[2] Zhang Y, Peng C, Wang Q, et al. Unified Multi-Modal Image Synthesis for Missing Modality Imputation[J]. arXiv preprint arXiv:2304.05340, 2023.

[3] Nielsen M, Wenderoth L, Sentker T, et al. Self-supervision for medical image classification: state-of-the-art performance with~ 100 labeled training samples per class[J]. arXiv preprint arXiv:2304.05163, 2023.

[4] Jungo A, Doorenbos L, Da Col T, et al. Unsupervised out-of-distribution detection for safer robotically guided retinal microsurgery[J]. International journal of computer assisted radiology and surgery, 2023: 1-7.

[5] Minarno, A. E., Bagas, S. Y., Yuda, M., Hanung, N. A., & Ibrahim, Z. (2022). Convolutional Neural Network featuring VGG-16 Model for Glioma Classification. JOIV: International Journal on Informatics Visualization, 6(3), 660-666.

DOI: 10.54254/2755-2721/44/20230676

[6] Haque, M. F., Lim, H. Y., & Kang, D. S. (2019). Object detection based on VGG with ResNet network. In 2019 International Conference on Electronics, Information, and Communication (ICEIC) (pp. 1-3). IEEE.
[7] Singla, A., Yuan, L., & Ebrahimi, T. (2016). Food/non-food image classification and food categorization using pre-trained googlenet model. In Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management (pp. 3-11).
[8] Çınar, A., & Tuncer, S. A. (2021). Classification of lymphocytes, monocytes, eosinophils, and neutrophils on white blood cells using hybrid Alexnet-GoogleNet-SVM. SN Applied Sciences, 3, 1-11.
[9] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.
[10] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
[11] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.