Application and analysis of machine learning in handwritten digit recognition

Zichun Mo

School of Intelligent Systems Science and Engineering, JINAN University, GuangZhou, GuangDong, 510655, China

mzc2001@stu2019.jnu.edu.cn

Abstract. It appears from the information that Character recognition research is currently focused on handwritten digit recognition, a significant subfield of optical character recognition, i.e. the use of computers to recognise and process digital information. In today's increasingly mainstream computer and data era, handwritten numeric recognition can simplify the process of paper-based offices, reduce the intensity of work when analysing data statistics afterwards and improve work efficiency. There are many algorithms to achieve recognition, each with different recognition accuracy, implementation efficiency and application scope. Based on the basic concepts described above, this thesis investigates the efficiency and accuracy of three algorithms - template matching, SVM and deep learning - in recognising handwritten digits with different sample sizes. The models or kernel functions currently used to process data of varying complexity and restricted scenarios also require continuous improvement to ensure accuracy, making it all the more important to discuss them in detail.

Keywords: machine learning, digit recognition, CNN.

1. Introduction

With the development of computer technology, artificial intelligence has gradually begun to replace some of the traditional work processes, and the widespread use of artificial intelligence technology has greatly improved the efficiency and accuracy of work. In the era of big data, data has also become more massive and complicated, so people are relying on machines to help them deal with it in order to make reasonable decisions more scientifically, quickly and accurately. Handwritten digit recognition is an essential component of optical character recognition, a technology that uses a computer to recognise handwritten digits based on optical character recognition [1].

The challenge of handwritten digit recognition lies mainly in the fact that the composition of the digit consists of only ten characters, the digit strokes are small and the structure is simple, but this also leads to poor accuracy and low discriminative power of computer recognition. Handwritten digit recognition has long been valued by researchers around the world and has been studied extensively. Among them, template matching algorithms were the first to be studied and used, and then as the research progressed, support vector machine algorithms were proposed, and this research entered a new field, and currently neural network-based algorithms are widely used [2,3]. Based on the current research results, the steps of handwritten digit recognition can be divided into image pre-processing, feature extraction and classification. Since the captured digital images may have noise and other problems that make the

recognition effect unsatisfactory, preprocessing operations such as image enhancement and noise reduction are needed before recognition. Depending on the feature extraction method, different classifiers and different algorithms can be selected. In the following article, the principles of image preprocessing and classification are discussed.

2. Dataset Processing

Commonly used datasets for handwritten digits are the United States National Institute of Standards and Technology Handwriting Dataset (MNIST) [4]; the United States Postal Service Handwriting Dataset (USPS); and the University of California, Irvine Handwriting Dataset (UCI). The MNIST dataset has a large number of samples, with a test set of approximately 10,000 samples and a training set of approximately 60,000 samples, and the greyscale values of each handwritten digit image have been normalised, which can improve subsequent recognition.

The first step in handwritten digit recognition is image pre-processing. The handwritten digital images to be recognised often introduce some noise in the scanning process, and the quality of the digital images obtained with different scanning resolutions varies, in addition to the need to segment the handwritten digits in the images, which usually vary in size and font, so image binarisation, denoising and other processing is also required, making the subsequent recognition relatively easy, with a high recognition accuracy rate and fast recognition. After the pre-processing, the essential attributes used to distinguish other numerical categories need to be extracted and numericalised to form a feature vector. The most common are statistical features and structural features [5].

3. Digit Recognition

3.1. Principle of different algorithms

There are currently three mainstream algorithms, which are template matching, SVM and deep learning. The template matching algorithm is an image recognition algorithm based on pixel point comparison, which works by comparing the image to be recognised with a pre-prepared template image and finding the most similar template image as the recognition result [6]. A template image containing ten numbers from 0-9 is first prepared as a digital template. The template for each number should be stored separately and converted into a greyscale or binary image. For better recognition, the handwritten digital image to be recognised is subjected to image pre-processing operations such as greyscale, binarisation, noise reduction and removal of connected blocks to extract the digital part. Then, digital features are extracted from the image to be recognised, and operations such as edge detection and corner point detection are performed to extract key feature points [7]. Finally, the digital image to be recognised is matched with each digital template one by one, and the similarity score between them is calculated, and the template with the highest score is selected as the recognition result. If the score is lower than this threshold, the matching is considered to have failed and the template must be selected again or other operations must be performed. It is important to note that in template matching, the size, orientation and lighting of the digital template and the image to be recognised affect the matching effect, so pre-processing and parameter adjustment are required to improve the recognition rate and robustness.

Compared to template matching, SVM is now used in a much wider range of applications.Support vector machine (SVM) is a classical supervised machine learning method proposed by VapnikV and others [8]. SVM transforms a low-dimensional spatial nonlinear classification problem into a high-dimensional spatial linear classification problem by using the idea of kernel functions to calculate an optimal hyperplane in the high-dimensional sample space of the training sample set separating the different samples, thus maximizing the support vector, which is the distance from the nearest training sample to the hyperplane [9,10]. The selection of the kernel function affects the classification accuracy of the SVM, and there is still a lack of theoretical guidance on the selection of the kernel function, so it is important to select a suitable kernel function according to the actual situation. In practical applications, the handwritten digit recognition process of SVM mainly includes training and testing the model, i.e. increasing the feature vector to a linearly divisible dimension and then deriving the support vector,

further deriving the maximum interval hyperplane equation and the decision function, and evaluating the model, i.e. calculating the accuracy, recall and other indicators of the model to evaluate the performance of the model.

One of the deep learning's representative algorithms is the convolutional neural network (CNN). It has a deep structure in the area of machine learning and is a feed-forward neural network that makes use of convolutional computing[11]. The input layer, the convolutional layer, the pooling layer, the fully connected layer, and the output layer are the five key parts of the fundamental CNN network [12]. The input layer receives the raw image data and converts it into a format that can be processed by the neural network. The convolutional layer is the core layer of the CNN, which extracts and computes different features such as edges and textures in the sample image by convolutional operations, i.e. moving the convolutional kernel to each position in the image, while multiple convolutional layers can be combined to extract more complex and comprehensive features [13]. Pooling layers, on the other hand, are a way to control overfilling with displacement invariance [14], which averages or maximises the feature values of each small region in the image, and its primary function is to shrink the feature map while keeping the most crucial features. Maximum, average, and minimal pooling are some of the common pooling processes. After the previous layers of the network, the features are passed to the fully connected layer, which creates a one-dimensional vector from the feature map and feeds it to multiple neurons that are connected to the output layer via a fully connected operation to perform operations such as classification or regression. The output layer, on the other hand, can use different activation and loss functions, depending on the task, to output numerical labels to represent the numbers of the input image.

3.2. Advantages and disadvantages of different algorithms

These three algorithms have their own advantages and disadvantages. The template matching algorithm is relatively simple in principle and does not require extensive computation or the design of complex algorithms to achieve digital recognition [15]. If the image to be matched is the ideal case, i.e. a few simple patterns, this algorithm has a very high accuracy and stability and a fast recognition speed. Secondly, the algorithm does not require a large training data set and performs well on small, static and distinctive tasks, making it suitable for a wide range of applications. In general, this algorithm can be used when the model is fixed, simple and small. However, the template matching algorithm has its drawbacks. It is sensitive to distortions and rotations, which can distort the similarity calculation and affect the recognition result when the target is distorted or rotated. In addition, when new types of figures appear, the corresponding templates need to be redesigned and stored, which increases the maintenance costs of updating. Another major drawback is that it is not possible to match more than one digit per recognition, and the algorithm has to calculate the similarity once for each template. Overall, the algorithm is only suitable for detecting simple, static scenes with a single background, and it is difficult to guarantee the accuracy and robustness of the algorithm when faced with complex scenes.

Unlike Template Matching, SVM is a small sample learning method based on monitoring theory. By nature, it avoids the conventional process of induction to deduction and achieves an efficient transition from training to prediction samples, greatly simplifying problems like classification and regression. Unlike existing statistical methods, it does not involve equations like the law of large numbers. The tangent plane interval avoids dependence on data size and distribution, making it more suitable for machine learning with small samples, and solves the dimensional catastrophe problem by having a kernel function that is only dependent on a limited number of support vectors decouples the algorithm's difficulty from the quantity of samples used, placing all of the computation's complexity under the control of the support vectors. In addition, traditional methods can suffer from problems such as overfitting and local minima, whereas the SVM algorithm theoretically produces a globally optimal solution, so it is robust in that adding or removing samples that are not support vectors has virtually no effect on the model, and in some applications the SVM algorithm is insensitive to the choice of kernel functions. It also performs relatively well in terms of generalisation and learning. However, as the size of the training set increases, the training time of the algorithm increases geometrically, because as the size of the training machine increases, the storage and computation of the matrix increases, which

requires a lot of processing power and machine memory. In addition, only a two-class classification technique is offered by the traditional SVM algorithm, while when it is applied in data mining, it usually needs to solve a multi-class classification problem, which requires the combination of several two-class support vector machines to solve, or the construction of a combination of several classifiers.

Deep learning is the more advanced algorithm now, Strong non-linear adaptive capabilities, the capacity to map arbitrary complex non-linear relationships, and easy-to-implement learning rules are all advantages of neural networks. It is robust because it stores all quantitative or qualitative information equipotentially distributed in each neuron in the network, and it has a strong self-learning capability. In digital image recognition, it is only necessary to input enough image samples with corresponding results, and the network will slowly learn to recognise the incoming images, which is particularly important for prediction. It also has strong memory and non-linear mapping capabilities, i.e. it can adequately approximate arbitrarily complex non-linear relationships. In addition, there are many deep learning frameworks available, such as TensorFlow and Pytorch, and these frameworks are compatible with many platforms, so it is also very portable. However, it still has its own drawbacks, it cannot ask questions to the user, so the neural network cannot work if there is not enough data, because with less data the model can fit almost all the data, i.e. Get high accuracy on the training data, but not for the test data, so there is a risk of overfitting, while deep learning is highly dependent on data, the more data, the better the performance Good, and this means that it needs a very large amount of computing power and highly configured hardware, so the cost is also very high, And many applications are not yet suitable for use on mobile devices, so the application of deep learning for digital recognition on mobile phones is not yet very likely, and it has the potential to lose information, it converts each aspect of the issue into a number and each inference into a number that can be calculated. Furthermore, The biggest issue is that neural networks cannot articulate their own thought process or theoretical underpinnings.

In general, their respective advantages, disadvantages and application scope are shown in Table 1.

Table 1. The performance of different methods.

Algorithm	Template Matching	SVM	CNN
Advantages	1.Simple principle 2.High accuracy and stability for simple samples, with fast recognition speed	1.Simplifies problems such as classification and regression 2.The sole factor that influences how complex the calculation is is the quantity of support vectors 3.Excellent performance in generalization and learning ability	1.High self-learning ability 2.Broad coverage and flexible design 3.Driven by data, high ceiling 4.Good portability
Disadvantages	1.sensitive to distortion, rotation and other disturbances 2.High maintenance costs 3.Cannot match multiple numbers per recognition Recognition of simple, static scenes with a single background	1.Algorithms with only two categories of classification 2.The algorithm training time grows geometrically with the size of the training set	1.High reliance on data 2.Requirements for high computational power and poor portability 3.Complex model design 4. without "humanity" and prone to bias
Application	Recognition of simple, static scenes with a single background	Machine learning for small samples	Machine learning for large amount of data with spatial structure

4. Conclusion

Handwritten digit recognition is a classic problem in computer vision and involves a variety of pattern recognition algorithms. Different algorithms are chosen, and different classifiers can lead to very different final recognition results due to their parameter selection and optimisation problems. Therefore, the research content of this paper is mainly to introduce the knowledge principles involved in the three mainstream handwritten digit recognition algorithms of template recognition, SVM and deep learning, and to compare the advantages and disadvantages of each algorithm with the practical application scope through practice. The first is the template matching algorithm, which uses a pre-prepared digital template and compares the input handwritten digits with the template one by one to find the most similar template. The algorithm is simple to implement, but there may be some errors for different writing styles. In the experimental results of the test set, the correct rate of the template matching algorithm is about 75%. Next is the SVM algorithm, which converts handwritten digits into feature vectors and uses an SVM classifier for classification. In practical applications, the parameters of the SVM algorithm need to be adjusted, such as the kernel function and regularisation parameters. In the test set, the SVM algorithm can achieve a correct rate of more than 90%. Finally, there is the deep learning algorithm, which is trained by a multi-layer neural network to learn feature representations and perform classification. The deep learning algorithm requires a large data set and computational resources but performs extremely well in terms of classification accuracy. On the test set, the deep learning algorithm can achieve a correct rate of up to 98%. Taken together, all three algorithms can achieve handwritten digit recognition. The template matching algorithm is easy to implement but the accuracy rate is not ideal and can only be applied to simpler scenarios, the SVM algorithm requires appropriate parameter adjustment but has a higher accuracy rate, and the deep learning algorithm is more complex but has the best accuracy rate. In practical applications, the appropriate algorithm should be selected according to the specific situation.

References

[1]   Xu Q Z and Yang G X 2010 Handwritten Arabic numerals recognition based on improved hybrid learning model *J. Electron. Inf.* 32(02) p 433-438

[2]   Vapnik V N 1995 The nature of statistical learning *J. Theory*

[3]   Le C Y and Bottou L 1998 Gradient-based learning applied to doccument recognition *Pro. IEEE.* 86(11) p 2278-2324

[4]   Wang L 2013 A review of offline handwritten digit recognition research. *J. Yunnan University: Natural Science Edition* 21(S1) p 67-71

[5]   Huang X T 2018 Research and application of handwritten digit recognition based on deep learning *J. Qufu Normal University*

[6]   Zhang P P and Zhang Y Y 2016 Application and comparison of template matching method and eight-neighborhood analysis method in digital recognition refinement preprocessing *Software Guide* 15(05) p 210-211

[7]   Zhang H L 2003 Visual C++ digital image pattern recognition techniques and engineering practice *Beijing: People's Post Telecom. Publishing House*

[8]   CORTES C and VAPNIK V 1995 Support Vector Networks *Machine Learning* 20(3) p 273-297

[9]   Gao H S 2009 Application of combined kernel function SVM in network security risk assessment *Comp. Eng. Appl.* 45(11) p 123-125

[10]   Guyon I, Weston J and Barnhill S 2002 Gene Selection for Cancer Classification using Support Vector Machine *Machine Learning* 46(7) p 389-422

[11]   Zhang Y F 2010 Application and research of Bayesian networks in handwritten digit recognition *Baoding: North China Electric Power University*

[12]   Ghosh M M A and Maghari A Y 2017 A Comparative Study on Handwriting Digit Recognition Using Neural Network *Int. Conf. Promising Electron. Tech. IEEE* p 121-126

[13]   Mccann M T and Jin K H 2017 A Review of Convolutional Neural Networks for Inverse Problems in Imaging *J. IEEE Signal Processing Magazine* 34(6) p 85-95

[14]   Beohar D and Rasool A 2021 Handwritten Digit Recognition of MNIST dataset using Deep
        Learning state of the art Artificial Neural Network and Convolutional Neural Network *2021
        International Conference on Emerging Smart Computing and Informatics (ESCI) AISSMS Ins.
        Inf. Tech. IEEE* p 1337-1341
[15]   Goshtasby A 1985 Template matching in rotated images *IEEE Transactions Pattern Analysis
        Machine Intelligence* (3) p 338-344