World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:13, No:9, 2019

A Recognition Method of Ancient Yi Script Based on Deep Learning

Shanxiong Chen, Xu Han, Xiaolong Wang, Hui Ma

Abstract**—**Yi is an ethnic group mainly living in mainland China, with its own spoken and written language systems, after development of thousands of years. Ancient Yi is one of the six ancient languages in the world, which keeps a record of the history of the Yi people and offers documents valuable for research into human civilization. Recognition of the characters in ancient Yi helps to transform the documents into an electronic form, making their storage and spreading convenient. Due to historical and regional limitations, research on recognition of ancient characters is still inadequate. Thus, deep learning technology was applied to the recognition of such characters. Five models were developed on the basis of the four-layer convolutional neural network (CNN). Alpha-Beta divergence was taken as a penalty term to re-encode output neurons of the five models. Two fully connected layers fulfilled the compression of the features. Finally, at the softmax layer, the orthographic features of ancient Yi characters were re-evaluated, their probability distributions were obtained, and characters with features of the highest probability were recognized. Tests conducted show that the method has achieved higher precision compared with the traditional CNN model for handwriting recognition of the ancient Yi.

Keyword**—**Recognition, CNN, convolutional neural network, Yi character, divergence.

I. Introduction

IN China, the Yi is a mysterious and great nationality. As an important minority script, ancient Yi character exists for more than 8000 years. It is coequal with Oracle bone script, Sumerian, Ancient Egyptian, Mayan and Halal script, and is one of the six ancient scripts in the world. It has been used till now and has left many precious books in history. These books written with ancient Yi script have important historical significance and social value [1], [2]. As the carriers of ancient Yi books, stone inscriptions, rock paintings, wood carving and paper books are often vague or incomplete due to their long history, which brings great challenges to the recognition of ancient Yi script.

In the world, there are relatively few achievements about the recognition of ancient Yi script, mainly from part of the research conducted by Chinese ethnic colleges and universities and research institutes. Jiamei of Yunnan Minzu University and his group have used image segmentation method to recognize ancient Yi script. Firstly, the Yi script is classified, normalized and binarized in the preprocessing, and then the template matching method is used for Yi script recognition [3]. Longhua et al. used the method of combination feature classification to recognize Yi script. The features they used are as follows: directional line element feature, stroke density feature and projective feature. In the process of classification, they used multiple classifiers to vote to determine the final category. Finally, a recognition rate close to 96% was obtained, which is a typical method of feature extraction and classification [4]. Zongxiao and Xianli first extracted the features of the contribution of the peripheral direction in the recognition of the printed Yi character, and compressed the features into 128 dimensions, and then used the three-level distance dictionary matching algorithm based on single feature to recognize, and finally obtained a primary recognition rate of 99.21% in the test set containing 10 kinds of Yi script [5]. In the study of Yi script recognition, it is worth mentioning that in March 2017, Professor Malayi of Southwest Minzu University jointly developed the handwritten recognition technology of Yi script with China Ethnic Languages Translation Bureau, and developed the relevant software of Yi script recognition [6], which effectively promoted the protection and development of Yi script and culture.

Compared with other script recognitions, the handwriting of ancient Yi script is more arbitrary and there is no unified standard, so the recognition is more complicated. Although the existing Chinese and English recognition technology has made great progress, due to the imbalance of historical and regional development, the current research on ancient Yi script recognition is very little. The existing ancient Yi script is basically in the handwritten form, and the diversity of handwritten form undoubtedly increases the difficulty of recognition [3], [6], [7].Therefore, the recognition of ancient Yi script is a challenging pattern recognition problem, which is mainly manifested as follows:

(1) There is no mature handwritten sample library. Handwritten sample library is the key factor for the success of ancient Yi script recognition, and directly determines the effect of recognition. At present, the research of ancient Yi script is still mainly concentrated on the collection of ancient Yi literature. Few people have specially studied the recognition of ancient Yi script, and the available libraries of ancient Yi script samples are scarce [8], [9].
(2) The character set is huge. Ancient Yi script has a huge character set. The Yunnan-Sichuan-Guizhou Yi Script Character Collection has been published in 2004. It contains more than 87,000 characters [8]. It is a difficult task to classify such a large character set.
(3) There are many changes in the fonts of ancient Yi script,

Shanxiong Chen*, Xu Han, and Xiaolong Wang are with the College of Computer and Information Science, Southwest University, Chongqing, China,400715 (*e-mail: csxpml@163.com).
Hui Ma is with the Research institute of Yi nationality, Guizhou University of Engineering Science, Bijie ,Guizhou, China,551700

and there is no unified standard. Writing rules are different in different regions, and styles and formats change a lot, which increases the difficulty of recognition.

(a) Stone inscriptions (b) Wood carving (c) Sheepskin

Fig. 1 The ancient Yi script in stone inscriptions, wood carving and sheepskin

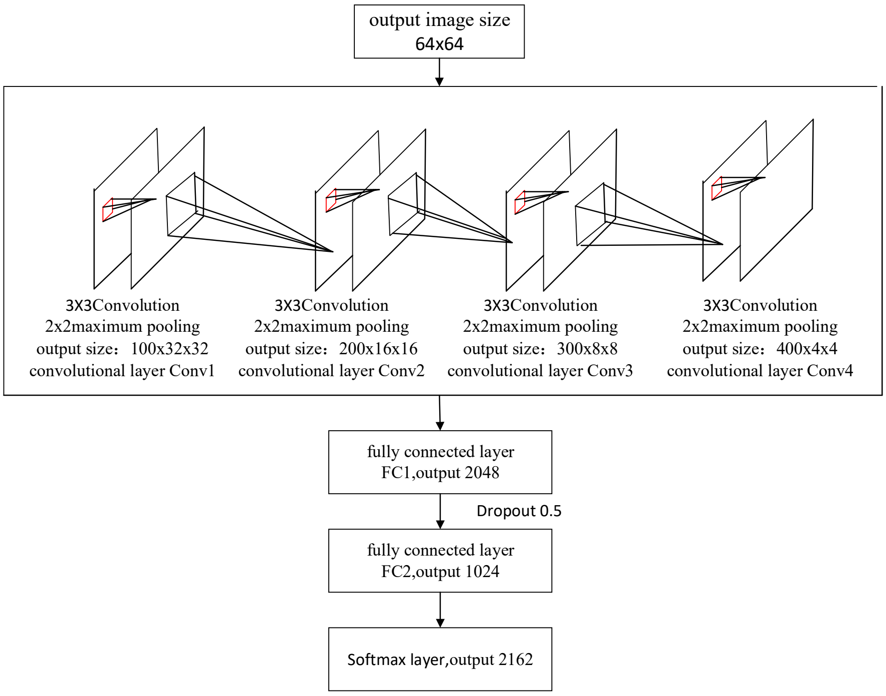


Fig. 2 The Structure of Model M0

In this paper, the convolution neural network in deep learning is used to recognize ancient Yi script. Five models are extended on the basis of four-layer CNN, and the output neurons of the five models are re-coded by using Alpha-Beta divergence as penalty term. Then two fully connected layers are used to complete feature compression. Finally, the ancient Yi characters are re-graded by softmax layer, and their probability distribution is obtained. The corresponding one with the highest probability is selected as the recognition character.

II. Basic Network Structure

In this study, a four-layer convolution neural network is constructed for the recognition of ancient Yi characters [9]-[11], which is called model M0, as shown in Fig. 2. M0 consists of 4 convolutional layers, 2 fully connected layers and 1 softmax layer. Its symbolic description is as follows: Conv: Convolutional layer; MP: Max Pooling layer; Drop: Dropout layer; Softmax: Softmax layer; FC: Fully Connected layer. For example, Conv (3x3, 64, S2, P1) is represented as a convolutional layer with size of 3×3, Output channel number of 64, stride = 2, and padding = 1. By default, Conv's step size is 1, and the fill is 1. This is mainly to make the feature map equal in size before and after convolution. By default, the size of MP is 2×2, the stride is 1, the padding is 0, and then the size of the feature map becomes 1/4 of the previous layer.

As for as Model M0, the symbolic description is as follows: Input(64x64x1) – Conv(3x3, 100) - MP – Conv(3x3, 200) – MP - Conv(3x3, 300) – MP - Conv(3x3, 400) – MP – FC(2048) – Drop(0.5) – FC(1024) – Softmax(2162).

Based on Model M0, an additional convolution layer 3×3 is added in front of each convolution layer to obtain four models: M1, M2, M3, M4. The symbolic description of all models is shown in Table I. M1 adds a convolution layer with a channel number of 50 before the first convolutional layer; M2 adds a convolutional layer with a channel number of 150 before the second convolutional layer; M3 adds a convolutional layer with a channel number of 250 before the third convolutional layer; M4 adds a convolutional layer with a channel number of 350 before the fourth convolutional layer; M5 applies all operations of M1-M4 to Model M0, and simultaneously adds a 3×3 convolutional layer in front of all convolutional layers. Model M5 is obtained as shown in Fig. 2.

The structure of Model M5 is shown in Fig. 3. The network consists of four large convolutional layers, two fully connected layers, and one softmax layer. The output of the other layers except softmax is activated by using the ReLU function. Each of the large convolutional layers is composed of two consecutive 3×3 convolutional layers and one 2×2 maximum pooling layer, and the number of output channel of the convolution kernel is incremented by 50. To further standardize the model description, the four convolutional layers contained in Model M5 are formally described as follows:

TABLE I
SYMBOLIC DESCRIPTION OF THE M0-M5 MODELS

Model	Symbolic description
M0	Input(64x64x1) – Conv(3x3, 100) - MP – Conv(3x3, 200) – MP - Conv(3x3, 300) – MP - Conv(3x3, 400) – MP – FC(2048) – Drop(0.5) – FC(1024) – Softmax(2162)
M1	Input(64x64x1) – Conv(3x3, 50) - Conv(3x3, 100) – MP – Conv(3x3, 200) – MP - Conv(3x3, 300) – MP - Conv(3x3, 400) - MP – FC(2048) – Drop(0.5) – FC(1024) – Softmax(2162)
M2	Input(64x64x1) – Conv(3x3, 100) – MP - Conv(3x3, 150) – Conv(3x3, 200) – MP - Conv(3x3, 300) – MP - Conv(3x3, 400) – MP – FC(2048) – Drop(0.5) – FC(1024) – Softmax(2162)
M3	Input(64x64x1) – Conv(3x3, 100) - MP – Conv(3x3, 200) – MP - Conv(3x3, 250) - Conv(3x3, 300) – MP - Conv(3x3, 400) – MP – FC(2048) – Drop(0.5) – FC(1024) – Softmax(2162)
M4	Input(64x64x1) – Conv(3x3, 100) - MP – Conv(3x3, 200) – MP - Conv(3x3, 300) – MP - Conv(3x3, 350) - Conv(3x3, 400) – MP – FC(2048) – Drop(0.5) – FC(1024) – Softmax(2162)
M5	Input(64x64x1) - Conv(3x3, 50) – Conv(3x3, 100) – MP - Conv(3x3, 150) – Conv(3x3, 200) – MP - Conv(3x3, 250) - Conv(3x3, 300) – MP - Conv(3x3, 350) - Conv(3x3, 400) – MP – FC(2048) – Drop(0.5) – FC(1024) – Softmax(2162)

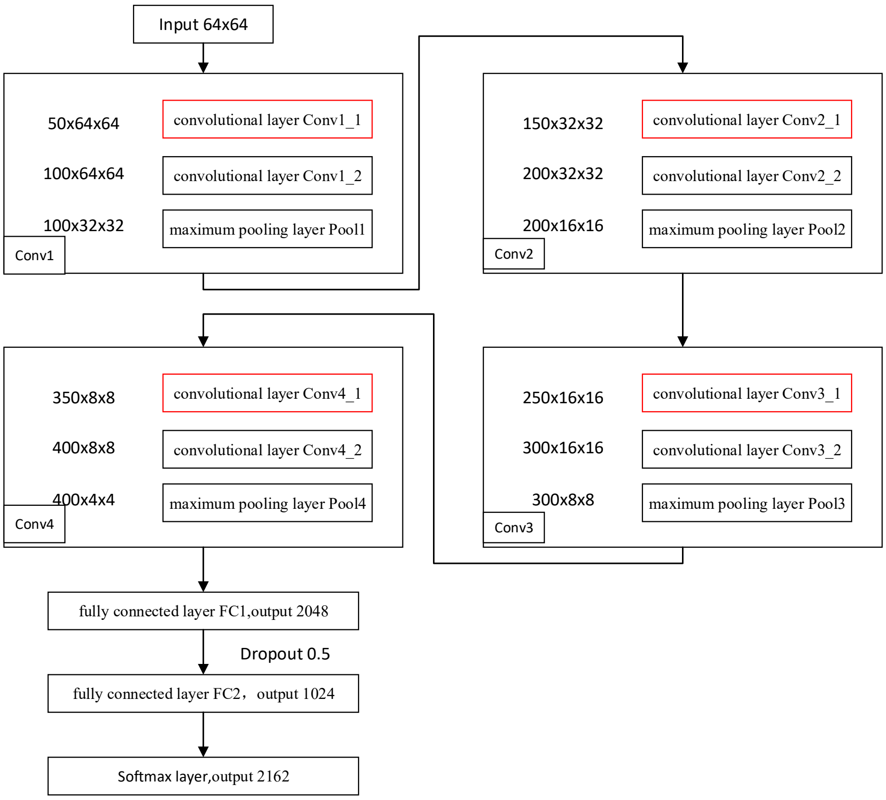


Fig. 3 Model M5: Red parts are additional convolution layers

Layer 1:

Conv1: Conv1_1 (Parameter number:4K), Conv1_2 (Parameter number:405K), Pool1

Number of channels: 50, 100, 100

Output size: 32×32

In total: 100

For the first convolutional layer Conv1, it consists of Conv1_1, Conv1_2, Pool1, and its numbers of channel are 50, 100, 100 respectively. The output of this layer is 100 feature maps with the size of 32×32. The parameter number of Conv1_1 and Conv1_2 are 4k and 405k, respectively; and the parameter number of the entire convolutional layer Conv1 is 409k.

Layer 2:

Conv2: Conv2_1 (Parameter number:1215K), Conv2_2 (Parameter number:2430K), Pool2

Number of channels: 150, 200, 200

Output size: 16×16

In total: 200

Layer 3:

Conv3: Conv3_1 (Parameter number:4050K), Conv3_2

(Parameter number:6075K), Pool3

Number of channels: 250, 300, 300

Output size: 8×8

In total: 300

Layer 4:

Conv4: Conv4_1 (Parameter number:8505K), Conv4_2 (Parameter number:11340K), Pool4

Number of channels: 350, 400, 400

Output size: ×4

In total: 400

As far as the fully connected layer and the softmax layer are concerned, their parameters are 13107k, 2097k, and 2214k, respectively. Among them, the dropout proportion of the first fully connected layer is 0.5. The parameters of the entire network are about 51442k in total.

III. Auto Encoder Structure of Alpha-Beta Divergence

The expressive ability of a single CNN model is limited. Different models have different abilities to solve the same problem and have their own preferences. It is needed to consider the reliability of different models for local category recognition and to consider the classification effect of different models for the whole as well [12], [13]. In this study, a generalization divergence is constructed. The Alpha-Beta divergence is used as penalty term to re-encode the output neurons of the Mode M0-M5 [14], [15], and then the features of the two fully connected layers are compressed. Finally, a Softmax layer is used to re-grade the ancient Yi character [16], and the probability distribution is obtained.

Assuming that P and Q are two probability density functions in the same space, the Alpha-Beta divergence between them can be expressed as [15]:

$$D_{AB}^{(\alpha,\beta)}(\mathrm{P}\,\|\,\mathrm{Q}) = -\frac{1}{\alpha\beta}\sum_{it}(p_{it}^{\alpha}q_{it}^{\beta} - \frac{\alpha}{\alpha+\beta}p_{it}^{\alpha+\beta} - \frac{\beta}{\alpha+\beta}q_{it}^{\alpha+\beta}) \quad (1)$$

in which, $\alpha, \beta, \alpha+\beta \neq 0$

The above formula conforms to the following constraints:

$$\frac{1}{\alpha\beta}p_{it}^{\alpha}q_{it}^{\beta} \leq \frac{1}{\beta(\alpha+\beta)}p_{it}^{\alpha+\beta} + \frac{1}{\alpha(\alpha+\beta)}q_{it}^{\alpha+\beta} \quad (2)$$

in which, $\alpha, \beta, \alpha+\beta \neq 0$

Alpha-Beta divergence is extended to cover all sets of real numbers in order to avoid uncertainty and singularity under a certain value, $\alpha,\beta \in \mathrm{R}$, so Alpha-Beta can be expressed more directly as:

$$D_{AB}^{(\alpha,\beta)}(\mathrm{P}\|\mathrm{Q}) = \sum_{it} d_{AB}^{(\alpha,\beta)}(p_{it}, q_{it}) \quad (3)$$

in which,

$$d_{AB}^{(\alpha,\beta)}(p_{it},q_{it}) = \begin{cases} \left(-\frac{1}{\alpha\beta}(p_{it}^{\alpha}q_{it}^{\beta} - \frac{\alpha}{(\alpha+\beta)}p_{it}^{\alpha+\beta} - \frac{\beta}{(\alpha+\beta)}q_{it}^{\alpha+\beta})\right), \alpha,\beta,\alpha+\beta\neq 0 \\ \frac{1}{\alpha^2}\left(p_{it}^{\alpha}\ln\frac{p_{it}^{\alpha}}{q_{it}^{\alpha}} - p_{it}^{\alpha} + q_{it}^{\alpha}\right), \alpha\neq 0, \beta\neq 0 \\ \frac{1}{\alpha^2}\left(\ln\frac{q_{it}^{\alpha}}{p_{it}^{\alpha}} - \left(\frac{q_{it}^{\alpha}}{p_{it}^{\alpha}}\right)^{-1} - 1\right), \alpha=-\beta\neq 0 \\ \frac{1}{\beta^2}\left(q_{it}^{\beta}\ln\frac{q_{it}^{\beta}}{p_{it}^{\beta}} - q_{it}^{\beta} + p_{it}^{\beta}\right), \alpha=0, \beta\neq 0 \\ \frac{1}{2}\left(\ln p_{it} - \ln q_{it}\right)^2, \alpha,\beta=0 \end{cases} \quad (4)$$

In this study, Alpha-Beta divergence is used as a penalty term to re-encode the M0-M5 output neurons (as shown in Fig. 4), and then feature extraction is carried out through two full connections. The aim is to globally optimize the M0-M5 model and improve the recognition accuracy of ancient Yi character.

According to the principle of the neural network of auto-encoder [19], [20], we need to learn a function, $h_{w,b}(x) \approx x$, namely, $\hat{x} \approx x$, use $a_j^{(2)}(x)$ to indicate the activation of hidden neuron *j* in auto-encoding neural network, when the given input is x. Furthermore, the parameters are processed as:

$$\hat{\rho}_j = \frac{1}{m}\sum_{i=1}^{m}\left[a_j^{(2)}(x^{(i)})\right], \hat{\rho}_1 = a_1^{(2)}, \hat{\rho}_2 = a_2^{(2)}, \hat{\rho}_3 = a_3^{(2)} \quad (5)$$

$\hat{\rho} \to \rho$ is the general mandatory constraint, ρ is a sparse parameter, which generally takes a value close to zero. For example, 0.05, that is, the average activation value of each hidden unit *j* is close to 0.05. Here, $\hat{\rho}_j$ is the average activation value of the hidden unit *j*. In this study, an additional penalty term is added to optimize the objective function and the divergence of Alpha-Beta.

$$D_{AB}^{(\alpha,\beta)}(\rho\,\|\,\hat{\rho}_j) = -\frac{1}{\alpha\beta}\sum_{it}(\rho_{it}^{\alpha}\hat{\rho}_{it}^{\beta} - \frac{\alpha}{\alpha+\beta}\rho_{it}^{\alpha+\beta} - \frac{\beta}{\alpha+\beta}\hat{\rho}_{it}^{\alpha+\beta}) \quad (6)$$

Therefore, the global loss function is:

$$J_{sparse}(W,b) = J(W,b) + \beta\sum_{j=1}^{s_2} D_{AB}^{(\alpha,\beta)}(\rho\,\|\,\hat{\rho}_j) \quad (7)$$

in which,

$$\begin{aligned} J(W,b) &= \left[\frac{1}{m}\sum_{i=1}^{m} J(W,b;x^{(i)},y^{(i)})\right] + \frac{\lambda}{2}\sum_{l=1}^{n_l-1}\sum_{i=1}^{s_l}\sum_{j=1}^{s_l+1}(W_{ji}^{(l)})^2 \\ &= \left[\frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{2}\left\|h_{\mathrm{W},b}(x^{(i)} - y^{(i)})\right\|^2\right)\right] + \frac{\lambda}{2}\sum_{l=1}^{n_l-1}\sum_{i=1}^{s_l}\sum_{j=1}^{s_l+1}(W_{ji}^{(l)})^2 \end{aligned} \quad (8)$$

Here, β controls the weight of the sparsity penalty factor, $\hat{\rho}_j$ also (indirectly) depends on *W*, *b*. Because it is the average activation degree of the next hidden layer neuron *j* of M0-M5 as the input layers, the activation degree of the hidden layer neurons depends on W,b. So far, the re-encoding of the M0-

World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:13, No:9, 2019

M5 layer neurons is completed.

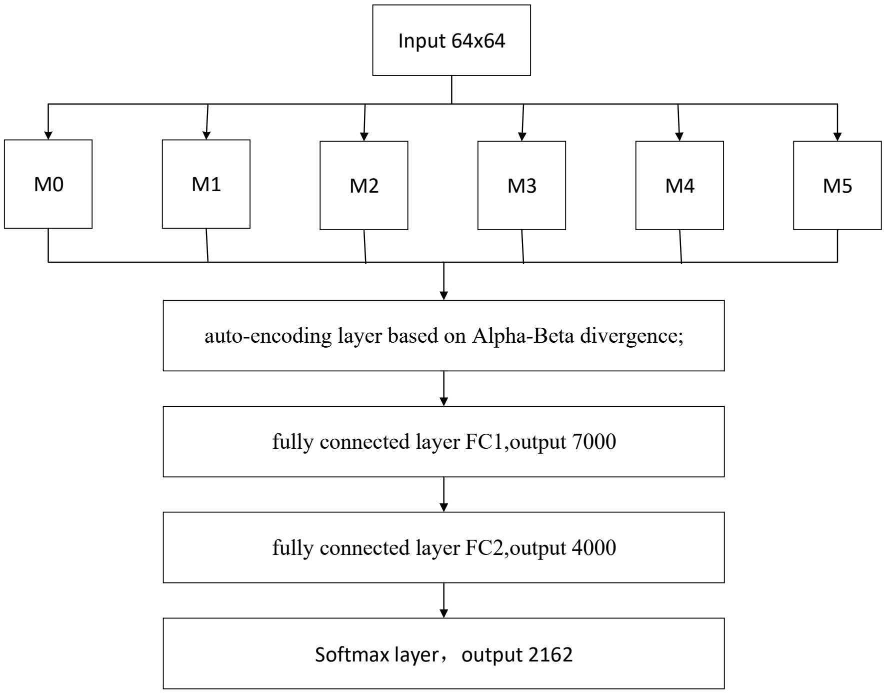


Fig. 4 Auto-encoder fusion model M6 based on Alpha-Beta divergence

IV. Model Training and Sample Collection

A. Model Training

In this study, the activation function of the model is set to be ReLU [19], and the optimization algorithm selects Adam [20], [21]. At the same time, the training set is incremented to enlarge the capacity of the training set, so that the model can learn the features of the image more fully. In addition, in order to make the model converge smoothly, a Batch Norm layer is attached to each convolution layer.

Although the Adam algorithm is computationally efficient, easy to implement and takes up less memory, the update step size and the gradient size are independent of the initial learning rate. But the optimization effect is not obvious after the incremental data set. When the learning rate is 0.001, the loss function cannot converge, and when the learning rate is set to be 0.0001, the loss function starts to decrease, as shown in Fig. 5, so the initial learning rate is set to be 0.0001 in this study.

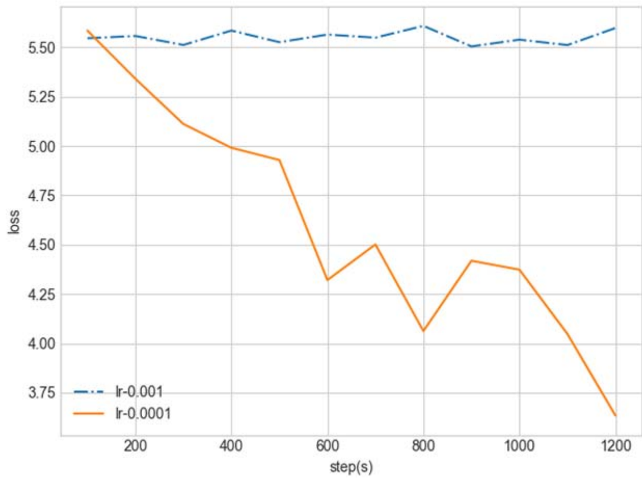
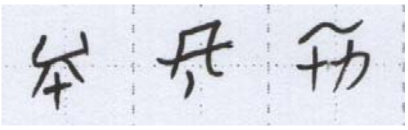


Fig. 5 Model M0 loss function at different learning rates

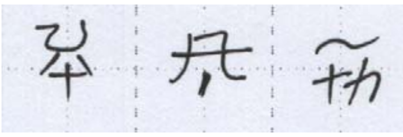
B. Sample Collection

The sample was selected 2142 commonly used ancient Yi characters from the 370,000-word "Yi Records in Southwest China" [22], and invited 400 Yi teachers and students to copy 1200 collection forms (as shown in Fig. 6). Among them, there are 800 Yi upright-letter collection tables, 200 soft-blush style collection tables and 200 pen-and-ink style collection tables. As shown in Fig. 7. 151200 font samples are obtained. At the same time, considering the convenience of processing and analysis, the corresponding font library (as shown in Fig. 8) and ancient Yi input methods are designed.

Fig. 6 Scanned sample of the collection table



(a) Pen-and-ink characters of Yi script



(b) Soft-blush characters of of Yi script

Fig. 7 The ancient characters of Yi script

Fig 8 Font library of ancient Yi

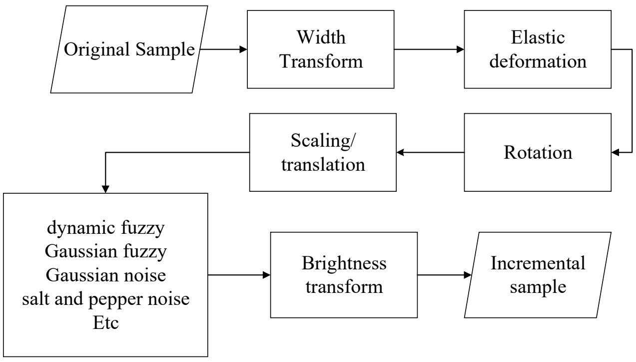


Fig. 9 Sample increment

To ensure that the model can learn enough features, the number of samples is augmented. The process of sample increment is shown in Fig. 9. Firstly, the width transform and elastic deformation of the original samples are performed, then the affine transformation of rotation, scaling and translation are performed, then the fuzzy noise is added, and finally the brightness and contrast are transformed. The sample after increment is shown in Fig. 10. Furthermore, 70% of the character samples are selected as training set and 30% as test set, which are called training set A and test set B respectively. At the same time, in order to quickly evaluate the model in the training process, a small batch of data is selected from the test set as the validation set C. In addition, in this study, the augmented technology is adopted to conduct sample increment on the training set A to get the training set A2.

Fig. 10 The sample after increment

After the above processing, we got training set A. To verify effect of the proposed method in this article, we select 20 documents from “bu dou bu zhou shu” for Yi Classics which is offered by Research Institute of Yi nationality, Guizhou University of Engineering Science. Every document includes about 150 characters. So there are about 3000 characters for test set. In addition, in this study, the augmented technology is adopted to conduct sample increment on the training set A to get the training set A2.

V. Experiment and Analysis

In this study, model M0-M5 are trained by training set A2 and tested on verification set C and test set B, as shown in Table II. The accuracy of the model on test set B is much higher than that on training set A2, which indicates that the model used in this study learns enough handwriting styles through the incremental training set, and also shows that the augmented method proposed in this study is feasible and effective. But on the other hand, after enough training, the accuracy of the training set is still no more than 90%, which indicates that the training data set used in this paper may have some difficulty in identifying samples. After the training data set was manually checked, it was found that handwriting loss, adhesion, and excessive blurring occurred in some samples in the process of incremental deformation due to the low resolution of the original samples, as shown in Fig. 11. As for the same character, it is almost completely deviated from the normal writing style of the word due to excessive deformation. It is these over-deformed samples that introduce additional noise into the model, making the model perform poorly on the training set.

TABLE II
Accuracy Rate of Model M0-M5

data set	M0	M1	M2	M3	M4	M5
A2	82.22	89.59	85.99	85.58	85.03	81.38
B	90.75	92.84	91.98	91.28	90.97	90.06

Fig. 11 Data after over-deformation

It can be learned from Table II that the performance of the model M1 is optimal, the accuracy rate on the test set B reaches 92.84%, and the worst accuracy rate is M5, which is only 90.06%. Compared with the simplest model M0, additional convolutional layers are added in each layer, which is obviously beneficial to the improvement of the model performance, but because the model inserts in a backward position, the increase speed of performance is gradually reduced. Model M5 adds an additional convolutional layer before each convolutional layer, expecting to achieve better results. The reality is that the performance of the entire model is significantly reduced. In addition, Table IV shows the time cost for 100 iterations of different models, and there is no doubt that the simplest model M0 takes the least time, only 237.27 s. The time consumed by Model M5 is as high as 563.40 s. At the same time, with the rise of the position of the convolutional layer, the parameters will increase and the corresponding time will increase sharply. However, compared M3 with M2 and M4, the consumption time decreases, which is mainly as the number of layers increases, the size of the feature map is reduced, and the cost of computation becomes less, as shown in Table III. Compared with Model M0, Model M2 increases the number of connections up to 2.07e9, while the numbers of connection of Model M3 and Model M4 decrease, in which the minimum number of connection of Model M4 is only 0.65e9.

TABLE III
THE NUMBER OF CONNECTIONS INCREASED BETWEEN MODEL M1-M4 AND M0

	M1	M2	M3	M4
Convolutional layer	Conv1	Conv2	Conv3	Conv4
Number of connections increased	1.64e9	2.07e9	1.34e9	0.65e9

It can be seen from Tables II and IV that M1 has the best comprehensive performance, which can maximize the performance of the model with only a small computation cost. And M5 is obviously the worst choice, spending almost twice the computing time of M0, the performance does not rise, instead, it falls.

TABLE IV
TIME CONSUMED BY MODEL M0-M5 ITERATION 100 TIMES (SECOND)

M0	M1	M2	M3	M4	M5
237.27	305.87	404.40	282.4	273.5	563.40

Through further analysis, it is found that convolution operation after convolutional layer is helpful to improve the performance of the model, but the effect of the lower layers is much best than that of the higher layers, and the cost of adding of the lower layers is also minimal (the amount of parameters are small). The effect of adding convolutional layer to higher layers is not obvious and it brings huge computation cost (a large number of parameters are introduced). Model M5 adds convolution directly to each layer, which obviously leads to gradient diffusion as well as a large amount of computation.

In order to synthesize the advantages of each model, model M6 in Fig. 3 is used in the experiment. The output probability distribution of the models M0-M5 is taken as input, trained on the training set A2, and tested on the test set B. Its accuracy on test set B reached 93.97%, and the accuracy on the training set A2 reached 90.63%.

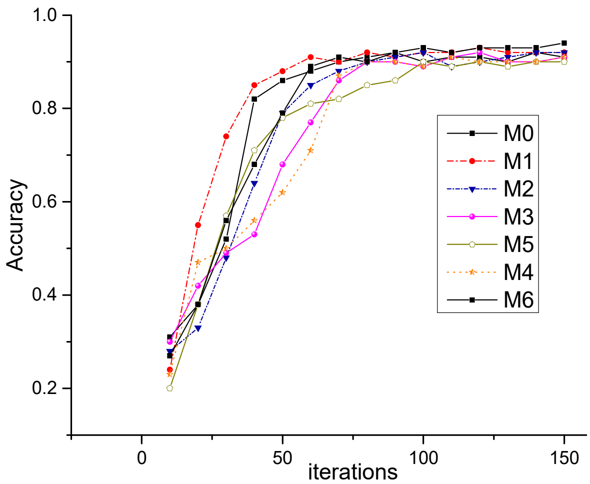


Fig. 12 Accuracy change of Models M0-M6 on test set B

Further experiments analyzed the change of the model accuracy, Fig. 12 shows the accuracy change of the models M0-M6 as the number of iterations increases on the test set (we found that the accuracy of the model tends to be stable between 100 and 150 iterations. If we increase iterations further, there is no change for accuracy significantly). It can be seen clearly that model M5 is obviously inferior to other models, which rises slowest and reaches the highest accuracy rate of 91.06% in the 12th cycle. Models M0 and M1 achieved the best accuracy rate of 91.54% and 92.84% in the 8th cycle, while model M6 reached the best accuracy rate of 92.97% in the 7th cycle. At the same time, it can be seen that the rising speeds of models M0, M1, M2 and M6 are relatively close to each other, and model M6 reached a relatively stable state at the 7th cycle, and reaches a stable state earlier than other models. In general, model M6 is superior to other models, which is the result of M6 re-coding and optimizing the output neurons of other models.

VI. CONCLUSIONS

In this study, the ancient Yi script sets are recognized by deeply learning CNN network and the recognition accuracy is high. Especially, Alpha-Beta divergence is used as a penalty term to re-code the output neurons of each model to generate fusion scheme, which can improve the performance of the whole recognition network with limited computational ability, and avoid the performance degradation caused by the increase of network layers as well. At present, the research of Yi script recognition is preliminary. It mainly extracts text from handwritten and printed scripts, and because of the limited character library, it is limited to process the common Yi scripts. And there are few studies on the recognition of the ancient Yi script in the ancient books, which is important for investigation to Yi culture. Our study combines deep learning with the processing of ancient scripts of ethnic minorities, and also makes some beneficial exploration on cultural protection and development. In addition, considering that the fusion scheme for re-auto-encoding the output of each model is measured by the probability distribution method, a large number of samples are needed, and the cost of handwritten samples by the Yi people is relatively high, it is proposed to use the Generative Adversarial Networks (GANs) of the deep learning to generate more ancient Yi handwritten samples in the subsequent study.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (41271292), China Postdoctoral Science Foundation (2015M580765), and Chongqing Postdoctoral Science Foundation (Xm2016041), the Fundamental Research Funds for the Central Universities, China (XDJK2018B020)

REFERENCES

[1] C.X Zhu, “The Study of the Yi Language Ancient Document”. Nationalities Publishing House,2008.
[2] J Gao, J.Z Liu, “Problems and Countermeasures of the Digitalization of Ancient Books in Mainland China”.Journal of Library Science In China, vol. 2013, pp. 110-119, 2013.
[3] J,W Wang, Y.L Wen ,Y.Q Li ,Y.L Gao, “The Recognition System of Old-Yi Character Based on the Image Segmentation”. Journal of Yunnan Nationalities University: Natural Sciences Edition, vol. 17, pp. 76-79, 2008
[4] L.H Zhu, J.W Wang, “Off-Line Handwritten Yi Character Recognition Based on the Multi-Classifier Ensemble with Combination Features”. Journal of Yunnan Nationalities University: Natural Sciences Edition, vol. 19, pp. 329-333, 2010.
[5] Z.G Zhu, X.L Wu, “Principles and Implementation of an Off-Line Printed Yi Character Recognition System”. Computer Technology and Development, vol. 22, pp. 85-88, 2012.
[6] S Liu, Y.D Li, “Design and Realization on Character Segmentation Method for Yi Language”. Journal of South-Central University for Nationalities: Natural Sciences Edition,vol. 26, pp. 74-76, 2007.
[7] B Wu, Analysis of Yi Characters Based on Character Recognition. Journal of Southwest Minzu University (Humanities and Social Science), pp. 47-53, 2018.
[8] Yi collaboration group of Yunnan, Sichuan, Guizhou and Guangxi in China, “A Collection of Yi Characters in Yunnan and Sichuan”. Yunnan nationalities press,2004:251-273
[9] X. H. Ren, Y. Zhou, J. H. He, K. Chen, X. K. Yang, and J. Sun, "A Convolutional Neural Network-Based Chinese Text Detection Algorithm via Text Structure Modeling," IEEE Transactions on Multimedia, vol. 19, pp. 506-518, Mar 2017.
[10] M. A. H. Akhand, M. Ahmed, M. M. H. Rahman, M. M. Islam, "Convolutional Neural Network Training incorporating Rotation-Based Generated Patterns and Handwritten Numeral Recognition of Major Indian Scripts," IETE Journal of Research, vol. 64, pp. 176-194, 2018.
[11] A. Nasee, K. Zafar, "Comparative Analysis of Raw Images and Meta Feature based Urdu OCR using CNN and LSTM," International Journal of Advanced Computer Science and Applications, vol. 9, pp. 419-424, Jan 2018.
[12] V. A. Sindagi and V. M. Patel, "A survey of recent advances in CNN-based single image crowd counting and density estimation," Pattern Recognition Letters, vol. 107, pp. 3-16, May 1 2018.
[13] X. M. Deng, Y. D. Zhang, S. Yang, P. Tan, L. Chang, Y. Yuan, and H. A. Wang, "Joint Hand Detection and Rotation Estimation Using CNN," Ieee Transactions on Image Processing, vol. 27, pp. 1888-1900, Apr 2018.
[14] C. A and A. S, " Families of Alpha- Beta- and Gamma- Divergences: Flexible and Robust Measures of Similarities," Entropy, vol. 12, pp. 1532-1568, 2010.
[15] C. A, C. S, and A. S, "Generalized Alpha-Beta Divergences and Their Application to Robust Nonnegative Matrix Factorization," Entropy, vol. 13, pp. 134-170, 2011.
[16] W. W. Shi, Y. H. Gong, X. Y. Tao, and N. N. Zheng, "Training DCNN by Combining Max-Margin, Max-Correlation Objectives, and Correntropy Loss for Multilabel Image Classification," Ieee Transactions on Neural Networks and Learning Systems, vol. 29, pp. 2896-2908, Jul 2018.
[17] A. Sengupta, Y. Shim, K. Roy, "Proposal for an All-Spin Artificial Neural Network: Emulating Neural and Synaptic Functionalities Through Domain Wall Motion in Ferromagnets," Ieee Transactions on Biomedical Circuits and Systems, vol. 10, pp. 1152-1160, Dec 2016.
[18] P. Knag, J. K. Kim, T. Chen, and Z. Y. Zhang, "A Sparse Coding Neural Network ASIC With On-Chip Learning for Feature Extraction and Encoding," Ieee Journal of Solid-State Circuits, vol. 50, pp. 1070-1079, Apr 2015.
[19] S. Qian, H. Liu, C. Liu, S. Wu, and H. S. Wong, "Adaptive activation functions in convolutional neural networks," Neurocomputing, vol. 272, pp. 204-212, Jan 10 2018.
[20] A. Arcos-Garcia, J. A. Alvarez-Garcia, and L. M. Soria-Morillo, "Deep neural network for traffic sign recognition systems: An analysis of spatial transformers and stochastic optimisation methods," Neural Networks, vol. 99, pp. 158-165, Mar 2018.
[21] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, and A. Agrawal, "Deep Convolutional Neural Networks with transfer learning for computer vision-based data-driven pavement distress detection," Construction and Building Materials, vol. 157, pp. 322-330, Dec 30 2017.
[22] Research Institute of Yi Nationality Studies in Guizhou Province, “Southwest Yi Zhi”. The Nationalities Publishing House of Guizhou, 2015.

Open Science Index, Computer and Information Engineering Vol:13, No:9, 2019 publications.waset.org/10010760/pdf