DOI: 10.54254/2755-2721/5/20230550

English handwriting recognition based on the convolutional neural network

Shengpei Le

Binhai International Cooperative School, No.11 Mingyue Road, Binhai New Town, Ningbo, Zhejiang Province, China 315800

15020440207@xs.hnit.edu.cn

Abstract. Handwriting recognition is widely used in the retrieval, recognition, and management of various information to improve efficiency in various industries. Convolutional neural networks help people get rid of the feature of extracting information feature sets manually and significantly improve recognition efficiency. In this paper, a generalization-enhanced network recognition model is proposed using an improved lightweight convolutional neural network model. An improved method is used to adapt word recognition by the idea of pre-recognition segmentation. In addition, the generalization is enhanced by diversifying and pre-processing the dataset so that the algorithm can obtain noise-resistant performance and detail retention and allow the recognition system to recognize various types of scenes. The results show that the model achieves an accuracy of 93% on the test set. Compared with other classical network models, the model has higher recognition accuracy, faster convergence, and better generalization ability. The system elaborated in this paper can be used for devices with weak computer processing power.

Keywords: manuscript recognition, image processing.

1. Introduction

Human society has always been improving material well-being, pursuing more advanced technology, and an easy, better way of living. The development of modern technology has greatly improved productivity and quality of human life, allowing people to break from heavy labour. However, regular machines could only replace people's work in the physical world but don't help them in solving mental problems, so artificial intelligence (AI) was derived from computer science. AI was created to replace human mind when dealing with large amounts of information. Information input is particularly important in artificial intelligence, where it directly determines the ability, efficiency, and even accuracy of the system to receive information. Pattern recognition is the ideal input state. It allows intelligent systems to learn without the process of manual feature extraction.

With the rise of the idea of "unmanned systems", pattern recognition has quickly become a popular topic. In the past few years, machine learning tasks, such as object detection, translation, handwriting recognition, and speech recognition, have heavily relied on the manual extracting of characteristics and feature sets. This phenomenon changed when various deep learning methods such as Convolutional Neural Networks (CNN): a deep structural network of feedforward neural networks with convolutional computation. Convolutional neural networks are capable of both supervised and unsupervised learning and are modelled after biological visual perception mechanisms [1]. Convolutional neural networks can learn grid-like topology features, like pixels and audio, with low computational effort, stable results, and no additional feature engineering requirements on the data [2] thanks to the sharing of convolutional kernel parameters within the hidden layers and the sparsity of connections between layers), Long Short-Term Memory (LSTM: a special type of RNN, primarily to solve the gradient disappearance and gradient explosion problems during t, Machine's adaptive learning capabilities made the extraction of feature sets rather simple. The mainstream applications of pattern recognition include text recognition, speech recognition, and fingerprint recognition. Text recognition or Optical Character Recognition (OCR) uses the technology of automatic computer recognition of characters, which is widely used in various fields of industry [3].

Handwritten Recognition is a branch of OCR, which focuses on the study and use of optical and computer technology to read out text printed or written on paper. The topic is now commonly used in everyday life, like reading, translation, literature retrieval, sorting of letters and parcels, editing and proofreading of manuscripts, aggregation and analysis of a large number of statistical reports and cards etc. [4].

OCR text recognition could be divided into two parts, printed text recognition, and handwritten text recognition. Since the printing body is mostly regular font, its recognition rate has been measured close to 100%. Handwriting recognition, on the other hand, had more difficulty in recognizing due to large individual font variability. Within handwritten text recognition, offline recognition was the most challenging. The other category, online recognition, has a higher accuracy because it can obtain more relevant information, such as the order of the strokes. On contrary, because of its large font variability, low semantic relevance, and credential background interference, offline handwritten character recognition was proved to be having a low recognition rate and accuracy with a demand for manual correction [5]. However, since the latter does not require specific input devices, it is more widely used and has become the focus of current OCR research, around which this paper is also focused.

This paper uses a modified LeNet-5 lightweight convolutional neural network model. Compared with other networks such as MLP and LeNet-1, this network contains relatively fewer parameters and obtains better results. This allows the recognition system to be used with maximum performance on even energy-inefficient devices. On top of that, the author uses feature enhancement to improve model generalization, for instance, inflation processing, and greying processing. A network recognition model enhanced by generalization is proposed, adding a dropout layer to improve the recognition accuracy. The model has higher recognition accuracy, lower computing requirements, faster model convergence, and a better generalization effect.

2. Method

2.1. Dataset selection

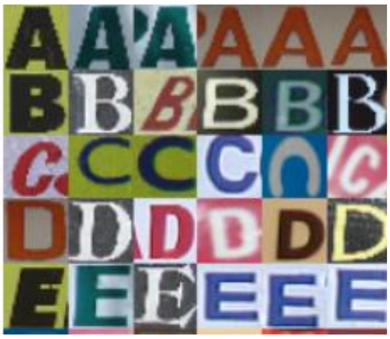

Figure 1. An example of the dataset.

This paper use both Chars74K dataset (printed grey images) [6], and EMNIST dataset (handwritten RGB images) [7]. The diverse dataset enables the trained model to have better generalization. The test set contains a total of 20,800 images and the training set contains a total of 124,800 images. Figure 1 shows a example on the dataset.

After parsing, the author obtains the labels for each letter (1-26 for letters A-Z, case-insensitive) and places the dataset labels into the class of documents so that it is easier to read the data. Next, defines the hyperparameters, and sets the training batch and the minimum size of training. Lastly, converts the images to grayscale and normalizes the dataset to (0,1).

2.2. Dataset Processing

Due to the extremely high demands on devices, datasets, and algorithm fusion for recognizing whole words in a single pass, the authors try to recognize a word by first separating it into letters. Firstly, the image is processed by Hough Line Detection. Using the Hough line detection technique, the line detection task is completed by locating the peak in the parameter space. This approach converts the line detection problem in the image space to the detection problem of points in the parameter space. The method is resistant to interference and insensitive to the missing parts of straight lines in the image, noise, and other coexisting non-linear structures. The segmentation method is based on edge detection, in other words, the localization and segmentation of characters. An edge is a discontinuity or abrupt change in grayscale values between two different adjacent regions. Therefore, detecting edges means, detecting the place where the grayscale changes significantly. The significant change in grayscale at the edge location can be detected with the help of calculating the differential of grayscale. Using first-order differentiation and second-order differentiation to detect edges, at the edge locations, the magnitude values of first-order differentiation will have local extremes and the magnitude values of second-order differentiation will have over-zero points [8]. The edge detection based on grey values technique, which is a method based on the observation that the edge grey values will exhibit step- or roof-type variations, was utilized by the authors. By normalizing the image, noises in the final segmented letters were reduced. The average of the total image pixels is then subtracted for each pixel and a threshold is set to zero out the background pixels. Morphological image processing (inflation and erosion) is also used for processing to improve contour symmetry and increase the generalization effect. Inflation, also known as "field expansion," is the process of expanding the image's highlighted area such that it is greater than it was in the original. The effect of erosion is a lower highlight region than the original image because the initial brilliant part of the image gets eroded, or "the field is eaten." In the segmentation process, it is easy to have the problem of inaccurate segmentation edges and the existence of fragments, and the difficulty can be solved by deleting the wrong picture with a width of not more than 5 pixels. Finally, this paper gets the boundary of each character, lock the character and save it to local. Figure 2 shows a general view of the processing.

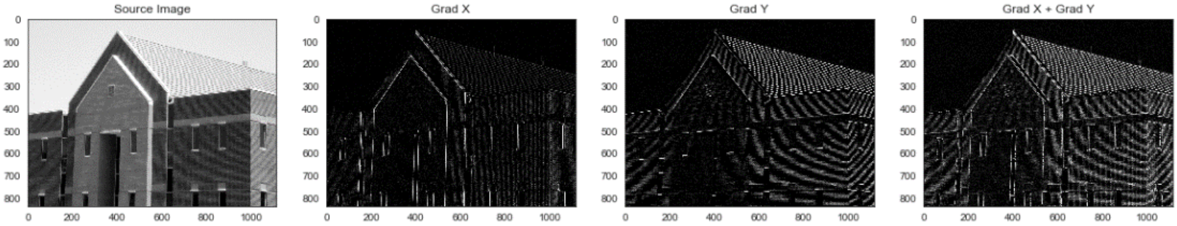

Figure 2. Process View.

2.3. Network Model

The network used in this training for letter recognition is similar to the structure of LeNet-5: two layers of the convolutional network (convolution and pooling) plus three fully connected layers. The convolutional layers use convolutional kernels filters of 5x5 size, and the convolutional kernels slide one pixel at a time, and the same convolutional kernel is used for one feature mapping. Each higher layer node's value is multiplied by the connection's parameters, and these products, along with a bias parameter, are summed to produce a sum. This sum is then fed into the activation function, which outputs the value of the next layer node [9]. Figure 3 shows the network structure of LeNet-5.

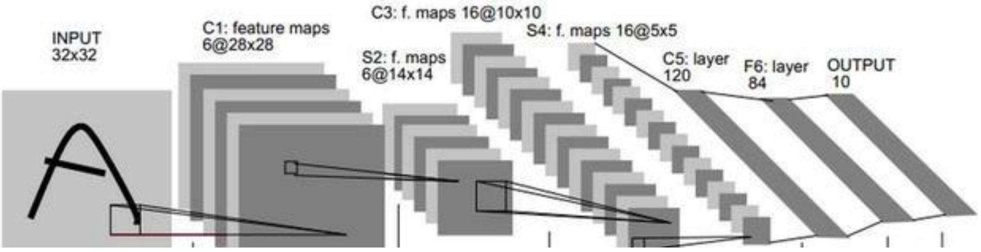

Figure 3. Network structure of LeNet-5.

Input is the input layer, and the size of the input image is uniformly normalized to 32*32.

Layer C1 is a convolution layer with 6 channels, which is obtained by convolving the input image with six 5x5 convolution kernels.

The S2 layer is a downsampling layer with 6 channels, which is obtained by averaging the feature maps of the C1 layer through a 2x2, 2-step window and transforming them using a sigmoid activation function.

C3 is a convolution layer with 16 channels, which is obtained by convolving S2 with 16 5x5 convolution kernels.

S4 is a downsampling layer with 16 channels, which is obtained by averaging the feature maps of the C3 layer through a 2x2 window with a step size of 2 and transforming them using the sigmoid activation function.

C5 is a convolutional layer containing 120 feature maps, obtained by convolving S2 with 120 5x5 convolutional kernels.

F6 is a fully-connected layer containing 84 neurons with a hyperbolic tangent activation function [10].

Output is the output layer, and the 84 neurons of the F6 layer are filled into a SoftMax function to obtain a tensor of output length 10, and the position of 1 in the tensor represents the category to which it belongs.

Finally, the numbers are converted into the corresponding ASCAII characters to obtain the prediction parameters.

2.4. Implementation details

The loss function chosen is the cross-entropy loss function. The cross-entropy loss function is also known as logarithmic loss or logistic loss. Once the model has generated predicted values, the predicted probability for the category is compared with the true value, the resulting loss is calculated, and then a penalty term in logarithmic form is set based on this loss. When training the model, the cross-entropy loss function is used with the aim of minimizing the loss, since the smaller the loss the better the model. The cross-entropy is a representation of two probability distributions p and q. P is the distribution of the actual markers, and q denotes the trained model's projected marker distribution. The similarity between p and q is gauged by the cross-entropy loss function. A fixed-step decay is also used. The learning rate is reduced to a fraction of the original gamma every certain number of steps (or epoch), using fixed-step decay still defines the optimizer first, and then binds objects to the optimizer. This allows the learning rate to be larger at the beginning of training, making the network converge quickly, and smaller at the end of the training, making the network converge better to the optimal solution. To reduce memory requirements and improve efficiency, the Adam optimization algorithm is used. The algorithm uses the mini-batch gradient descent method to calculate dw and db, computes the Momentum exponentially weighted average, and calculates the corrected deviation of Momentum and RMSprop by updating and continuously updating the weights [11]. The algorithm is relatively simple to adjust the reference and can calculate the adaptive learning rate to accelerate the learning speed. The dropout is also set to 50%, which can effectively alleviate the overfitting problem of the model, thus making it possible to train deeper and wider networks. In the training process, this study cycles according to the defined Epoch, and each Epoch is divided into several processes, generally Process=Total/Epoch, and save the trained parameters after the training is completed to facilitate secondary adjustment as well as training.

In the first half of the training, the model does not converge to train loss and the test loss tends to be constant, and then the fixed learning rate is changed to a fixed step decay. Since the softmax input feature is calculated from two parts: one part is the input data, and the other part is the weights of each layer, etc., reduce the initialization weights to make the input feature of softmax in a relatively small range. Table 1 presents the more details about the testing environment.

Table 1. Testing environment.

Pytorch 1.11.0	Python 3.8	Cuda 11.3
GPU：3090	Epoch=10	batchsize=128

3. Results and Discussion

The training process is visualized using tensorboard and the final test set maintains a correct rate of 93% shown in Table 2. The system implemented in this paper is actually a handwritten digit recognition system based on a convolutional neural network. The system can quickly implement handwritten English recognition with a high success rate of recognition. Figure 4 present the visualized results and Table 3 shows the testing result of different types of writing.

Table 2. Test set results.

Eopch: 20	train loss: 0.1348	test accracy:0.93072	step: 1000

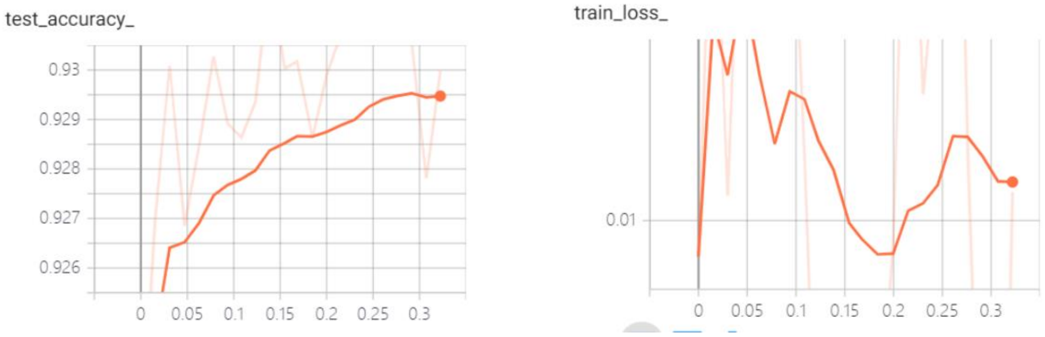

Figure 4. Visualization of results.

Table 3. Testing result of different types of writing.

Writing standard	Input character number	Repeated times	Correct recognition rate
Standardized	5-12	15	93%
Preferable	4-13	15	78.4%
Rather scratchy	8-12	15	40.8%
Illegible	5-8	15	5.31%

The percentage of correct characters in neatly written words and names is high. Among the ten times of inputting Hello World, 8 times are correctly recognized, one time is recognized as Hella Wovld, and one time is recognized as HeII(ii)o World. On the other hand, the recognition rate is very low when the English are written in a rather scribbled manner. It is almost impossible to recognize correctly.

Firstly, there is no good corrective measure for the situation of serious character deformation. Secondly, since this system only uses matching and does not use dictionary checking, firstly, there is no good corrective measure for the case of severe character distortion. Second, since the system uses only matching and not dictionary checking, there are no good corrective measures for outputting wrong words. The first is that there is no good corrective measure for the case of severe character distortion. Secondly, since this system only uses matching and does not use dictionary checking, there is no good corrective measure when outputting wrong words. There is no good corrective measure when outputting wrong words.

For the first problem, the data pre-processing method to deal with outlier points can be improved. According to the box line diagram and each quantile, point to determine whether there is an anomaly, based on the Median Absolute Dispersion (MAD), distance, density, and clustering to distinguish the true anomaly. For the second problem, the dictionary can be added, and in the recognition process, after judging as space, the obtained output will be queried in the word points, if it exists, it will be determined as output, if not, a similar word can be selected as output or the procedure can be modified so that there is a certain bias in the comparison, and when there is no output word in the dictionary, the bias can be modified and the recognition process can be re-entered until the correct result is obtained The second problem, due to the processing of the word, is that it is not possible to get the correct result. For the second problem, there is no countermeasure for the time being because the process will modify the design extensively.

4. Conclusion

The article proposes a recognition method for the English recognition problem that can be used on computers with low computing power. It can be applied to most users and can be used to recognize formal correspondence, notes, etc.

The system could be enriched in many ways, such as adding dictionaries, matching templates based on user selection, etc. If these features were implemented, it would be an easy-to-use and fast input platform. As mentioned at the beginning of this paper, it can be ported to less powerful platforms with fewer changes. The application outlook is very promising. This motivates me to go back and enrich it, refine it, and make it practical in the future.

References

[1] Yu, Q 2019 Semantic segmentation of intracranial hemorrhages in head CT scans [C]. I n 2019 IEEE 10th International Conference on Software Engineering and Service Scie nce (ICSESS) (pp. 112-115). IEEE.

[2] Machine Learning. 2022. Google Machine Learning Education [R]. https://developers.google.com/machine-learning/practica/imageclassification/convolutional-neural-networks

[3] Packt Hub. 2022 What is LSTM? [R], packtpub.com.

[4] Tang W C 2009 Handwritten English character recognition system [D], Doctoral dissertat ion Shenyang University of Technology.

[5] Marti, U V et al. 2002 The IAM-database: an English sentence database for offline han dwriting recognition [J]. International Journal on Document Analysis and Recognition, 5(1), 39-46.

[6] T de Campos. 2012. The Chars74K dataset [R]. http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/

[7] Gregory C. 2017. EMNIST [R]. https://rds.westernsydney.edu.au/Institutes/MARCS/BENS/EMNIST/emnist-gzip.zip

[8] Andrea G et al. 2006. Wavelet based image segmentation [C] Annual Conference Techni cal Computing. 2006, (14th)

[9] Analyticsvidhya. 2021. The architecture of lenet 5 [R], https://www.analyticsvidhya.com/blog/2021/03/the-architecture-of-lenet-5/

[10] Zhang Z 2018 June Improved adam optimizer for deep neural networks [C], In 2018 IE EE/ACM 26th International Symposium on Quality of Service (IWQoS) (pp. 1-2)

[11] Kuo and C C J 2016 Understanding convolutional neural networks with a mathematical model [J], Journal of Visual Communication and Image Representation, 41, 406-413.