IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 10, 2019, accepted July 20, 2019, date of publication July 24, 2019, date of current version August 9, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2930799

Recognition of Handwritten Chinese Characters Based on Concept Learning

LIANG XU[1], (Member, IEEE), YUXI WANG[1], XIUXI LI[2], AND MING PAN[3]

[1]School of Automation, Guangdong University of Technology, Guangzhou 510006, China
[2]School of Chemistry and Chemical Engineering, South China University of Technology, Guangzhou 510640, China
[3]Chemical Engineering and Technology, Sun Yat-sen University, Guangzhou 510275, China

Corresponding authors: Liang Xu (celiangxu@gdut.edu.cn) and Ming Pan (panm5@mail.sysu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 21376091, in part by the Guangdong Provincial Key Laboratory of Green Chemical Product Technology, and in part by the Zhujiang Talent Program High Talent Project of Guangdong Province under Grant 2017GC010614.

ABSTRACT Many deep-learning character recognition methods have been developed over the past few years. Chinese characters are widely used in many countries; however, the deep-learning-based Chinese character recognition methods are faced with various problems, such as a large amount of data required for training, numerous parameters, and a large consumption of computing resources. Concept learning is a hominine learning approach. Unlike existing deep-learning models, conceptual model learning can be realized by using as little as one sample. This paper is the first to propose a handwritten Chinese character recognition method based on concept learning. Different from the existing image representation-based character recognition methods, the proposed method builds a meta stroke library with prior knowledge, and then, presents a Chinese character conceptual model based on stroke relationship learning using a character stroke extraction method and Bayesian program learning. During character recognition, Monte Carlo Markov chain sampling is utilized to obtain the character generation model for each character conceptual. This generation model can calculate the probability of the target and training characters being the same classification, and thereby determines the classification of the target character. The experimental results indicate that, with the proposed method, the conceptual model of each character can be built for character classification prediction using as few as one character sample. Our approach obtains better performance than the state-of-the-art methods on ICDAR-2013 competition dataset.

INDEX TERMS Character conceptual model, character recognition, concept learning, stroke extraction.

I. INTRODUCTION

Character recognition is a popular subject in pattern recognition. Over the past few decades, many researchers have focused on Chinese character recognition [1]–[6]. However, handwritten Chinese character recognition (HCCR) is still challenging for various reasons, including variations in handwriting between different individuals, character similarities, and the large number of characters that must be classified. Handwritten character recognition is complicated and usually contains the following three steps: text line detection and division, character segmentation and extraction from the text line, and character recognition. To address HCCR problems, a novel HCCR method is proposed herein based on concept learning.

A number of methods have been applied to address character recognition issues over the past few decades [1]. In general, there are two key steps for character recognition: (a) extracting features from the character image and (b) analysis and recognition of the character sequence of a given word component output by the classifier. In both strategies, different computer vision techniques are usually used to process and detect character images. Alternatively, the column pixels are directly used as the features.

Principal component analysis (PCA) is used to extract the character components in fixed-size images [7], [8]. In a study by Bideault *et al.* [9], a histogram of oriented gradients (HOG) was proposed for character feature extraction. Graves and Schmidhuber [10] used the feature extraction of

The associate editor coordinating the review of this manuscript and approving it for publication was Sudhakar Babu Thanikanti.

column pixels for the character image, such as the mean, centroid, conversion, and fusion. Pixel features were directly used as the inputs of the character models [11]. To recognize and analyze handwritten characters, the words must be converted into character sequences. The hidden Markov model (HMM) [8] and connectionist temporal classification (CTC) [12] are the most commonly used methods. The features contained in character images are extracted as the input of the classifier, which is a posteriori estimate based on an artificial neural network (ANN) and Gaussian mixture model (GMM) [13]–[15].

Deep learning has become the most influential model in computer vision and pattern recognition. Many new achievements in character recognition have been made with the application of deep learning [16], [17]. A cellular neural network (CNN) was used by Jaderberg *et al.* [16] for the optical character recognition (OCR) of natural scene images. A deep CNN was applied to fixed-dictionary character recognition by Poznanski and Wolf [17], but the character recognition performance of this method decreased with increasing dictionary size. CNNs have also been successfully applied to Chinese character recognition and exhibited superior performance [18]–[20]. In the most traditional CNN-based Chinese character recognition method, the image of a given character (including handwritten features) is taken as the input of the CNN, and character recognition is realized by means of character image classification. An improved method that enhanced the recognition performance of the CNN which included handwriting changes in the feature image using a spatio-temporal feature composition strategy for characters was developed by Yang *et al.* [19]. Although the CNN model is superior to traditional methods in character recognition, there are also deficiencies in this image classification-based model: 1) sufficient knowledge of a certain field is required to extract complex image features; 2) large amounts of training data are needed to enhance the model accuracy; and 3) the handwritten strokes of characters must be converted to images.

The recurrent neural network (RNN), especially long short-term memory (LSTM), is an important model in character recognition. With this method, the sequence structures of handwritten characters are processed without knowledge of the field. Various strategies have been used by researchers for character feature extraction and character analysis, and multiple RNN-based character recognition models have been proposed [8], [21]–[24]. A RNN-HMM hybrid model was developed by Doetsch *et al.* [8] for handwritten English character recognition. In this method, the HMM was used for data training, while labeled data frames were taken as the input of the RNN. A RNN-based improved model called a multidimensional recurrent neural network (MDRNN) was proposed by Graves and Fernández [21], which extended the two-dimensional property of handwritten images. To address in-air handwritten Chinese character recognition, Ren *et al.* [22] developed an RNN-based end-to-end character recognizer, in which the sequence of handwritten points served as the input, and feature extraction was not required. In a study by Xie *et al.* [23], the handwriting extraction method was improved by converting pen-tip trajectories into informative signature feature maps. In addition, a multi-spatial-context fully convolutional recurrent network (MC-FCRN) was developed to generate a handwriting prediction sequence using the multi-spatial-context in characteristic mapping. RNNs have not only been used for Chinese character recognition. A RNN-based Chinese character generation model that had an end-to-end structure integrating LSTM and a gated recurrent unit (GRU) without knowledge of the particular field was developed by Zhang *et al.* [24].

As discussed above, excellent achievements in Chinese character recognition have been made using deep-learning methods. However, whether they are CNN-based, RNN-based, or improved methods, these methods are still deficient. These methods require large amounts of training data and consume significant amounts of computing resources to achieve favorable performances. Model performance can be enhanced by means of data enhancement [25], however a smaller amount of training data results in reduced model accuracy.

Humans are able to learn new concepts and reasoning from a small sample, which is also known as "one-shot learning." In a study by Ghahramani [26], a conceptual model framework describing the methods of learning from data using experiential knowledge was proposed. To make the model training and performance closer to the human learning process, a conceptual learning model was put forward by Lake *et al.* [27], which was a new hominine learning method that integrated probability distributions with the cognition and learning of the potential rules of the human brain. This was a conceptual model built using the framework of Bayesian program learning (BPL) [27], which provides the machine with the ability to perform one-shot learning and generalization. A computational model where images were divided into shapes and colors was developed by Ahmed and Bikmal [28]. By calculating the possibility of each category using a Bayesian rule, this model required only one sample to learn from.

Herein, this is the first work on using the concept learning for handwritten Chinese character recognition. In this framework, character concept learning that simulates human learning begins from basic character strokes; thus, character learning is no longer simple end-to-end learning. Instead, the method aims to acquire a conceptual model corresponding to each different character. The model utilizes the strokes composing the character, including the stroke sequence and stroke combination. The contributions of this study are summarized as follows: 1) A concept-learning-based handwritten Chinese character recognition method framework is proposed. The proposed method differs from existing ones, such as CNNs and RNNs. This is the first method, to the best of our knowledge, to focus on handwritten Chinese character recognition using the concept learning. Unlike existing methods, e.g., CNNs and RNNs, this method does

TABLE 1. **Variables and descriptions.**

Variable	The meaning of representation
S_{ij}	Basic character strokes of meta stroke library
x_{ij}	Control point of basic character stroke S_{ij}
y_{ij}	Scale of basic character stroke S_{ij}
z_{ij}	Index value of the basic stroke S_{ij}
S_i	The i-th stroke of character sample
R	Relations between various strokes
ξ_i	Connection relationship between the i-th stroke and a previous stroke
L_i	Starting point of the i-th stroke
τ_i	Specific connection position between the i-th stroke and a previous stroke
I^C	Training sample
I^T	Target sample
ϕ^k	Conceptual model
φ^{Tk}	Fitting model
A^{Tk}	Global transformation parameter
σ^{Tk}	Gaussian nuclear standard deviation

not require a large sample size for training. Instead, model training and building is realized with as few as one sample. 2) A stroke-width- and directional-distance-histogram-based Chinese character stroke extraction method are proposed to extract effective strokes by merging redundant feature point in stroke intersection regions.

The rest of this paper is organized as follows. Table 1 introduces the variables and meaning descriptions used in the paper. Section II presents the method for character stroke extraction. Section III describes the construction of the character concept model and recognition method. The proposed method is evaluated with experimental results, and the implementation detail and analysis of the method are also discussed in Section IV. Finally, the outcome of this work is summarized in Section V.

II. CHARACTER STROKE EXTRACTION METHOD

Various stroke representation methods have been proposed in previous works [24], [29], [30], but are rarely concerned with the processing of redundant strokes. Since the character conceptual model is based on the character strokes and the stroke relations, character stroke extraction plays an important role in generating the entire character conceptual model. During stroke extraction procedure, redundant strokes may occur in stroke intersection regions, which will degrade the performance of the model. Thus, we propose the novel stroke extraction method based on the stroke widths and directional distance histograms for the Chinese characters.

For the sake of description, the following terms are defined. As shown in Fig. 1, the red points are feature points which can be marked as end points and intersection points. The end points are the feature points at the end of the stroke skeleton, such as 1, 2, 3 and 4. While the intersections points occur when several red points merge together, for instance, feature points 5, 6, 7, 8, and 9 are merged into an intersection point 9. Thus, the end points and intersection points are principal points in the extraction of character strokes. The character skeleton can be extracted from the character image after binarization. Based on the positions of the skeleton pixel

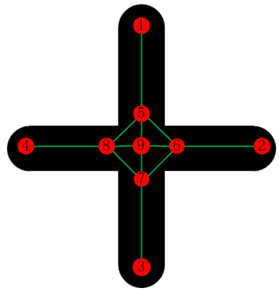


FIGURE 1. **Diagram of feature points.**

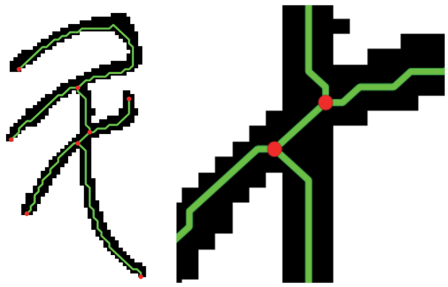


FIGURE 2. **Distortion in stroke crossing region.**

points and the principal points of skeleton strokes, the stroke linked by the principal points can be extracted.

As shown in Fig. 2, using XiaoZhuan characters as an example, stroke skeleton distortion will occur at stroke intersections that occur in redundant strokes between these feature points or, in some extreme cases, unreasonable redundant strokes. Redundant strokes can impact the accuracy of the character conceptual model and lead to recognition errors.

To address this issue, a stroke-width and directional-distance-histogram-based Chinese character stroke extraction method is depicted in Fig. 3.

A. EXTRACTION OF FEATURE POINT

In the feature point extraction, the input Chinese character image is binarized, followed by single-pixel-wide character skeleton extraction using the thinning method developed by Lam *et al.* [29]; this is shown by the green lines in the character area of Fig. 2. Next, the method proposed by Liu *et al.* [30] is used to obtain the character skeleton feature points by calculating the eight neighborhood pixel values of the skeleton pixel points. The neighborhood of skeleton pixel point P is shown in Fig. 4. Supposing that the pixel value of each skeleton pixel point is 1 in the binarized image and 0 in the other areas, the skeleton points complying with (1) are considered to be feature points.

$$\begin{cases} N(P) = 1 \\ N(P) \geq 4 \\ N_c(P) \geq 3, \end{cases} \tag{1}$$

where $N(P)$ means there is only one pixel point with a value of 1 in the neighborhood of the stroke skeleton pixel point P. Such skeleton points are used as end points. $N_c(P)$ is calculated using (2), where $P = P_0$. If $N_c(P) \geq 3$ or $N(P) \geq 4$,

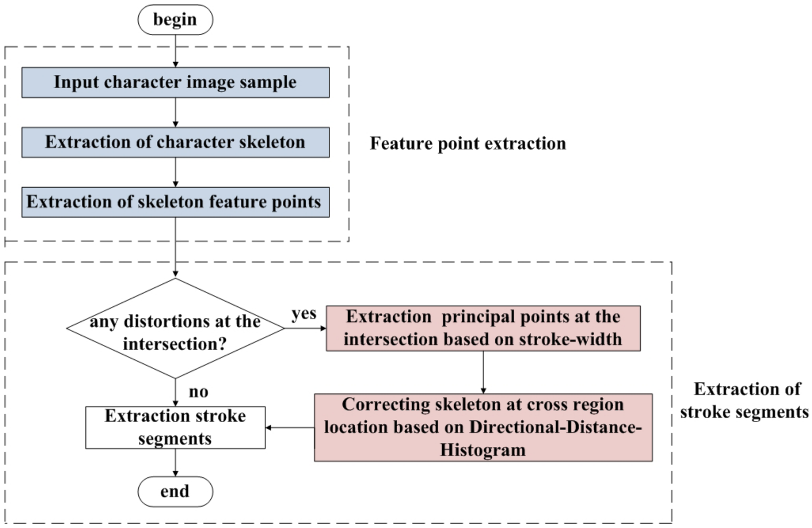


FIGURE 3. Flowchart of Chinese character stroke extraction method.

P_7	P_0	P_1
P_6	P	P_2
P_5	P_4	P_3

FIGURE 4. Eight neighbors of pixel point *P*.

and point P is considered a feature point.

$$N_c(P) = \frac{1}{2}\sum_{0}^{8}|P_{i+1} - P_i|. \tag{2}$$

As shown in Figs. 1 and 2, a stroke-width-based intersection point acquisition (SWIP) method is proposed to address the problem that there are multiple feature points at one stroke intersection (only one point is actually required) when the aforementioned feature point extraction method is used.

B. STROKE-WIDTH-BASED INTERSECTION POINT ACQUISITION

The stroke-width-based intersection point acquisition method, which is described in Algorithm 1, focuses on the calculation of the Euler distance (d) between neighboring intersection points. When d is smaller than threshold T, the intersection points are merged using the maximum circle method [31]. As shown in Fig. 5, the stroke width is the transverse width of the stroke segment in which each end point is located based on the angle of the stroke skeleton. The number of character end points (n) is obtained using (1). To avoid failure in calculating stroke width due to excessively short strokes corresponding to the end point and enhance the calculation accuracy of the angle of the skeleton point, the skeleton segment spline where the skeleton end point is located was smoothed. The skeleton segments where the number of pixel points is smaller than e were interpolated to e pixel points (empirical value $e = 10$). Skeleton segment splines are shown as yellow curves in Fig. 5.

Algorithm 1 Stroke-Width-Based Intersection Point Acquisition

Input:(x_i, y_i) // feature point in intersection region
Output:(x_l, y_l) // intersection point after merging
Begin
count number of end point n as let $N(P) = 1$
for $i = 1$ to n **do**
  $\overrightarrow{P_{i1}P_{ie}}, \overrightarrow{P_{ie}P_{i2}} \leftarrow$ linear interpolate where end point P_{ie} is located
  calculate angle θ_i between $\overrightarrow{P_{i1}P_{ie}}$ and $\overrightarrow{P_{i1}P_{i2}}$
  calculate stroke width w_i as: $w_i \leftarrow \frac{1}{2}\theta_i$
end for
calculate stroke width w of sample character as: $w = \frac{1}{n}\sum_{i=1}^{n} w_i$
if $d < \sqrt{2 \times w^2}$ **then** // d: the Euler distance between intersection points
  acquire interaction point (x_l, y_l) using the maximum circle method
end if
End

The blue spline points that are distances of $e/2$ and e away from the red point $P_e(x_e, y_e)$ in the spline curve are selected and denoted as $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$, respectively. As shown by the white arrow in Fig. 5, the angle of point $P_1(x_1, y_1)$ is the angle (θ) between vectors $\overrightarrow{P_1P_e} = (x_e - x_1, y_e - y_1)$ and $\overrightarrow{P_1P_2} = (x_2 - x_1, y_2 - y_1)$. Angle θ is obtained using (3):

$$\theta = \arccos\left(\frac{\overrightarrow{P_1P_e} \cdot \overrightarrow{P_1P_2}}{|\overrightarrow{P_1P_e}||\overrightarrow{P_1P_2}|}\right). \tag{3}$$

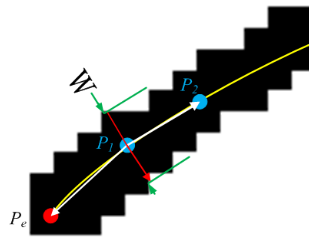


FIGURE 5. Diagram of stroke width calculation.

As shown in Fig. 5, there is a red line segment that is the angular bisector of θ and is between the edges of the stroke. The length (W) of this line segment is taken as the width of the stroke in which the current end point is located. As shown in (4), the stroke width (W) of the sample character can be obtained by taking the average value of all the n end points on the skeleton followed by the same operation.

$$T_{threshold} = \sqrt{2 \times w^2}, w = \frac{w_1 + w_2 + \cdots + w_n}{n}. \tag{4}$$

If the Euler distance (d) between feature points is smaller than threshold $T_{threshold}$ in (4), they are merged into one intersection point using the maximum circle method.

It is necessary to correct the distorted character skeletons after the feature points are merged. If all the feature points are principal points (end points and intersection points) after the feature point extraction, the skeletons between the principal points can be directly acquired as strokes. However, the maximum circles of the feature points in the same stroke crossing region will not intersect if these feature points are too distant from each other. In this case, there will still be multiple unmerged feature points in the stroke crossing region using maximum circle method, which will result in distortion at the stroke intersection. To address this problem, a directional-distance-histogram-based crossing region location method is proposed in this study.

C. DIRECTIONAL-DISTANCE-HISTOGRAM BASED CROSSING REGION LOCATION

As shown in Fig. 6, the horizontal and vertical strokes cross in the white region. P_f and D_{Pf} are defined as the stroke skeleton intersection point (the red point at the center) and the distance between intersection point P_f and the edge point of the stroke along the ρ-th direction, respectively. $\rho = 3m$, where m is an empirical integer between 0 and 120. The yellow intersection point in Fig. 7 is merged by the previous two red feature points using the SWIP method. Based on the analysis above, the histogram of direction-distance (HODD) of this intersection point can be obtained by taking ρ and D_{Pf} as the horizontal and vertical axes, respectively, as shown in Fig. 8.

As shown in Figs. 6 and 7, the four intersection edge points used to locate the stroke crossing region correspond to the positions of the four troughs of the histogram in Fig. 8.

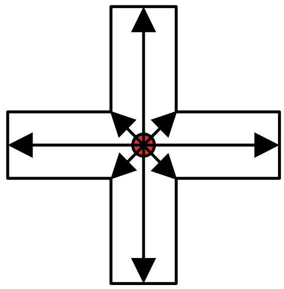


FIGURE 6. Directional distance between intersection points.

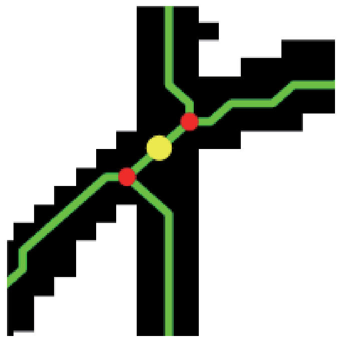


FIGURE 7. Intersection point merging.

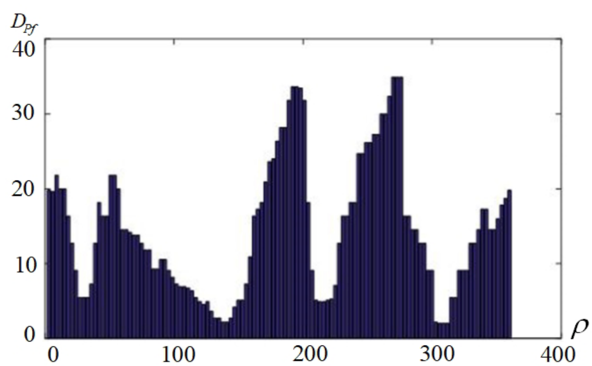


FIGURE 8. Histogram of direction distance (HODD).

The positions of the four troughs are found based on the HODD of the intersection point. The equal neighboring columns of the HODD are deleted first, followed by the identification of the directions corresponding to the positions of the four troughs on the HODD using (5). t is the serial number of the position of a trough on the HODD. The segmentation point can be located by combining t with the corresponding direction ρ and direction distance D_{Pf}.

$$t = \left\{ t | D_{Pf}(t - m) > D_{Pf} \right\}, \quad m \in [-2, 2] \,\&m \neq 0. \tag{5}$$

The region surrounded by the stroke crossing region segmentation points is represented by the positions of the four troughs in Fig. 8. This region is the stroke crossing region. The stroke skeletons within this region are deleted. Linear interpolation from the edge of this region to the intersection points is conducted to correct the stroke skeleton distortion in the crossing region. Algorithm 2 illustrates the directional-distance-histogram-based crossing region location (DDHCR) method.

Algorithm 2 Directional-Distance-Histogram-Based Crossing Region Location

Input: $\rho \in N$, $1 \leq \rho \leq 360$, D_{Pf}
Output: segmentation points t
Begin

HODD $\leftarrow \rho, D_{Pf}$ // fetch histogram of directional-distance in cross point
for each column $i = 1$ to ρ **do**
  if $D_{Pf}(i) == D_{Pf}(i+1)$ **then**
    delete $D_{Pf}(i+1)$ // deleting equal and adjacent column of HODD
  end if
  if $D_{Pf}(i-m) > D_{Pf}(i)$ && $(m \in [-2, 2]$ && $m! = 0)$ **then**
    output $t = i$ // t is segmentation point
  end if
end for
End

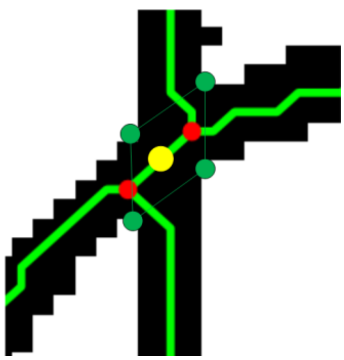


FIGURE 9. Stroke crossing region.

The segmentation points in stroke crossing region correspond to the positions of the four troughs in Fig. 8 are shown as the four green points in Fig. 9. They are connected end to end to circle into the stroke crossing region. The stroke skeletons within this region are deleted. Linear interpolation from the edge of this region to the intersection points is conducted to correct the stroke skeleton distortion in the crossing region. Fig. 10 shows the comparison before and after the correction. The feature points at the stroke intersection are merged without influencing the end points. The skeletons between neighboring principal points serve as the strokes to be extracted, which can be used for follow-up character conceptual modeling and recognition.

III. CHARACTER CONCEPTUAL MODEL BUILDING AND RECOGNITION METHOD

In the concept-learning-based handwritten Chinese character recognition method, the strokes constructing a certain character are extracted using the aforementioned character stroke

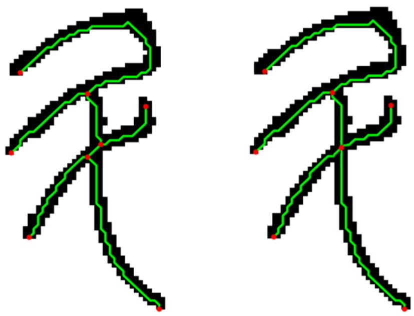


FIGURE 10. Comparison of strokes in the crossing region before and after distortion correction.

extraction method. Next, these strokes undergo stochastic traversal to generate different handwriting forms of the character. These different forms are subsequently modeled using BPL. The models with the highest probability are taken as the conceptual model of the Chinese character. During character recognition, the character conceptual model of the training sample is fitted to the target sample. Finally, the classification of the target character is predicted based on the fitting probability. The proposed concept-learning-based HCCR method contains two major steps. One step is character conceptual modeling in which each Chinese character is decomposed into sub-strokes using the character stroke extraction method. This is followed by the conceptual modeling of the training sample based on the stroke and stroke connection relationship. The other step is character classification prediction in which the classification of the training character sample that fits the target sample most closely (probability value) is determined to be the target classification to be predicted by building the target character generation model, as shown in Fig. 11.

A. CONSTRUCTION OF CHARACTER CONCEPTUAL MODEL

To build the character conceptual model, a reasonable decomposition of the character sample is required until the basic strokes (elements) are obtained. Next, character conceptual models are built based on the strokes, stroke combination sequence, and stroke connection relationships, described as follows:

Step 1: A character meta stroke library is constructed, which includes the basic character strokes $s_{ij} = \{z_{ij}, x_{ij}, y_{ij}\}$, stroke sequence, and the relations between various strokes (R). Control point x_{ij} follows a normal distribution $P(x_{ij}|z_{ij})$, and scale y_{ij} follows a gamma distribution $P(y_{ij}|z_{ij})$, where z_{ij} is the index value of the basic stroke.

Step 2: The strokes and handwriting styles of training character sample I^C are extracted. The character image skeleton is thinned first to ensure there is only one pixel point on each point of the strokes. Next, the feature points of the thinned image are calculated. The SWIP and DDHCR are used to handle distorted and redundant strokes, thereby yielding the strokes between principal points. A stochastic traversal of strokes is carried out to generate multiple traversal modes

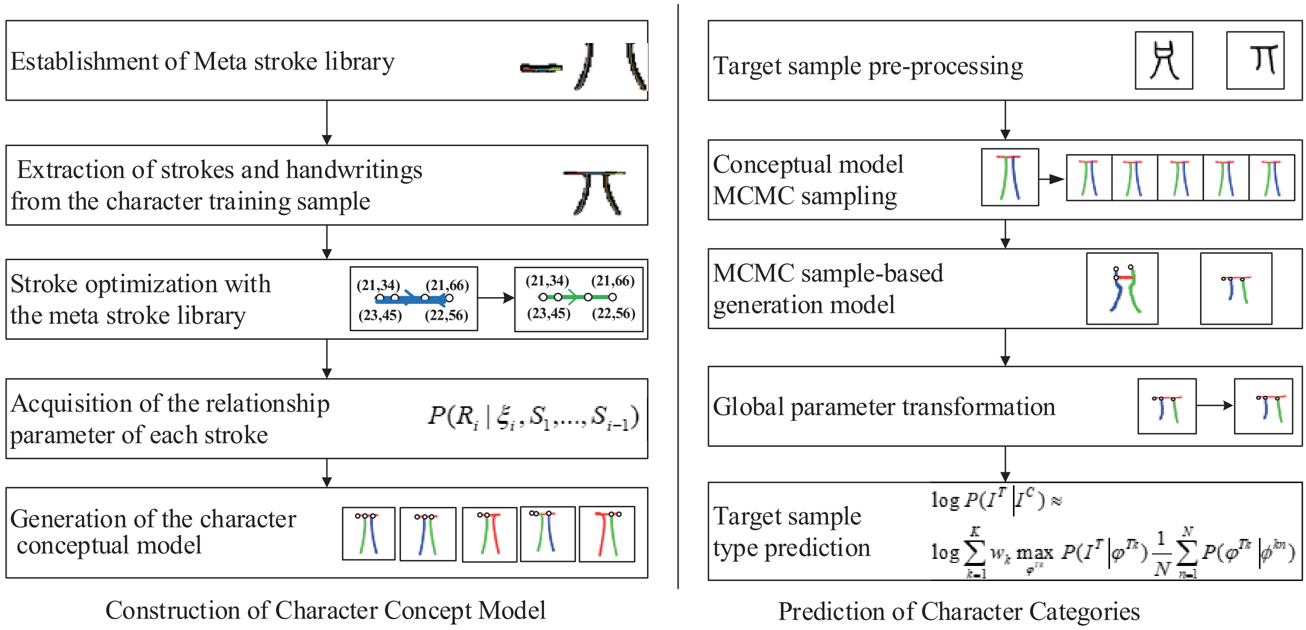


FIGURE 11. Framework of the concept-learning-based Chinese character recognition method.

and construct strokes using basic strokes. With multiple traversals, the stroke S_i and handwriting forms of the character sample can be obtained. The more times the traversal is conducted, the higher the accuracy of the conceptual model.

Step 3: Breakpoint sampling of the stroke S_i is conducted using a greedy algorithm [32]. If the error among the splines of the stroke control points in the traversal process before the fitting and that after the fitting are larger than the threshold, the stroke S_i is broken into sub-strokes $s_{ij} = \{z_{ij}, x_{ij}, y_{ij}\}$. Considering that the different stroke control points follow a normal distribution and the scales follow a gamma distribution, the probabilities of the broken sub-strokes in these two distributions are determined. The stroke in the meta stroke library corresponding to the maximum probability is the matched stroke (represented by its index value z_{ij}). The spline trajectory of the current stroke control points is fit with the trajectory obtained by the previous traversal. If the fitting error is smaller than the threshold, the point-breaking method is correct; otherwise, breakpoint re-sampling is needed until the sub-strokes match with the strokes in the stroke library. All the n_i sub-strokes of stroke S_i can be obtained using this method. The composition process of n_i sub-strokes is obtained using the first-order transition probabilities of various strokes in the meta stroke library, expressed as $P(z_i) = P(z_{i1})\Pi_{j=2}^{n_i} P(z_{ij}|z_{i(j-1)})$. Thus, the joint probability distribution of S_i can be obtained as follows: $P(S_i) = P(z_i)\Pi_{j=1}^{n_i} P(x_{ij}|z_{ij})P(y_{ij}|z_{ij})$.

Step 4: The parameters of each stroke relation $P(R_i|\xi_i, S_1, \ldots, S_{i-1})$, including the connection relationship ξ_i (along, start, end, and independent), the stroke starting position L_i, and the position (τ_i) where strokes are connected, are obtained based on the relevant stroke in the meta stroke library, after the stroke S_i is determined.

Step 5: With the combination relationship set to condition R, the conceptual model ϕ of the training character sample is generated by combining k strokes. The joint probability distribution of the conceptual model ϕ is calculated as follows: $P(k)\Pi_{i=1}^{k} P(S_i)P(R_i|S_1, \ldots, S_{i-1})$. k models with the highest probabilities, namely $\phi^1, \ldots, \phi^k$, are selected as the conceptual models of this character.

Based on the steps above, the conceptual models corresponding to different characters can be built. Essentially, a character conceptual model is a model hierarchically connected by the Bayesian network based on parameters such as the sub-strokes, stroke sequence, and stroke connection relations. In other words, one character corresponds to a unique Bayesian network. When a new character is recognized and classified, the pre-built character Bayesian network is fit to the new character. Since there is a different Bayesian network for each character, the fitting of the target sample will also appear with different joint probability distributions. The character classification with the highest probability is taken as the result of target sample classification. In a one-shot learning task, however, the conceptual model of each classification of Chinese characters can be built using a training sample containing only one Chinese character. The establishment of Chinese character conceptual model is depicted in Algorithm 3.

B. PREDICTION OF CHARACTER CLASSIFICATION

To predict the classification of the target character sample, the generation model can be obtained by fitting the pre-built character conceptual model to the target character.

Algorithm 3 Establishment of Chinese Character Conceptual Model

Input: training sample I^C
Output: character conceptual model ϕ
Begin
1: count number of sample stroke k as: $k \leftarrow P(k)$
2: **for** $i = 1$ to k **do**
3: count number of sample sub-stroke n_i as: $n_i \leftarrow P(n_i|k)$
4: calculate the joint probability distribution of S_i as: $S_i \leftarrow P(S_i) = P(z_i)\prod_{j=1}^{n_i} P(x_{ij}|z_{ij})P(y_{ij}|z_{ij})$
5: acquire stroke relation R as: $R_i \leftarrow P(R_i|\xi_i, S_1, \ldots, S_{i-1})$
6: **end for**
7: build concept model ϕ as: $\phi \leftarrow P(k)\prod_{i=1}^{k} P(S_i)P(R_i|S_1, \ldots, S_{i-1})$
End

Next, a Bayesian program learning is used to calculate the probability (training sample I^C and target sample I^T to the same classification). The character classification with the highest probability is the target classification.

The classification of target sample I^T is predicted based on k candidate models built using training sample I^C. First, the $K \times N$ Monte Carlo Markov chain (MCMC) [27] samples $(\phi^{11}, \ldots, \phi^{1n}, \ldots, \phi^{k1}, \ldots, \phi^{kn})$ of each candidate conceptual model in the training sample I^C are calculated. Next, based on the strokes of the target sample I^T, the MCMC samples of the candidate conceptual models are generated and fitted to the strokes of target sample I^T after parameter adjustment, thereby generating a fitting model φ^{Tk}. In this way, the task of predicting the classification of sample I^T is converted to one of finding the classification of sample I^C that fits to target sample I^T with the highest probability, i.e., making the value of $P(I^T|I^C)$ the largest.

Conceptual models are fitted to target sample I^T, the probability models are $\varphi^{T1} \ldots \varphi^{Tk}$. The probability $P(I^T|I^C)$ represents the possibility of I^T and I^C being the same classification. The probability that each classification $(I^1, \ldots, I^c)$ is of the same classification as target sample I^T is calculated using (6). The character classification with the highest probability in I^C is the classification of target sample I^T. Equation (6) is as follows:

$$\log P(I^T \middle| I^C) \approx \log \sum_{k=1}^{K} w_k \times \left(\max_{\varphi^{Tk}} P(I^T \middle| \varphi^{Tk}) \frac{1}{N} \sum_{n=1}^{N} P(\varphi^{Tk} \middle| \phi^{kn}) \right). \quad (6)$$

Specifically, φ^{Tk} is obtained by fitting conceptual model ϕ^k to target sample I^T, as shown in (7):

$$\varphi^{Tk} = \max_{\varphi^{Tk}} P(I^T|\varphi^{Tk}) \frac{1}{N} \sum_{n=1}^{N} P(\varphi^{Tk}|\phi^{kn}). \quad (7)$$

The process of character prediction classification is as follows:

Step 1: The $K \times N$ MCMC samples of k candidate conceptual models ϕ^k of training sample I^C are calculated.

Step 2: For each MCMC sample ϕ^{kn} of conceptual model ϕ^k, the starting point and shape of each stroke are traversed while the stroke combination classifications R remain unchanged in target sample I^T. The MCMC samples are fitted to target sample I^T by sampling relevant parameters and generating fitting models $\varphi^{T1}, \ldots, \varphi^{Tk}$.

Step 3: The sampling character global transformation parameters $P(A^{Tk})$ are used to make the strokes of fitting model φ^{Tk} fall within the stroke range of the target sample as much as possible.

Step 4: The probability of each classification of $I^1, \ldots, I^c$ and target sample I^T being the same classification is calculated using (6). The character classification with the highest probability is the target classification.

IV. EXPERIMENT

A. DATA DESCRIPTION

Four hundred classifications of XiaoZhuan characters (http://www.ziti123.com/ziti/zhuanshu/) were selected from the standard XiaoZhuan font database as experimental subjects. First, they were downloaded from the XiaoZhuan file in TTF format and the XiaoZhuan characters were exported to the standard character library with BabelMap software. As shown in Fig. 12, they were stochastically divided into 20 groups. In addition, the handwritten samples provided by 22 participants were collected. In each group, the XiaoZhuan characters in the first row were standard fonts, and those in the second to seventh rows were the corresponding handwritten characters. Twenty standard XiaoZhuan characters served as the training sample, while the corresponding 20 handwritten XiaoZhuan characters were the target sample. The testing and operating environment of this experiment was as follows: processor: Inter Core (TM) i7-7500U CPU running at 2.70 GHz; RAM: 8.00 GB; operating system: Win 10, 64 bit; software, MatLabR2015b.

First, XiaoZhuan samples received pre-processing to ensure all the characters were of the same size i.e. 105×105. This was followed by sample image binarization, with the pixel value of the character area set to 1 and that of the blank area set to 0. In the experiment, different handwriting samples in each row belonged to the same character concept. The conceptual model of XiaoZhuan was built by learning the standard XiaoZhuan character. The conceptual learning result was not a simple XiaoZhuan character recognition. Instead, it aimed to obtain a model containing the strokes composing the XiaoZhuan character samples and stroke connection relations. Thus, it was more a model for learning the writing processes of XiaoZhuan characters, thereby predicting target sample classifications with fewer samples (e.g., one sample) by building the Chinese character conceptual models.

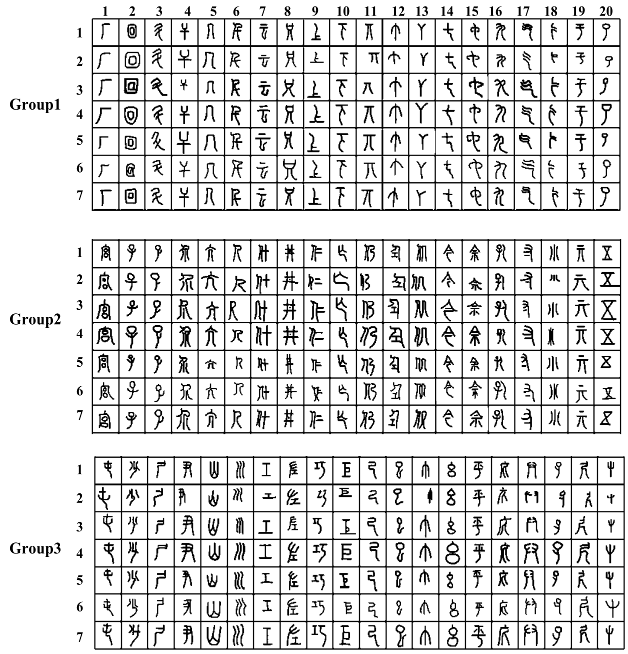
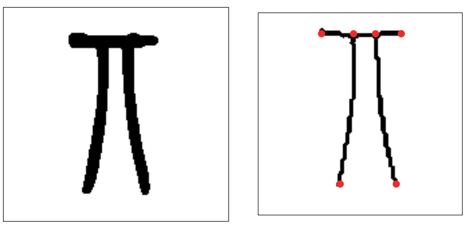


FIGURE 12. Part of XiaoZhuan samples.



(a) (b)

FIGURE 13. Standard XiaoZhuan character images. (a) Standard XiaoZhuan character training sample. (b) Character image after thinning.

B. RESULTS AND ANALYSIS

1) STROKE EXTRACTION

a: PRINCIPAL POINT EXTRACTION

Training sample I^C contains the standard XiaoZhuan character images shown in Fig. 13(a). The pre-treated XiaoZhuan character images were thinned to ensure that there was only one pixel on each point in the image, as shown in Fig. 13(b). Seven feature points of this training image, including four end points ((21, 34), (21, 66), (88, 38), (88, 63)) and the other three feature points ((23, 45), (22, 56), and (23, 56)) (they represent the positions of pixels in the 105 × 105 matrix), were obtained using the proposed method. Considering that there were redundant feature points, the SWIP and maximum circle method could be used to remove feature point (23, 56); therefore, six principal points of the training sample were obtained (Fig. 14).

b: STROKE SEGMENT EXTRACTION

As shown in Fig. 14, with the redundant feature points removed, the DDHCR location was determined to delete the stroke skeleton between feature point (22, 56) and redundant point (23, 56) so that it could be reasonably connected to the intersection point. This would facilitate accurate stroke extraction and final acquisition of the stroke skeleton. As shown in Fig. 15, five stroke segments (marked by different colors) were extracted.

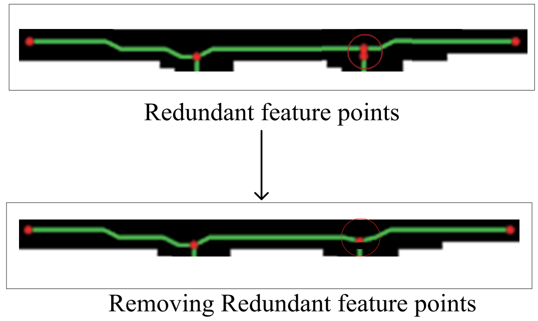


FIGURE 14. Redundant feature points.

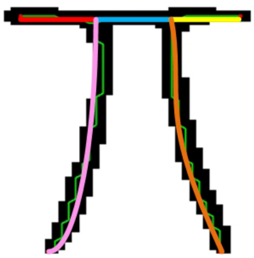


FIGURE 15. Stroke segments extracted from the character in the training sample.

2) CONSTRUCTION OF CONCEPTUAL MODEL FOR TRAINING SAMPLE

For the purpose of handwritten character classification prediction, the character conceptual model was built by learning sample data of multiple specific characters. The concrete concept of this character could be generated by building these conceptual models. In other words, character recognition based on a small sample size could be realized by learning the handwriting form of this character.

a: META STROKE LIBRARY

A stroke could be represented by five control points and the scale parameter. In the meta stroke library, different basic strokes corresponded to different directional control points x_{ij} and scales y_{ij} (control points and scales were subject to normal and gamma distributions, respectively). This process was demonstrated by the first stroke of training character as shown in Fig. 13(a). The index value of this stroke was seven in the 1212 basic strokes of the meta stroke library. All 10 parameters from the coordinates $((x_i, x_{i+5})(i = 1, \ldots, 5))$ of the five control points of the seventh basic stroke in the meta stroke library and stroke scales y complied with normal and gamma distributions, respectively. Their corresponding parameters are shown in Fig. 16.

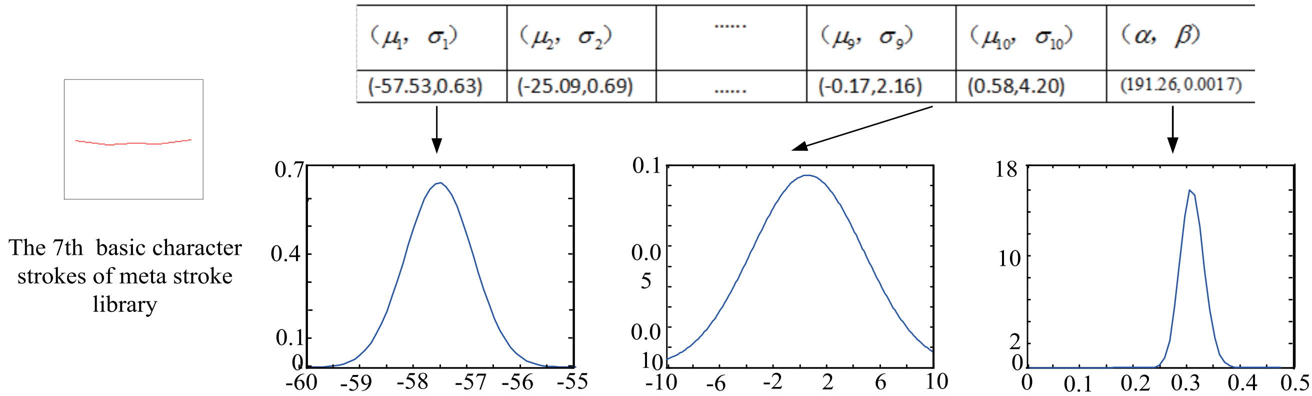


FIGURE 16. Parameter distribution of the seventh basic stroke in the meta stroke library.

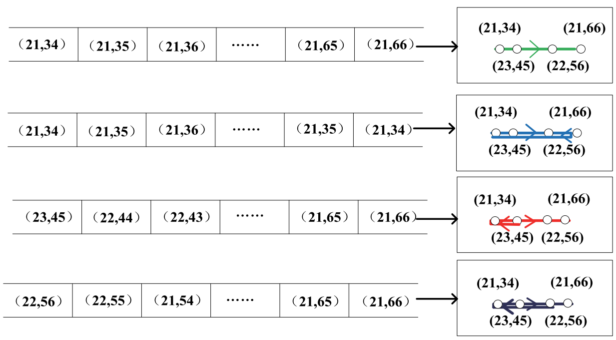


FIGURE 17. Four different handwritings of the first stroke.

b: EXTRACTION OF STROKES AND HANDWRITINGS FROM THE CHARACTER TRAINING SAMPLE

Stochastic stroke traversal was carried out after extracting character strokes, and therefore, multiple traversal modes as well as the strokes that were made up of the corresponding sub-strokes of this character were obtained. First, a certain principal point (six in total) was stochastically selected as the starting point of the character trajectory traversal. It was supposed that principal point (21, 34) was the starting point of the traversal. The path between neighboring principal points was obtained. The end point of this path was taken as the new starting point of the traversal. If this new starting point was an end point, however, it suggested that this stroke was complete, and the traversal should be continued by stochastically generating a starting point. Otherwise, this stroke did not end, and the traversal continued with this new starting point until it reached the end point. Meanwhile, different writing sequences and stroke directions were optimized. There were 57 strokes and 75 handwriting forms of this character obtained with multiple traversals. Different handwriting forms might be traversed with the same stroke, as shown in Fig. 17 (the coordinate points represent the positions of stroke pixels in the 105×105 matrix).

c: STROKE OPTIMIZATION WITH META STROKE LIBRARY

A greedy algorithm was used to sample each stroke breakpoint, thereby determining whether it was necessary to break the stroke into sub-strokes until they matched with the strokes in the meta stroke library. The first stroke in the previous step is used as an example. To match the stroke specifications in the meta stroke library, stroke smoothing and coordinate transformation were conducted, thereby yielding the two-dimensional coordinates of the five control points ((−57.48,0.07), (−24.84,0.52), (0.00, −43.00), (24.84,0.13), and (57.48,1.34)) and the stroke scale parameter (the distance of the stroke in the Y direction/image size=32/105=0.3048). Subsequently, the probability of each stroke was calculated based on the five control points and scale parameter of this sub-stroke and the probability distribution of the stroke in the meta library. The stroke with the highest probability was selected. Thus, it could be determined that its index value z_{ij} was 7. The distribution that parameters of the seventh stroke were subject to is shown in Fig. 16. Finally, the spline trajectories generated by the five control points of this sub-stroke were fitted with original trajectories. The fitting error was the mode length of the coordinate of the point corresponding to the two trajectories. Breakpoint re-sampling was not required since the max fitting error was 0.46, which was smaller than the empirical threshold of 3.

All 57 strokes were optimized using the above method, while each stroke was decomposed into the sub-strokes which must be matched with the basic strokes in the meta stroke library.

d: ACQUISITION OF RELATIONSHIP PARAMETER OF EACH STROKE

When each stroke was determined, the relationship parameters (R) between various strokes could be obtained based on the meta stroke library. As shown in Fig. 18, the relationship parameters (R) between various strokes, including the connection relations between various strokes, the stroke starting positions, and the specific positions where strokes were connected, could be obtained. The connection relations were as follows: the second and third strokes were connected with a sub-stroke of the first stroke in the middle, denoted as $\xi_{1,2} = Along$ and $\xi_{1,3} = Along$, respectively. The sub-strokes of the second and the third strokes were independent,

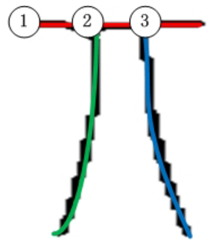


FIGURE 18. Basic character strokes and stroke sequence.

TABLE 2. Joint probability distribution ($k = 5$) of candidate conceptual model.

	The conceptual model with the highest probability				
	ϕ^1	ϕ^2	ϕ^3	ϕ^4	ϕ^5
Corresponding probability	−547.2	−560.6	−561.1	−562.2	−566.9

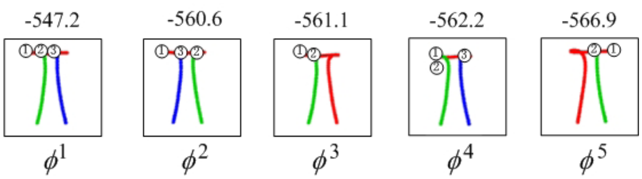


FIGURE 19. Five candidate conceptual models.

and were denoted as $\xi_{2,3} = Independent$. The stroke starting points were $L_1 = (21, 34)$. The stroke connection positions were as follows: the i-th stroke was connected to some position of a stroke in the front and were denoted as $\tau_{1,2} = (23, 45)$ and $\tau_{3,1} = (22, 56)$, respectively.

e: GENERATION OF THE CHARACTER CONCEPTUAL MODEL

The conceptual model ϕ for each character consists of k models that are of different writing styles and stroke connection relationships. The concept model was obtained by computing the joint probability distribution of ϕ. The $n = 1$ conceptual model was generated for a character when one training process was carried out. The models $\phi^1, \ldots, \phi^k (k = 5)$ with the highest probabilities, as shown in Table 2, could be selected as the conceptual models of this character, as shown in Fig. 19 (the numbers are stroke sequences and starting positions and different colors indicate different writing sequences).

3) PREDICTION OF CLASSIFICATION FOR TARGET SAMPLE

a: TARGET SAMPLE PRE-PROCESSING

The target character is shown in Fig. 20. The target sample received pre-processing to ensure that all the character sizes were 105×105. This was followed by sample image binarization, with the pixel value of the character area set to 1 and that of the blank area set to 0.

b: CONCEPTUAL MODEL MCMC SAMPLING

A total of 5×10 sampling samples (ϕ^{kn}: $\phi^{1,1}, \ldots, \phi^{1,10}, \ldots, \phi^{5,1}, \ldots, \phi^{5,10}$) were obtained by means of MCMC sampling of each of the five candidate conceptual models $\phi^1, \phi^2, \ldots, \phi^5$ built, as shown in Fig. 21.

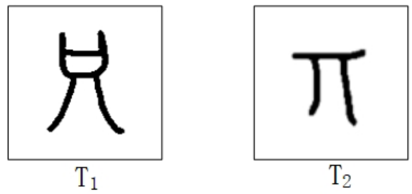


FIGURE 20. Characters in the target samples.

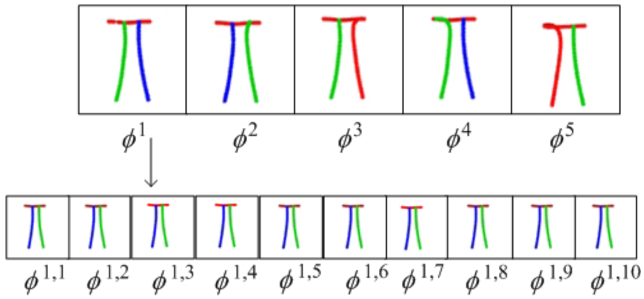


FIGURE 21. Five candidate models and the ten Monte Carlo Markov chain (MCMC) samples of ϕ^1.

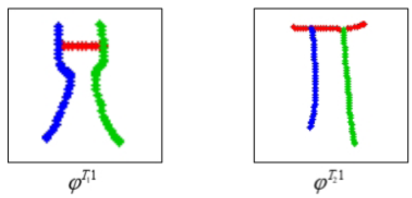


FIGURE 22. Generation model.

c: MCMC SAMPLE-BASED GENERATION MODEL

Each stroke was traversed based on each MCMC sample ϕ^{kn}. A fitting model φ^{Tk} was generated by optimization of the starting point, shape, and scale parameter of each stroke. When the conceptual model was ϕ^1, the starting point, shape, and scale parameter of each stroke in conceptual model ϕ^1 could be transformed based on the target sample, thereby yielding the corresponding generation model φ^{Tk}. The generation models $\varphi^{T_1 1}$ and $\varphi^{T_2 1}$ are shown in Fig. 22.

d: A^{Tk} GLOBAL PARAMETER TRANSFORMATION

Since the handwritings of the same character can vary in shape and the centroid of the pixel points corresponding to the handwritings are not fixed, the four parameters ($A = [\frac{w^{Tk}}{w^T}, \frac{h^{Tk}}{h^T}, (x^{Tk}_{mid} - x^T_{mid}), (y^{Tk}_{mid} - y^T_{mid})]$) of A^{Tk} were determined by the training sample and generation model A^{Tk} after the φ^{Tk} was fitted by ϕ^{kn} at the stroke level. For fitting model $\varphi^{T_2 1}$, the four parameters of A^{Tk} were 1.170, 0.677, 14.944, and −4.157. Strokes were mapped by A^{Tk}, thereby ensuring that the handwritings in generation model $\varphi^{T_2 1}$ fell into the handwriting range of target sample T_2 as much as possible.

e: TARGET SAMPLE CLASSIFICATION PREDICTION

Based on fitting models $\varphi^{T_1 1}$ and $\varphi^{T_2 1}$, there was a large change in the stroke in generation model $\varphi^{T_1 1}$, whereas the

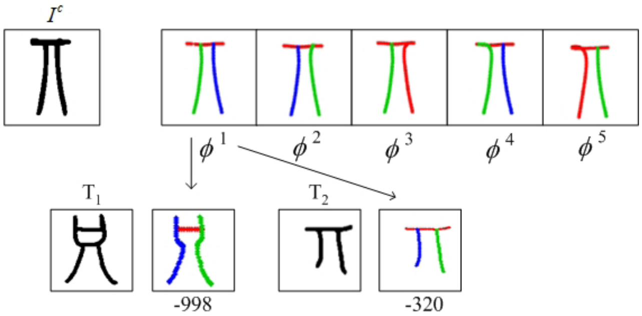


FIGURE 23. Classified prediction of the target sample of traditional XiaoZhuan characters.

stroke of conceptual model ϕ^1 varied little due to the generation model $\varphi^{T_2 1}$. The less the stroke changed, the higher the probability was. In (6), represents the degree of the approximation between the strokes before and after the fitting. The better the stroke of model $\varphi^{T_2 1}$ approximated target sample T_2, the larger $P(I^{T_2}|\varphi^{T_2 1})$. Therefore, $P(I^T|I^C)$ could be maximized only when the training and target samples were of the same classification. As shown in Fig. 23, the probabilities of target sample T_1/T_2 and training sample I^C could be calculated using (8) and (9), respectively:

$$\log(\max_{\varphi^{T_1 1}} P(I^{T_1}|\varphi^{T_1 1})\frac{1}{N}\sum_{n=1}^{N} P(\varphi^{T_1 1}|\phi^{[1n]})) = -998, \tag{8}$$

$$\log(\max_{\varphi^{T_2 1}} P(I^{T_2}|\varphi^{T_2 1})\frac{1}{N}\sum_{n=1}^{N} P(\varphi^{T_2 1}|\phi^{[1n]})) = -320. \tag{9}$$

By comparison, the probability that target sample T_2 and the training sample I^C fell into the same classification was larger than that of the target sample T_1 and the training sample I^C being the same classification. In other words, target sample T_2 was of the same classification as training sample I^C. Thus, handwritten character classification prediction was realized.

To verify the effectiveness of the concept learning method in character recognition, the character recognition accuracy were evaluated. Four hundred different standard XiaoZhuan characters were stochastically divided into 20 groups. Each group contained 20 classifications in which there was one standard XiaoZhuan character used as the training sample and five corresponding handwritten XiaoZhuan characters as the target samples. Thus, there were 100 target samples in each group. The experimental results using the proposed method are shown in Fig. 24.

C. COMPARISON WITH DIFFERENT METHODS

To further demonstrate the performance of our method, we compared the proposed method with the state-of-the-art approaches for their effectiveness on the ICDAR-2013 HCCR competition dataset. The ICDAR-2013 dataset consists of CASIA-OLHWDB1.0 & 1.1 and ICDAR-2013 test set [33]. The ICDAR-2013 test set was sampled from 60 different writers and includes 224,590 samples from 3755 character classes. Some samples of handwritten Chinese characters are shown in Fig. 25. The experimental results are summarized in Table 3.

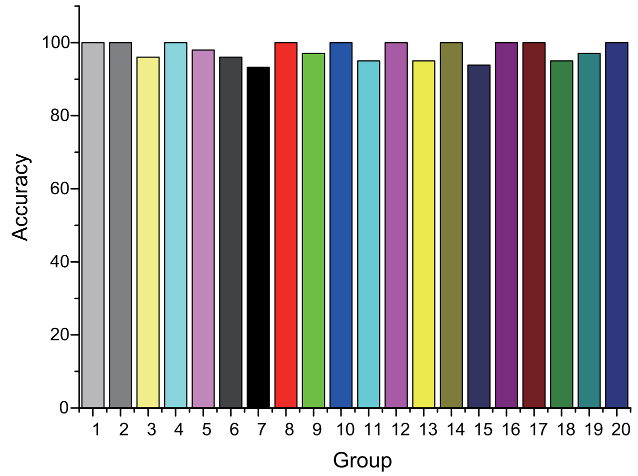


FIGURE 24. Recognition accuracy of XaioZhuan character using our method.

FIGURE 25. Examples of handwritten Chinese characters in ICDAR-2013 dataset.

TABLE 3. Performance comparison of different methods on ICDAR-2013 HCCR competition database.

Methods	Ref.	Accuracy (%)
Human Performance	[33]	95.19
DFE + DLQDF	[6]	95.31
ICDAR-2011 Winner:VO-3	[18]	95.77
ICDAR-2013 Winner:UWarwick	[33]	97.39
ICDAR-2013 Runner-up:VO-3	[33]	96.87
DropSample-DCNN	[19]	97.23
DropSample-DCNN+Ensemble	[19]	97.51
DirectMap-ConvNet	[34]	97.55
DirectMap-Convnet+Ensemble	[34]	97.64
RNN:NET4	[24]	97.76
RNN:NET4-subseq30	[24]	97.89
RNN:Ensemble-NET123456	[24]	98.15
RNN:Simple+NCR	[35]	97.30
RNN:Ensemble+NCR	[35]	97.60
Our approach ($n = 1$)		97.32
Our approach ($n = 5$)		98.20

As revealed in results for ICDAR-2013 in Table 3, the traditional method [6] slightly outperforms the human performance with respect to the accuracy; however, the deep-learning-based approaches surpass the traditional method to a large extent [18], [19], [24], [33]–[35]. The winner of the ICDAR-2011 competition employed a multilayer perceptron (MLP) classifier. Consequently, the prevalent deep-learning methods have dominated the field of HCCR.

The previous work used the path signature feature map and a deep convolutional neural network (CNN) [33]. A better performance (97.64%) was achieved in [34] with domain-specific knowledge and a CNN. However, CNN methods have a particular requirement of transforming the online character handwriting into some image-like representation and generating sufficient extra sampling to improve recognition accuracy. Recently, the best performance was achieved in [24], which requires a large amount of complex data processing, such as, eliminating redundant points and coordinate normalization. Subsequently, to reduce the complexity of data processing, Ren *et al.* [35] used a recurrent neural networks (RNN) directly to deal with the raw sequential data by a simple coordinate normalization, which achieves a better performance (97.60%).

We here propose a handwritten Chinese character recognition method that uses concept learning and compare it to these previous methods of HCCR. As shown by the results on ICDAR-2013 in Table 3, our approach achieves competitive recognition performance with as few as one training sample from the mixed dataset composed of the standard Chinese character database and OLHWDB1.1. In the previous experiments, better recognition performance can be achieved by using the standard Chinese character database alone. To improve the performance, we used the mixed dataset to train the model. Our method builds $n = 5$ concept models where k is set to 5 for each character during the training process, which means that training samples are augmented and the model performance is also improved. One conceptual model for each character consists of k models $\phi^1, \ldots, \phi^k$. Subtle differences may exist among these models. Moreover, the more complicated the character samples are, the more differently the models may be built at every training stage. Thus, many conceptual models need to be built to improve the performance. The number of corresponding fitting models increases. Each fitting corresponds to a predictive result of a character. Finally, the recognition results of target characters are obtained by computing the average value of predictive results.

For handwritten Chinese character recognition, our method is advantageous as only one sample is needed to train the model, which is different from the state-of-the-art deep-learning methods which have a requirement of a large amount of data to serve as a training model. Our method has also another competitive characteristic. For previous deep-learning approaches, the test set is nearly identical to the training set in feature distributions. On the contrary, the model for our method is trained by the mixed dataset, and the target characters from handwritten characters of ICDAR-2013 test set are verified. This means our method has more adaptability for new samples and more generalization ability than the other methods. The applicability of our approach is that it can be also used to recognize Latin character, English character, digit as well as various characters, which have a requirement of stroke starting/ending point information.

However, a limitation of our method is that the number of conceptual models will be increased by character complexity to obtain better performance, which will be time-consuming for constructing and fitting these models.

We have also paid attention to IAHCC-UCAS2016 dataset which is an in-air handwritten Chinese character dataset and is applied in previous many research work for character recognition. Each character in the IAHCC-UCAS2016 was written in a single stroke not including any pen-up/pen-down information [25], [35], [36]. However, the construction of conceptual model depends on the relationship among strokes which has s requirement of a stroke starting/ending point information. Our stroke extraction approach may be not suitable for use on IAHCC-UCAS2016. When our method is applied to address the problem of character recognition like IAHCC-UCAS2016 dataset, we will develop a new stroke extraction method to obtain strokes with starting/ending point information in future work.

V. CONCLUSION

In this study, we advocate the use of the concept learning for the recognition of handwritten Chinese characters. This work is the first, to the best our knowledge, to propose a handwritten Chinese character recognition approach based on concept learning. In our method, a Chinese character conceptual model was built by using character stroke extraction and Bayesian program learning, and a character generation model for each character conceptual model built by using Monte Carlo Markov Chain sampling during the character recognition. The experimental results show that the proposed method can train the conceptual model for character classification prediction using as few as one character sample. Moreover, our method obtains better performance than the state-of-the-art methods on ICDAR-2013 dataset.

ACKNOWLEDGMENT

The authors would like to thank LetPub for its linguistic assistance during the preparation of this manuscript. They would also like to thank CASIA (http://www.nlpr.ia.ac.cn/databases/handwriting/Home.html) for its database of CASIA-OLHWDB. The authors are grateful to all of the reviewers for suggestions and insights that improved the paper.

REFERENCES

[1] R. Plamondon and S. N. Srihari, "Online and off-line handwriting recognition: A comprehensive survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 1, pp. 63–84, Jan. 2000. doi: 10.1109/34.824821.

[2] C. L. Liu, S. Jaeger, and M. Nakagawa, "Online recognition of Chinese characters: The state-of-the-art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 198–213, Feb. 2004. doi: 10.1109/tpami.2004.1262182.

[3] T. Plötz and G. A. Fink, "Markov models for offline handwriting recognition: A survey," *Int. J. Document Anal. Recognit.*, vol. 12, no. 4, pp. 269–298, Oct. 2009. doi: 10.1007/s10032-009-0098-4.

[4] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 855–868, May 2009. doi: 10.1109/tpami.2008.137.

[5] T. Wang, D. J. Wu, A. Coates, and A. Y. Ng, "End-to-end text recognition with convolutional neural networks," in *Proc. 21st Int. Conf. Pattern Recognit.*, Tsukuba, Japan, Nov. 2012, pp. 3304–3308.

[6] C.-L. Liu, F. Yin, D.-H. Wang, and Q.-F. Wang, "Online and offline handwritten Chinese character recognition: Benchmarking on new databases," *Pattern Recognit.*, vol. 46, no. 1, pp. 155–162, 2013. doi: 10.1016/j.patcog.2012.06.021.

[7] M. Kozielski, P. Doetsch, and H. Ney, "Improvements in RWTH's system for off-line handwriting recognition," in *Proc. 12th Int. Conf. Document Anal. Recognit.*, Washington, DC, USA, Aug. 2013, pp. 935–939. doi: 10.1109/icdar.2013.190.

[8] P. Doetsch, M. Kozielski, and H. Ney, "Fast and robust training of recurrent neural networks for offline handwriting recognition," in *Proc. 14th Int. Conf. Frontiers Handwriting Recognit.*, Heraklion, Greece, Sep. 2014, pp. 279–284. doi: 10.1109/icfhr.2014.54.

[9] G. Bideault, L. Mioulet, C. Chatelain, and T. Paquet, "Spotting Handwritten Words and REGEX using a two stage BLSTM-HMM architecture," *Proc. SPIE*, vol. 9402, Feb. 2015, Art. no. 94020G. doi: 10.1117/12.2075796.

[10] A. Graves and J. Schmidhuber, "Offline handwriting recognition with multidimensional recurrent neural networks," in *Proc. 21st Int. Conf. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2008, pp. 545–552.

[11] T. Bluche, J. Louradour, and R. Messina, "Scan, attend and Read: End-to-end handwritten paragraph recognition with MDLSTM attention," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Kyoto, Japan, Nov. 2017, pp. 1050–1055. doi: 10.1109/icdar.2017.174.

[12] A. Graves and S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proc. 23rd Int. Conf. Mach. Learn.*, Pittsburgh, PA, USA, Jun. 2006, pp. 369–376. doi: 10.1145/1143844.1143891.

[13] M. Kang and D. Palmer-Brown, "A modal learning adaptive function neural network applied to handwritten digit recognition," *Inf. Sci.*, vol. 178, no. 20, pp. 3802–3812, Oct. 2008. doi: 10.1016/j.ins.2008.05.011.

[14] F. Zamora-Martínez, V. Frinken, S. España-Boquera, M. J. Castro-Bleda, A. Fischer, and H. Bunke, "Neural network language models for off-line handwriting recognition," *Pattern Recognit.*, vol. 47, no. 4, pp. 1642–1652, Apr. 2014. doi: 10.1016/j.patcog.2013.10.020.

[15] S. Impedovo, "More than twenty years of advancements on Frontiers in handwriting recognition," *Pattern Recognit.*, vol. 47, no. 3, pp. 916–928, Mar. 2014. doi: 10.1016/j.patcog.2013.05.027.

[16] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic data and artificial neural networks for natural scene text recognition," Jun. 2014, *arXiv:1406.2227*. [Online]. Available: https://arxiv.org/abs/1406.2227

[17] A. Poznanski and L. Wolf, "CNN-N-gram for handwritingword recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 2305–2314. doi: 10.1109/cvpr.2016.253.

[18] C.-L. Liu, F. Yin, Q.-F. Wang, and D.-H. Wang, "ICDAR 2011 Chinese handwriting recognition competition," in *Proc. Int. Conf. Document Anal. Recognit.*, Beijing, China, Sep. 2011, pp. 1464–1469. doi: 10.1109/icdar.2011.291.

[19] W. Yang, L. Jin, D. Tao, Z. Xie, and Z. Feng, "DropSample: A new training method to enhance deep convolutional neural networks for large-scale unconstrained handwritten Chinese character recognition," *Pattern Recognit.*, vol. 58, pp. 190–203, Oct. 2016. doi: 10.1016/j.patcog.2016.04.007.

[20] W. Yang, L. Jin, Z. Xie, and Z. Feng, "Improved deep convolutional neural network for online handwritten Chinese character recognition using domain-specific knowledge," in *Proc. 13th Int. Conf. Document Anal. Recognit. (ICDAR)*, Tunis, Tunisia, Aug. 2015, pp. 551–555. doi: 10.1109/icdar.2015.7333822.

[21] A. Graves and S. Fernández, and J. Schmidhuber, "Multi-dimensional recurrent neural networks," in *Proc. Int. Conf. Artif. Neural Netw.*, 2007, pp. 549–558. doi: 10.1007/978-3-540-74690-4_56.

[22] H. Ren, W. Wang, K. Lu, J. Zhou, and Q. Yuan, "An end-to-end recognizer for in-air handwritten Chinese characters based on a new recurrent neural networks," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Hong Kong, Jul. 2017, pp. 841–846. doi: 10.1109/icme.2017.8019443.

[23] Z. Xie, Z. Sun, L. Jin, H. Ni, and T. Lyons, "Learning spatial-semantic context with fully convolutional recurrent network for online handwritten chinese text recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 8, pp. 1903–1917, Aug. 2018. doi: 10.1109/tpami.2017.2732978.

[24] X. Y. Zhang, F. Yin, Y.-M. Zhang, C.-L. Liu, and Y. Bengio, "Drawing and recognizing chinese characters with recurrent neural network," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 849–862, Apr. 2018. doi: 10.1109/tpami.2017.2695539.

[25] X. Qu, W. Wang, K. Lu, and J. Zhou, "Data augmentation and directional feature maps extraction for in-air handwritten Chinese character recognition based on convolutional neural network," *Pattern Recognit. Lett.*, vol. 111, pp. 9–15, Aug. 2018. doi: 10.1016/j.patrec.2018.04.001.

[26] Z. Ghahramani, "Probabilistic machine learning and artificial intelligence," *Nature*, vol. 521, no. 7553, pp. 452–459, May 2015. doi: 10.1038/nature14541.

[27] B. M. Lake, R. Salakhutdinov, and J. B. Tenebaum, "Human-level concept learning through probabilistic program induction," *Science*, vol. 350, no. 6266, pp. 1332–1338, 2015. doi: 10.1126/science.aab3050.

[28] A. Ahmed and A. Bikmal, "One-shot concept learning by simulating evolutionary instinct development," Aug. 2017, *arXiv:1708.08141*. [Online]. Available: https://arxiv.org/abs/1708.08141

[29] L. Lam, S.-W. Lee, and C. Y. Suen, "Thinning methodologies-a comprehensive survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 9, pp. 869–885, Sep. 1992. doi: 10.1109/34.161346.

[30] K. Liu, Y. S. Huang, and C. Y. Suen, "Identification of fork points on the skeletons of handwritten Chinese characters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 10, pp. 1095–1100, Oct. 1999. doi: 10.1109/34.799914.

[31] C.-W. Liao and J. S. Huang, "Stroke segmentation by bernstein-bezier curve fitting," *Pattern Recognit.*, vol. 23, no. 5, pp. 475–484, 1990. doi: 10.1016/0031-3203(90)90068-v.

[32] L. F. O. Chamon and A. Ribeiro, "Greedy sampling of graph signals," *IEEE Trans. Signal Process.*, vol. 66, no. 1, pp. 34–47, Jan. 2018. doi: 10.1109/tsp.2017.2755586.

[33] F. Yin, Q.-F. Wang, X.-Y. Zhang, and C.-L. Liu, "ICDAR 2013 Chinese handwriting recognition competition," in *Proc. 12th Int. Conf. Document Anal. Recognit.*, Washington, DC, USA, Aug. 2013, pp. 1464–1470. doi: 10.1109/icdar.2013.218.

[34] X-Y. Zhang, Y. Bengio, and C.-L. Liu, "Online and offline handwritten Chinese character recognition: A comprehensive study and new benchmark," *Pattern Recognit.*, vol. 61, pp. 348–360, Jan. 2017. doi: 10.1016/j.patcog.2016.08.005.

[35] H. Ren, W. Wang, and C. Liu, "Recognizing online handwritten Chinese characters using RNNs with new computing architectures," *Pattern Recognit.*, vol. 93, pp. 179–192, Sep. 2019. doi: 10.1016/j.patcog.2019.04.015.

[36] X. Qu, W. Wang, K. Lu, and J. Zhou, "In-air handwritten Chinese character recognition with locality-sensitive sparse representation toward optimized prototype classifier," *Pattern Recognit.*, vol. 78, pp. 267–276, Jun. 2018. doi: 10.1016/j.patcog.2018.01.021.

LIANG XU (M'19) received the master's degree in control theroy and control engineering from the Lanzhou University of Technology, in 2002, and the Ph.D. degree in computer science and technology from the South China University of Technology, in 2007. He is currently an Associate Professor with the School of Automation, Guangdong University of Technology. His current research interests include machine vision, wireless sensor networks, character recognition, and deep learning.

YUXI WANG received the B.Sc. degree in electronics and communications engineering from Guangdong Polytechnic Normal University, in 2017. He is currently pursuing the master's degree in control theory and control engineering with the School of Automation, Guangdong University of Technology. His research interests include machine learning, character recognition, and concept learning.

XIUXI LI received the master's and Ph.D. degrees in chemical engineering from the South China University of Technology, in 1998 and 2008, respectively, where he is currently a Professor with the School of Chemical and Chemical Engineering. His current research interests include chemical process safety, product quality control, and process optimal.

MING PAN received the B.Sc. degree from Zhejiang University, in 2003, and the Ph.D. degree from the South China University of Technology, in 2008, both in chemical engineering. He is currently an Associate Professor with the School of Chemical Engineering and Technology, Sun Yat-sen University. His current research interests include industrial intelligence, process system engineering, product quality control, and process optimal.