IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 29, 2020, accepted September 15, 2020, date of publication September 28, 2020, date of current version October 9, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3027019

A Novel Semantic Segmentation Model for Chinese Characters

ZHENYU GAO[1], JIN LIU[1], (Member, IEEE), YIYAO LI[1], YIHE YANG[1], AND HUIHUA HE[2]
[1]College of Information Engineering, Shanghai Maritime University, Shanghai 201306, China
[2]College of Education, Shanghai Normal University, Shanghai 200234, China

Corresponding author: Huihua He (hehuihua@shnu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61872231 and Grant 61701297, and in part by the Shanghai Education Research Fund Project under Grant C17053.

ABSTRACT Character segmentation plays an important role in optical character recognition (OCR). Due to the limitations of feature representation, traditional image analyzing based methods cannot well segment characters with connected or broken strokes, especially for the Chinese characters which usually have complex structures. To solve this issue, this paper proposes a novel segmentation model based on fully convolutional neural networks (FCN). The model first uses convolutional neural networks to extract spatial features, then shares them throughout the whole model. Two FCNs are used to extract character information to form a score map. Finally, character features are reused to adjust the accurate segmentation points in the score map. What's more, to strengthen the ability of feature representation, a novel compound character feature which can well describe the characters' outline is also proposed. The proposed method is validated on two datasets: GBSD and CASIA-HWDB-MT, against the methods proposed in the literature. Experimental results show that the proposed model outperforms state-of-the-art methods.

INDEX TERMS Chinese character segmentation, character feature extraction, fully convolutional neural networks.

I. INTRODUCTION

Character segmentation is an important task in optical character recognition (OCR) process. The quality of the segmentation not only effects the OCR performance, but also plays vital roles in a variety of related application. For example, using machine to read CAPTCHA, which is a challenge–response test to determine whether or not the user is human, needs sophisticated character segmentation algorithm to correctly figure out each character from a row of them with sticky strokes. Therefore, character segmentation is a complicated task, especially for pictographic characters such as Chinese. Compared to the character of alphabet and numbers, Chinese characters have more complex structure, which are easily segmented into wrong blocks by the computers. Several OCR techniques have been developed during the past decade. However, these techniques simply focus on the improvement of the recognition, rather than the performance of character segmentation. Therefore, it is of great significance to study how to make character segmentation more accurate. In this paper, we take Chinese character segmentation as an independent work.

Contemporary character segmentation methods can be divided into two categories – segmentation methods based on image analysis and segmentation methods based on deep learning. The former requires the extraction of structural and statistical features such as the height and width of the characters, segmenting character images into character blocks. However, features extracted by these methods are too shallow. To extract high-level features, Deep Learning based techniques are used in these years. Deep learning has a variety of applications, like ontology [1], object detection [28] and image segmentation [20]. Compared to segmentation method based on image analysis, these methods have a strong ability of feature learning. However, these methods take segmentation as a detection task, usually used for text line segmentation [14], therefore cannot segment Chinese characters well. In 2015, Long *et al.* [20] firstly regarded the text detection problem as a semantic segmentation problem, and proposed using fully convolutional neural networks (FCN), which can classify each pixel in the image, taking consideration of characters' detail. But this method still fails to obtain fine

The associate editor coordinating the review of this manuscript and approving it for publication was Donato Impedovo.

segmentation results, due to the lack of spatial relations between pixels and sparse features after up-sampling.

In this article, we propose a Chinese character segmentation method (CCSeg). CCSeg is based on fully convolutional neural networks, and consists of three parts. First, it uses the convolutional neural network to extract image level features which are later shared throughout the network. Then, for the extraction of multiple character feature, two sub-networks are used. Finally, character features are reused to adjust the accurate segmentation points in the score map. Moreover, CCSeg has a novel multi-object feature extraction mechanism, which can exploit regional multiple character feature. Traditionally, character feature extracted describes rough character morphology. In this paper, regional multiple character feature can well describe the outline of characters. Experimental results show that the proposed method outperforms the traditional methods.

This paper is organized as follows: Section II introduces the related work, Section III gives a detailed description of the proposed method; In Section IV, our method is illustrated in detail. In Section V, we evaluate the experimental results, as well as a comparison with several state-of-the-art methods. Finally, the work is concluded in Section VI.

II. RELATED WORK

A. SEGMENTATION METHOD BASED ON IMAGE ANALYSIS

Segmentation methods based on image analysis can be divided into four categories: namely water droplet segmentation method [2], projection method [21], connected domain analysis method [22], and clustering method [36]. These methods have experienced long-term development from 1990s to 2000s. In 1995, the drip-water segmentation algorithm was firstly proposed by Congedo G *et al.*. This algorithm mimics the process of dropping water from a high point to a low point and mainly aims to segment characters with adhesions. However, when it comes to segment characters like verifying human operation with CAPTCHA, it is impossible to determine an accurate drip leak because of the twisted strokes and concave parts. In 2003, Pal and Datta *et al.* [3] presented a projection analysis-based algorithm. To deal with the variability of writing style from different individuals, the whole text is divided into vertical stripes. Then the horizontal histogram of these stripes are used to segment text lines. Finally, these lines are segmented into words based on vertical projection profile. This method has simple calculation and is widely used in various systems. Furthermore, to improve the accuracy of segmentation, morphological processing [6] and multi-level projection [8] are applied in the segmentation tasks. Specially, instead of multi-level projection, Manmatha and Rothfeder (2005) [9] only performed projection in the horizontal direction to obtain the location of the text line. In this method, they use the contour results of the projection in the horizontal direction to obtain the location of the text line. Results need to be corrected during the process. In addition, several other image histogram-based projection segmentation methods were proposed (U.V. Marti 2002 [10], B. Gatos 2007 [11], and Rodolfo P 2009 [14]). Besides the projection method, Shi Z *et al.* (2009) [5] also used filters with various directions to construct a generalized adaptive local connected graph, mainly used for text lines with different directions. Unlike the above methods, the clustering method uses basic components in text images, such as pixels, connected blocks, stroke segments, etc., to make up characters.

B. SEGMENTATION METHOD BASED ON DEEP LEARNING

Deep Learning (DL) is an important research direction in the field of Machine Learning (ML). It was firstly proposed by Hinton and Salakhutdinov [23] in 2006. At the same time, a layer-by-layer training method was also presented. At present, convolutional neural networks (CNN) are commonly used in OCR tasks. A large amount of models has been improved on this basis such as AlexNet in 2012, ZF Net in 2013, VGG Net and Google Net in 2014, and ResNet [24] in 2015. At present, there are several main research works: R-CNN, Fast R-CNN, Faster R-CNN and Mask R-CNN [28]–[31] which are focused on object detection and classification. In recent years, some researchers have used deep learning techniques for the text segmentation. From 2015 to 2017, Moysset *et al.* [16]–[19] proposed Recurrent Neural Network (RNN), Local- Recurrent Neural Network (LRNN) and Multi-dimension Long Short-Term Memory (LSTM) to predict the boundary area of the text area. Compared to traditional segmentation methods based on image analysis, these networks can bring more comprehensive feature representation. However, segmentation results obtained by these methods are still coarse. To make a detailed description of the contour, FCN was firstly proposed by Long Jonathan *et al.* in 2015 [20]. The FCN model uses a convolutional layer instead of a fully connected layer to classify each pixel in the image. In recent years, an increasing number of FCN-based methods has been proposed. Chen Liang-Chieh *et al.* (2014) [32] used fully connected conditional random fields (CRFs) after FCN structures, adding more spatial information to the model. Shuai Zheng *et al.* (2015) [33] integrated the structure of CNN on this basis. Konstantinos Kamnitsas *et al.* (2016) [34] extended this structure to a three-dimensional scale which can be applied to brain tumor segmentation. To extract 'thicker' feature, U-NET combines feature maps in channel dimension. In addition to semantic segmentation, Mask R-CNN also produces a kind of instance segmentation to obtain more accurate classes of the objects.

III. PROBLEM DESCRIPTION

Traditional character segmentation methods cannot segment characters with connected or broken strokes due to the limitation of feature representation. In recent years, deep learning methods have begun to be used in OCR tasks, and have achieved better results than traditional methods. However, such research works pay more attention to character

recognition, but greatly neglect the importance of segmentation. In this work, we regard character segmentation as an independent task of semantic segmentation in images, and propose an improved full convolutional neural network model.

Our solution of character segmentation in text images is formalized as follows. An input is defined as:

$$\mathrm{G}_{in}^{i\times \acute{\mathrm{A}}j} = \left(\left(g_{11},\ldots,g_{1j}\right),\ldots,\left(g_{i1},g_{i2},\ldots,g_{ij}\right)\right) \quad (1)$$

where g_{ij} is the value of pixel in the location (i, j) of an input image. After processed by convolutional networks, we can get two feature tensors–*CMF* and *CNF*, and a semantic segmentation map F.

$$CMF = [w_1, w_2, \ldots, w_n] \quad (2)$$

$$CNF = [n_1, n_2, \ldots, n_n], \quad (3)$$

$$F_{out}^{i\times j} = \left(\left(\widehat{g_{11}},\ldots,\widehat{g_{1j}}\right),\ldots,\left(\widehat{g_{i1}},\widehat{g_{i2}},\ldots,\widehat{g_{ij}}\right)\right), \quad (4)$$

where *CMF* is used to denote the character morphology feature which is represented by the width of each character. *CNF* is used to denote the character number feature which is represented by the number of characters in each text line. w_{i} and n_{i} represent *CMF* and *CNF* of the *i*th image respectively. $\widehat{g_{ij}}$ is the value of pixel in the location (i, j) of an output image. The output F has the same size as the input G. Finally, accurate segmentation points are marked in F with *CMF* and *CNF*.

IV. METHODS

A. COMPOUND CHARACTER FEATURE EXTRACTION

Character feature extraction in this paper is based on convolutional neural networks (CNN). Unlike traditional image-analysis based methods, CNN have the ability to extract advanced features. The basic components of CNN are convolution, pooling and activation functions. In each location (i, j), we use x_{ij} as the input data vector in one particular layer, and y_{ij} as the output.

$$y_{ij} = f_{ks}(\{X_{si+\delta i,sj+\delta j}\}\ 0 \le \delta i, \delta j \le k) \quad (5)$$

where k is the kernel size, s is the stride factor.

Traditional character segmentation methods often use shallow information so that they cannot find precise segmentation points, greatly reducing the accuracy of recognition. In addition, traditional FCN are not sensitive to image details and neglects the relationship between pixels. In order to obtain precise results in complex tasks such as adhesion, missing, etc., this paper proposes compound character feature extraction. The proposed extraction method extracts feature from both character morphology and number, for the final semantic segmentation in the deconvolution process. In addition, as shown by the yellow arrow in Figure 1, since *CMF* affects the range of the character region, *CMF* is merged when extracting *CNF*. *CMF* and *CNF* are introduced in section III. We use concatenate operation to merge features:

$$Z_{concat} = \sum_{i=1}^{c} X_i * K_i + \sum_{i=1}^{c} Y_i * K_{i+c} \quad (6)$$

where $X_{1\ldots c}$ and $Y_{1\ldots c}$ are the two feature channels that need to be fused. K represents the convolution kernel, and $*$ represents the convolution operation. Z represents the output.

To extract these two different features, we construct two sub-networks, which are named as CMFEN and CNFEN respectively. On the one hand, we construct CMFEN to reduce the interference of many irrelevant pixel information in the semantic segmentation task. On the other hand, we construct CNFEN to minimize cases like over-segmentation for missing strokes and under-segmentation for sticky strokes.

Inspired by the structure of Faster R-CNN [30], we add a feature sharing mechanism to the proposed network. We send the final feature map from the backbone network to the above two sub-networks, as shown in Figure 1. Besides, *CMFs* are also shared between sub-networks. It is obvious that candidate character regions in the image differ from each other because of containing different character shapes. Thus the morphology of characters can indirectly influence the number of them. At the same time, given a certain character area, difference in the number of characters will also affect the morphology of each character.

B. CCSeg BASED ON COMPOUND CHARACTER FEATURE

This paper takes character segmentation as a semantic segmentation task. Inspired by FCN, the proposed architecture uses the deconvolution layer to upsample the feature map of the last convolutional layer until it has the same size as the input image. Therefore, each pixel can be classified, which can better describe the outline of characters than methods with proposals. The main operation of the proposed network can be described as:

$$f_{ks}^{\circ} g_{k's'} = \left(f^{\circ} g\right)_{k'+(k-1)s',ss'} \quad (7)$$

where k is the kernel size, s is the stride or subsampling factor, f_{ks} and $g_{k's'}$ determine the layer type.

Compared to traditional convolutional neural networks, the input size of FCN is unlimited. Because of the pixel-level operation, FCN can efficiently preserve the spatial information in the original input image. However, since the upsampling process uses simple padding, the results obtained by the FCN are still coarse. In order to produce accurate and detailed segmentation, an improved FCN model based on the character information extraction method is proposed in this paper. The improved FCN use multiple features as the input. After processed by deconvolution layers, the extracted character information is reused for the final semantic segmentation task. Compared with the basic FCN method, the extracted character features will provide more information for the deconvolution process.

As shown in Figure 1, the improved FCN have three outputs, *CMF*, *CNF* and score map F. In the score map F, value of each pixel represents the probability of character class. In this paper, pixels of the whole image are divided into two categories: character class and the non-character class.

Semantic segmentation tasks normally improve the architecture of neural network models to make them perform well

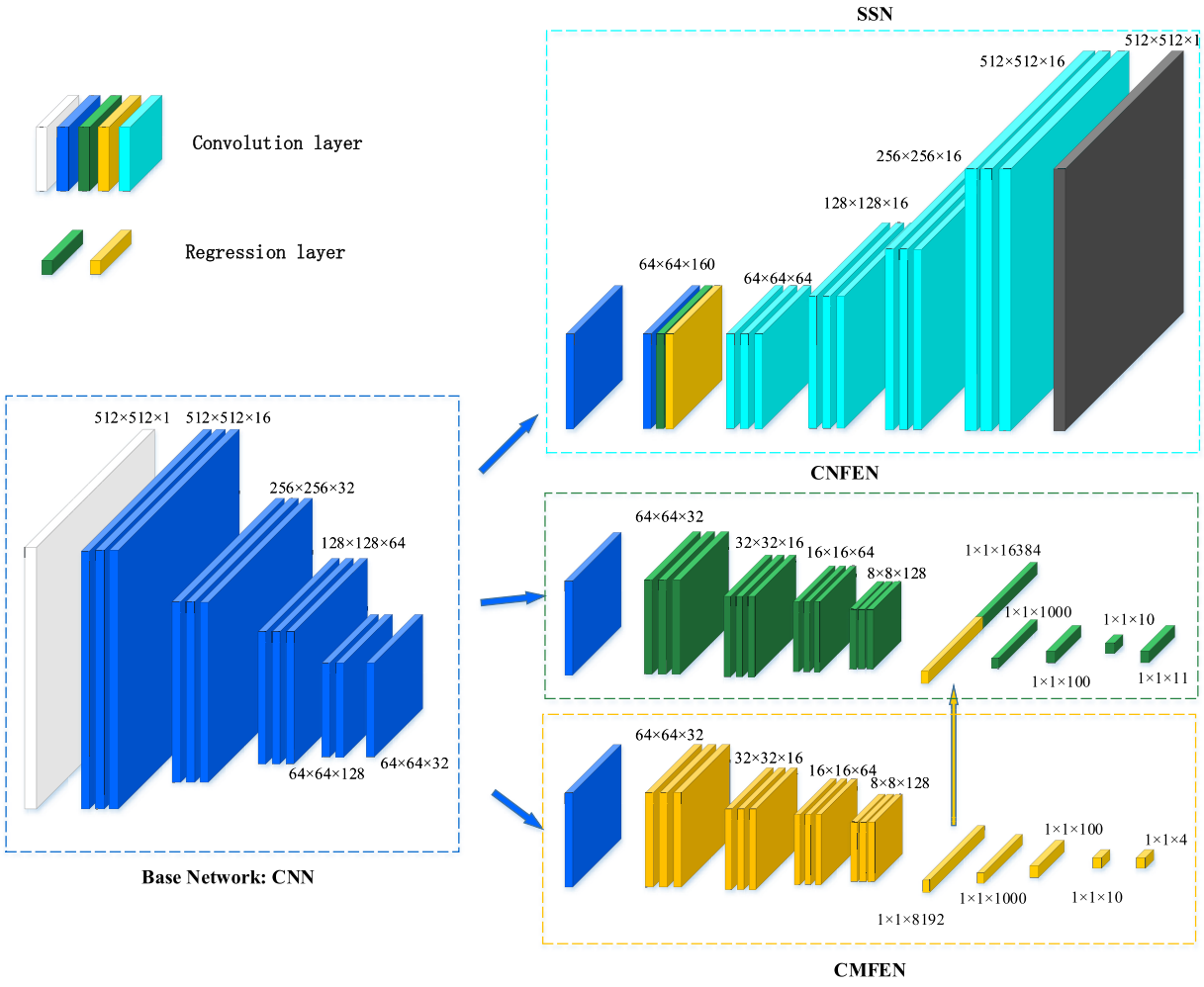

FIGURE 1. The entire architecture of CCSeg. The blue blocks refer to base CNN, the green blocks refer to CNFEN. The yellow blocks refer to CMFEN. The light blue blocks refer to semantic segmentation network (SSN).

in complex tasks. However, it is difficult for a semantic segmentation model to deal with image details. In order to solve this problem, CRFs [35] has been broadly used in semantic segmentation to combine class scores and the extracted low-level information [37]. This paper proposes a new edge-optimizing algorithm based on CRFs. Typically, CRFs are mainly used to smooth noisy segmentation maps. But our CCSeg has already achieved quite smooth score maps and produced homogeneous classification results. As a result, it is not suitable to use traditional short-range CRFs. The edge-optimizing algorithm proposed in this paper does not further smooth the results, but illustrates the relationship between all the pixels in the graph to recover detailed local structure. Additionally, *CNF* and *CMF* are reused in this part. The proposed model employs the energy function as follows:

$$E(x) = \sum_i \theta_i(x_i) + \sum_{ij} \theta_{ij}(x_i, x_j) \tag{8}$$

$$\theta_i(x_i) = -logP(x_i) \tag{9}$$

$$\theta_{ij}(x_i, x_j) = \mu(x_i, x_j)\left[w_1 exp\left(-\frac{\|p_i - p_j\|^2}{2\sigma_\alpha^2} - \frac{\|I_i - I_j\|^2}{2\sigma_\beta^2}\right) + w_2 exp\left(\frac{\|p_i - p_j\|^2}{2\sigma_\gamma^2}\right)\right] \tag{10}$$

where $P(x_i)$ is the class score of pixel i which is computed by the improved FCN, $\theta_{ij}(x_i, x_j)$ means all pairs of image pixels. Especially, in the equation (10), $\mu(\cdot d)$ is a penalty function. It is set to 1 when $x_i \neq x_j$ and is set to 0 otherwise. p_i and I_i are denoted as the position and the RGB color of the *ith* pixel. $\sigma_\alpha, \sigma_\beta, \sigma_\gamma$ are hyper parameters. Experiments show that this algorithm can significantly improve the accuracy of character segmentation.

C. MULTI-OBJECT DROPOUT MECHANISM

Due to the limited size of the self-built dataset GBSD, using traditional dropout mechanism is easy to obtain an over-fitting result. So we introduce a new multi-object dropout mechanism. It allows each neuron node to be set to zero

with a certain probability during the forward propagating. These neurons do not process the data passed by the previous layer nor the next layer. This way, the amount of forward-propagating data and calculation of partial neuron nodes can be reduced. In another forward propagation, neurons of all hidden layers are selected again. During back propagation, random gradient descent method is used to optimize the unselected neurons.

The traditional dropout mechanism [45] trains only a single data feature. For the multiple character feature extraction, we use a multi-object dropout mechanism. This new mechanism can process both CMFEN and CNFEN, which are introduced in section IV-A.

Take the lth layer as an example. In order to make a simple expression, CMF is replaced by M and CNF is replaced by N. The output of neurons is denoted a $\{M, N\}^l$. w_i^{l+1} and b_i^{l+1} can be the weight and bias of the ith neuron in the lth network respectively.

I. The process of neural network without our multi-object dropout mechanism is shown as:

$$z_i^{l+1} = w_i^{l+1} * y^l + b_i^{l+1} \tag{11}$$
$$y_i^{l+1} = f(z_i^{l+1}), \tag{12}$$

where z_i^{l+1} refers to the calculation results of the ith neuron in the lth network, y^{l+1} is the output after activation.

II. The process of neural network with Dropout mechanism is shown as:

$$r_i^l = Bernoulli(p)_i \tag{13}$$
$$\widetilde{\{M, N\}}_i^l = r_i^l * \{M, N\}_i^l, \tag{14}$$
$$z_i^{l+1} = w_i^{l+1} * \widetilde{\{M, N\}}^l + b_i^{l+1} \tag{15}$$
$$\{M, N\}_i^{l+1} = f(z_i^{l+1}) \tag{16}$$

where $Bernoulli(p)_i$ represents the following process. The ith neuron in the lth network randomly generates a 0-1 vector represented by r_i^l with a probability of p. Then this vector is used to process the output $\{M, N\}^l$. $\widetilde{\{M, N\}}_i^l$ is the processing result.

The trained network is used for prediction:

$$z_i^{l+1} = p * w_i^{l+1} * \{M, N\}^l + b_i^{l+1} \tag{17}$$
$$\{M, N\}_i^{l+1} = f(z_i^{l+1}), \tag{18}$$

V. EXPERIMENT

A. DATASET

CASIA-HWDB-MT [40]: CASIA-HWDB-1.0 is an handwriting dataset containing isolated characters. To create a Chinese handwritting character dataset for segmentation, characters from CASIA-HWDB-1.0 are randomly selected to form text lines. The final created dataset is called CASIA-HWDB-MT. The generation rule is illustrated in TABLE 1, including the number of selected character images and the character spacing. Some samples of CASIA-HWDB-MT are shown in Figure 2.

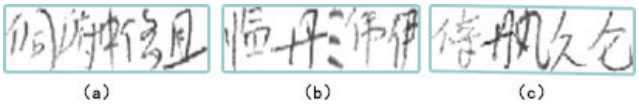

FIGURE 2. Some samples of CASIA-HWDB-MT.

TABLE 1. Key parameters of GBSD.

Parameter	Value
Image size	512*512
Image number	*30000*
Character type	*3755*
Number of characters(per image)	*[2, 5]*
Character size	*[70px, 80px]*
Character spacing	*[-10px, 3px]*
String starting coordinate point x	*[5, 20]*
String starting coordinate point y	*[50, 250]*
Number of noise points(per image)	*[200, 300]*
Number of interference textures(per image)	*[5, 10]*
Regional corrosion parameters(per image)	*cv2.MORPH_ELLIPSE, (3, 3), 1*
Regional expansion parameter(per image)	*cv2.MORPH_ELLIPSE, (3, 3), 1*

TABLE 2. Some samples of GBSD.

Data Type	Input	Label	Proportion
Normal	奘鞲	*2_79*	*5%*
Left-right	譠栩	*2_80*	*80%*
Sticky	師騶銨膴	*4_72*	*95%*
Missing	槮環丨斟	*4_80*	*10%*

GBSD: Since CASIA-HWDB-MT is created primarily to represent sticky strokes. In order to verify the effectiveness of the proposed method when there are missing or broken strokes, we also create a dataset with an increased proportion of missing or broken strokes. GBSD includes 30,000 pictures in total. They are generated by 3,755 Chinese characters in the GB2312 first-level national standard. All the experimental images have a uniform size: 512*512 pixels. The size of the characters in each picture is between 70px and 80px. Each image contains characters ranging from 2 to 5. For data augmentation, white noise and interference textures are manually added when constructing character pictures. Table 1 depicts key parameters of our dataset. Table 2 shows several examples in our data set. There are four types of data in the dataset. Besides the normal type, it also has three kinds of data in complex tasks. Each image is labeled by "number_width". The width here is the average width of characters in an image.

B. IMPLEMENTATION DETAILS

base CNN: For spatial feature extraction, four convolution operation units are constructed. One convolution operation unit is called as a block. The basic structure of each operation

TABLE 3. Base CNN architecture.

Layers	Output size	Operations
Input	*512×512*	
Block (1)	*512×512*	$[3 \times 3\ conv] \times 16$, *stride 1*
	256×256	*max_pooling*
Block (2)	*256×256*	$[3 \times 3\ conv] \times 32$, *stride 1*
	128×128	*max_pooling*
Block (3)	*128×128*	$[3 \times 3\ conv] \times 64$, *stride 1*
	64×64	*max_pooling*
Block (4)	*64×64*	$[3 \times 3\ conv] \times$ 128, *stride 1*
	64×64	$[3 \times 3\ conv] \times 32$, *stride 1*
	32×32	*max_pooling*

TABLE 4. Architecture of CMFEN. Parameter d in the bottom line represents the number of outputs.

Layers	Output size	Operations
Input	*64×64*	
Block (1)	*64×64*	$[3 \times 3\ conv] \times 32$, *stride 1*
	32×32	*max_pooling*
Block (2)	*32×32*	$[3 \times 3\ conv] \times 16$, *stride 1*
	16×16	*max_pooling*
Block (3)	*16×16*	$[3 \times 3\ conv] \times 64$, *stride 1*
	8×8	*max_pooling*
	8×8	$[3 \times 3\ conv] \times$ 128, *stride 1*
	1×1	*flatten*
Block (4)	*fully-connected*	
11d fully-connected		

TABLE 5. Architecture of CNFEN. Parameter d in the bottom line represents the number of outputs.

Layers	Output size	Operations
Input	*64×64*	
Block (1)	*64×64*	$[3 \times 3\ conv] \times 32$, *stride 1*
	32×32	*max_pooling*
Block (2)	*32×32*	$[3 \times 3\ conv] \times 16$, *stride 1*
	16×16	*max_pooling*
Block (3)	*16×16*	$[3 \times 3\ conv] \times 64$, *stride 1*
	8×8	*max_pooling*
	8×8	$[3 \times 3\ conv] \times$ 128, *stride 1*
	1×1	*flatten*
Block (4)	*fully-connected*	
4d fully-connected		

TABLE 6. Architecture of SSN.

Layers	Output size	Operations
Input	*64×64*	
concatenate		
Block (1)	*64×64*	$[3 \times 3\ deconv] \times$ 64, *stride 1*
	128×128	*up_sampling*
Block (2)	*128×128*	$[3 \times 3\ deconv] \times$ 16, *stride 1*
	256×256	*up_sampling*
Block (3)	*256×256*	$[3 \times 3\ deconv] \times$ 16, *stride 1*
	512×512	*up_sampling*
Block (4)	*512×512*	$[3 \times 3\ deconv] \times$ 128, *stride 1*

unit includes: convolution layer – activation layer – convolution layer – activation layer – convolution layer – activation layer – down-sampling layer. Detailed parameter settings of base CNN are shown in the Table 3.

CMFEN: This network uses spatial features extracted by base CNN. Similarly, one convolution operation unit is a block, which contains several convolution layers and pooling layers. Here we use a flatten layer to squeeze features into a 1D sequence of values. This sequence is used to make feature fusion and sent to dense layer. Finally, we get 11 outputs. Table 4 describes more details about this network.

CNFEN: Like CMFEN, it receives spatial features from base CNN. After processed by three blocks and a flatten layer, a concatenate layer is used to fuse features. Finally, we get 6 outputs of dense layers. Table 5 shows all the details of parameter settings.

SSN: Before upsampling, there is also a concatenate layer to fuse all the features obtained by the above three networks. Then the data size is restored to the same size as the input image by deconvolution layers. The value of each pixel in the score map indicates the probability of character class. The detailed parameter settings of this network is given in Table 6.

C. TRAINING

Training settings: All the networks are optimized using Adaptive Moment Estimation (Adam). The loss weight of character information extraction network is set to 0.5. The loss weight of SSN is set to 1.0. According to the loss calculation method, we use Cross-entropy loss to evaluate the loss of CMFEN, CNFEN and SSN. The initial learning rate is set to 1.0. According to the loss calculation method, we use Cross-entropy loss to evaluate the loss of CMFEN, CNFEN and SSN. The initial learning rate is set to 0.0001(1e-4). The exponential decay rate of the first moment estimate is set to 0.9 and that of the second moment estimate is set to 0.999. Besides, to prevent dividing by zero in the calculation, we set epsilon to 1e-08 and set decay to 0.0.

Training: Our experiment set up 5000 iterations of training. The batch size is set to 50. The whole training totally costs about 57 hours. Figure 3 pictorially demonstrate the loss changes of different networks. All the results are recorded by Tensorboard.

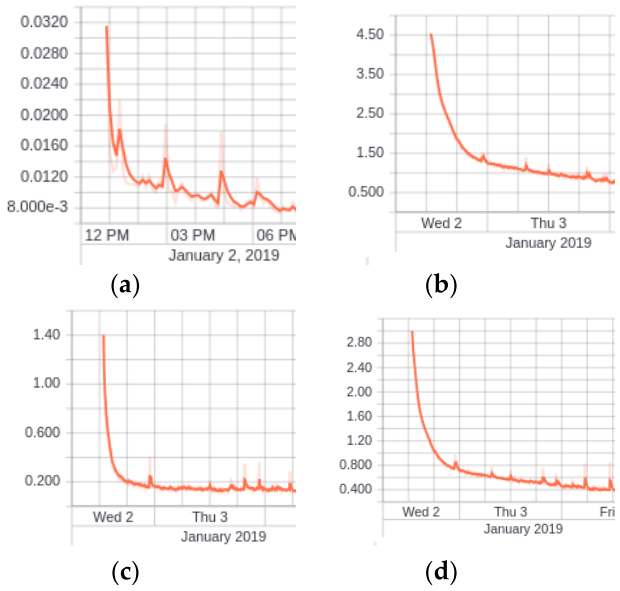

FIGURE 3. The loss change of different networks. (a) Loss change in SSN; (b) Loss change in CMFEN; (c) Loss change in CNFEN; (d) Loss change in the entire neural networks.

D. EXPERIMENTAL RESULTS

Several examples of segmentation results obtained on GBSD are shown in Table 7. There exists a certain amount of random noise interference in the images of the left line. Besides, there are adhesions between these Chinese characters, which are hard to be segmented. For example, "讥–蔷", "怃–导", "獭–耗". Unlike English, Chinese characters have a special structure: a left-right structure. Characters with the left-right structure are easy to be over-segmented. Here are some examples, "嚼", "哺", "幼", "汝", and "儿", etc.

As listed before, there are lots of common but difficult character segmentation problems. For these cases, qualitative samples of semantic segmentation conducted by traditional FCN are shown as the middle column of Table 7. Pixel points closer to white in the figure are classified into character class, while those closer to black are classified as a non-character class. After training, the neural network cannot learn the segmentation information between characters well. Therefore, we cannot receive an accurate result according to the score map. Moreover, after several hundred iterations, the loss of FCN will remain low and it is difficult to further improve this neural network. As shown in the right column of Table 7, we can achieve an even better result by just adding good features into the FCN model. Segmented regions are marked with red boxes. On this basis, the proposed method reuses character information features in the final segmentation process. Segmented regions are marked with green boxes.

The proposed method is also verified on CASIA-HWDB-MT, and the quantitative results are shown in Figure 4. Three types of data are taken for illustration: single-touching string, multiple-touching string and single-touching string with more than two characters. As is shown in Figure 4, both of the other two methods focused on Chinese character segmentation have the problem of over-segmentation and inaccurate segmentation points. However, the proposed method can handle these three situations very well.

To handle Chinese characters with left-right structure, we add *CMF* into our model, so that the characters will not be over-segmented. To handle the situation that adhesions between characters, we add *CNF* into our model. Thus we can segment a specific number of individual character blocks

TABLE 7. Examples of segmentation results based on GBSD.

Original picture	Traditional FCN	Different results
笔沁讥蔷嚼		笔沁讥蔷嚼
哆衅哺象		哆衅哺象
獭耗嚣擦香		獭耗嚣擦香

FIGURE 4. Examples of segmentation results based on CASIA-HWDB-MT. (a) original images; (b) segmentation results of the method proposed by Xu L *et al.* [40]; (c) segmentation results of the method proposed by Tian J *et al.* [39]; (d) segmentation results of the method proposed by this paper.

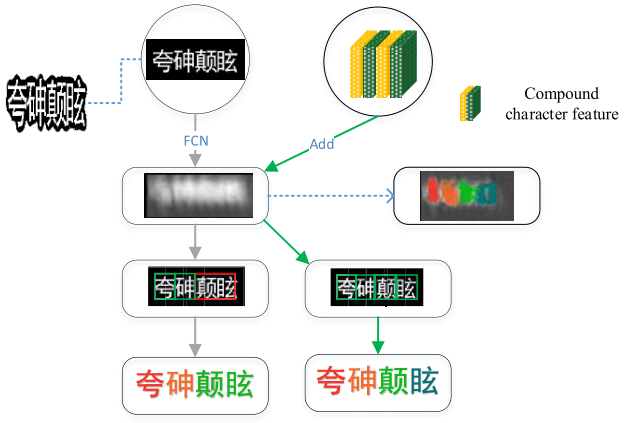

FIGURE 5. Different results obtained by only FCN (left) and our proposed method (right).

TABLE 8. Results of comparative experiments based on GBSD. Higher Ps means better accuracy of segmentation. Lower O_S and U_S mean less error in segmentation.

Methods	C1			C2			C3		
	O_S	U_S	P_S	O_S	U_S	P_S	O_S	U_S	P_S
Vertical projection [21]	*68.13*	*16.30*	*15.57*	*65.39*	*18.15*	*16.46*	*65.27*	*14.23*	*20.50*
Water droplet [1]	*40.00*	*3.56*	*56.43*	*38.00*	*2.31*	*59.69*	*42.12*	*2.00*	*55.88*
Connected domain [22]	*55.90*	*13.70*	*30.40*	*53.22*	*12.34*	*34.44*	*56.00*	*12.42*	*31.58*
Clustering [36]	*36.11*	*8.87*	*55.02*	*36.00*	*7.24*	*56.76*	*40.25*	*8.00*	*51.75*
Multi-layer perceptron [23]	*8.40*	*6.30*	*85.30*	*9.75*	*5.77*	*84.48*	*5.48*	*5.72*	*88.80*
LSTM [19]	*7.02*	*8.50*	*84.48*	*11.71*	*5.27*	*83.02*	*2.83*	*8.25*	*88.92*
FCN [20]	*7.60*	*5.12*	*87.28*	*3.48*	*7.76*	*88.76*	*3.99*	*6.28*	*89.73*
U-Net [42]	*7.20*	*5.30*	*87.50*	*3.55*	*7.69*	*88.76*	*4.05*	*5.95*	*90.00*
DeepLab [35]	*2.97*	*5.12*	*91.92*	*4.21*	*3.89*	*91.90*	*4.95*	*5.16*	*90.88*
Mask R-CNN [41]	*3.64*	*4.22*	*92.14*	*4.32*	*3.78*	*91.90*	*5.21*	*4.86*	*89.93*
CCSeg (our model)	***3.50***	***3.50***	***93.00***	***3.36***	***3.52***	***93.12***	***3.30***	***3.20***	***93.50***

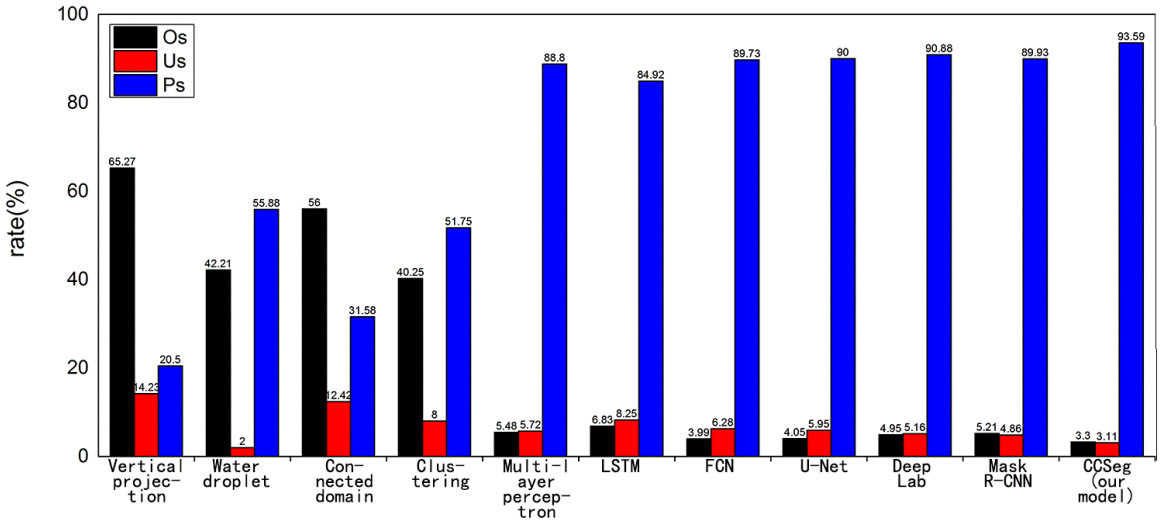

FIGURE 6. Results of comparative experiments for the condition of left-right architecture, which is normal in Chinese characters.

TABLE 9. Results of comparative experiments on CASIA-HWDB-MT.

Methods	O_S	U_S	P_S
Liu et al.[43]	*31.50*	*16.30*	*68.50*
Zhao et al.[46]	*67.04*	*13.56*	*19.40*
Xu L et al.[40]	*39.60*	*13.70*	*46.70*
Tian J et al.[39]	*39.33*	*18.87*	*41.80*
CCSeg (our model)	***21.30***	***7.50***	*71.20*

in the character region. In summary, the proposed method can deal with common and difficult character segmentation problems.

E. ANALYSES AND DISCUSSIONS

In this paper, the experimental results are evaluated by three metrics: ''Over-segmentation'' (Os), ''Under-segmentation''

(Us), and ''Positive-segmentation'' (Ps). Os means that more than one ''complete'' character is included in the segmentation

block. Us means that the segmentation block contains less than one ''complete'' character. Ps means that the segmentation block contains just one ''complete'' character. Here, *cIoU* is used to judge whether a segmented character is ''complete''.

cIoU is calculated as follows:

$$cIoU = \frac{S \cap \acute{E}C}{C} \tag{19}$$

TABLE 10. Effect of character spacing.

Character spacing	-10px	-5px	3px
Number of pictures	50	50	50
Number of characters	148	120	94
Positive segmentation	137	115	89
Os	1.35	1.67	5.32
Us	6.08	2.50	0.00
Ps	92.57	95.8	94.7

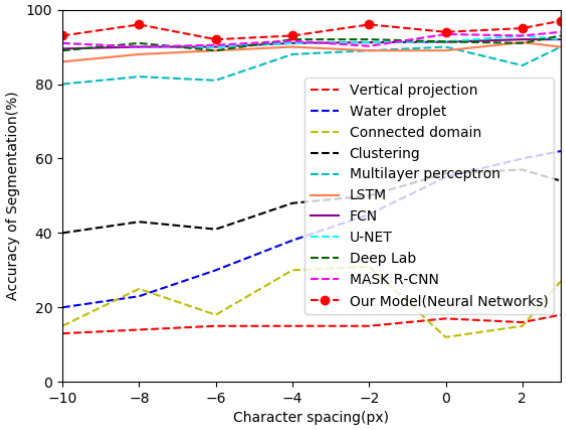

FIGURE 7. The rate of "Positive-segmentation" for 11 methods in the case of character spacing.

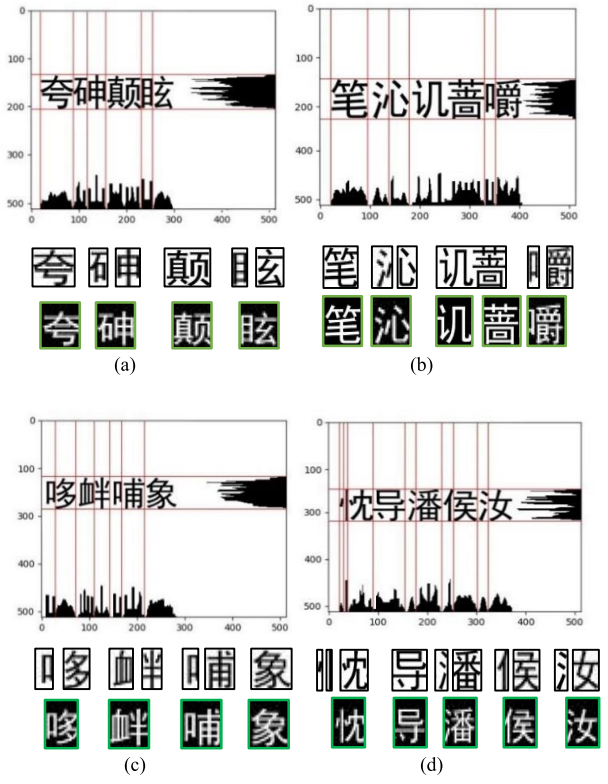

FIGURE 8. Comparison results. The upper rows of segmented blocks are obtained by projection. The other rows of them are obtained by the proposed method.

where S is the area of the segmented character block. C represents the area of the original character in the image. The threshold is set to 0.95. A segmentation block is considered to have a "complete" character if its $cIoU$ is greater than 0.95.

In the comparative experiment, 6000 images in the dataset are randomly chose for character segmentation. Results are shown in Table 8. Except for the proposed method, we still realize 10 methods based on the same dataset. Pictures in the dataset are divided into several categories: adhesion (C1), missing strokes (C2) and left-right architecture (C3). As can be observed, some early proposed methods like Vertical projection and Connected domain are extremely prone to "Over-segmentation". This is because both of them do not have a solution to characters with left-right architecture. Meanwhile, their "Under-segmentation" ratio is also higher than other methods. Compared to these two methods, Water droplet and Clustering take morphology features of characters into consideration. So they achieve better results. But they only optimize local features of characters. Thus they cannot get accurate results too. Multi-layer perceptron method was then proposed which differs from the morphological processing method in that it extracts low-level feature in the image. Semantic or instance segmentation methods like FCN, U-Net, Deep Lab and Mask R-CNN can make segmentation in pixel-level. These methods effectively reduce the ratio of "Over-segmentation" and "Under-segmentation". Similar to them, our proposed method uses information of original image for character segmentation. The most noticeable trend may originate from the bottom row of Table 8, which shows our proposed method performs well on all the three metrics. Especially for the condition of missing strokes and left-right architecture, even the semantic segmentation methods which are mentioned above cannot obtain good results. Figure 6 depicts the results of comparative experiments for the condition of left-right architecture. We compare the proposed method with other Chinese character segmentation methods on CASIA-HWDB-MT dataset. The results are shown in TABLE 9. Our method can obtain more accurate results than others.

In order to study the effect of character spacing on character segmentation, we conduct another experiment, as shown in Table 10. [−10px], [−5px] and [3px] are three conditions of character spacing. In the case of [−10px], CCSeg performs well despite the difficulty of extraction, and the rate of

positive-segmentation is 92.57%. When the character spacing is set to [3px], the larger character spacing may interfere with information extraction, resulting in a slight decrease in Ps to 94.7%. But it still maintains high accuracy. The bottom of Table 10 shows the proposed method can still perform well even existing long-range character spacing. Figure 7 depicts the rate of ''Positive-segmentation'' for 11 methods with character spacing.

In order to visually show the difference, Figure 8 illustrates character segmentation using the projection approach and CCSeg. There are projections in two directions– vertical and horizontal. Since the boundary of black and white pixel points is directly used as segmentation lines, characters with left-right structure will be over-segmented, such as ''砷'' in (a) and ''沁'' in (b). In another case, ''讥'' and ''蔷'' in (b) have adhesions in the vertical direction. As a result, they cannot be segmented correctly. With the regional multiple character feature, our method has obtained better results. As shown in the bottom of Figure 8, CCSeg outperforms the traditional one.

However, since the proposed method aims at optimizing the segmentation lines between characters, it cannot solve problems when there is large overlap between characters. This is also the direction we will keep working on.

VI. CONCLUSION

This paper proposes a character segmentation method called CCSeg to conduct Chinese character segmentation task. CCSeg consists of three major components. First, a convolutional network is used to extract spatial information. Next, CMFEN and CNFEN are used to extract character features to form a score map. At last, character features are reused to adjust the accurate segmentation points in the score map. A novel compound character feature is further proposed to describe the outline of characters. To effectively fine-tune our CCSeg, a multi-object dropout mechanism is also proposed. Based on the self-built dataset GBSD and CASIA-HWDB-MT(a dataset built based on CASIA-HWDB-1.0), we evaluate performance of CCSeg by focusing on Chinese character segmentation. Experimental results show that CCSeg effectively outperforms other methods in the literature. In the future, we will further extend our model to handle segmentation tasks when there are large overlaps between Chinese characters.

REFERENCES

[1] J. Liu, X. Zhang, Y. Li, J. Wang, and H.-J. Kim, ''Deep learning-based reasoning with multi-ontology for IoT applications,'' *IEEE Access*, vol. 7, pp. 124688–124701, 2019.
[2] G. Congedo, G. Dimauro, S. Impedovo, and G. Pirlo, ''Segmentation of numeric strings,'' in *Proc. 3rd Int. Conf. Document Anal. Recognit.*, Montreal, QC, Canada, 1995, pp. 1038–1041.
[3] U. Pal and S. Datta, ''Segmentation of Bangla unconstrained handwritten text,'' in *Proc. 7th Int. Conf. Document Anal. Recognit.*, 2003, pp. 1128–1132.
[4] V. Alabau, C.-D. Martínez-Hinarejos, V. Romero, and A.-L. Lagarda, ''An iterative multimodal framework for the transcription of handwritten historical documents,'' *Pattern Recognit. Lett.*, vol. 35, pp. 195–203, Jan. 2014.
[5] Z. Shi, S. Setlur, and V. Govindaraju, ''A steerable directional local profile technique for extraction of handwritten arabic text lines,'' in *Proc. 10th Int. Conf. Document Anal. Recognit.*, Barcelona, Spain, 2009, pp. 176–180.
[6] J.-C. Wu, J.-W. Hsieh, and Y.-S. Chen, ''Morphology-based text line extraction,'' *Mach. Vis. Appl.*, vol. 19, no. 3, pp. 195–207, May 2008.
[7] B. L. D. Bezerra, C. Zanchettin, and V. B. de Andrade, ''A hybrid RNN model for cursive offline handwriting recognition,'' in *Proc. Brazilian Symp. Neural Netw.*, Oct. 2012, pp. 113–118.
[8] S. Marinai and P. Nesi, ''Projection based segmentation of musical sheets,'' in *Proc. 5th Int. Conf. Document Anal. Recognit. (ICDAR)*, 1999, pp. 515–518.
[9] R. Manmatha and J. L. Rothfeder, ''A scale space approach for automatically segmenting words from historical handwritten documents,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1212–1225, Aug. 2005.
[10] U.-V. Marti and H. Bunke, ''The IAM-database: An english sentence database for offline handwriting recognition,'' *Int. J. Document Anal. Recognit.*, vol. 5, no. 1, pp. 39–46, Nov. 2002.
[11] B. Gatos, N. Stamatopoulos, and G. Louloudis, ''ICDAR2009 handwriting segmentation contest,'' in *Proc. IJDAR*, 2011, vol. 14, no. 1, pp. 25–33.
[12] J. Liu, C. Gu, J. Wang, G. Youn, and J.-U. Kim, ''Multi-scale multi-class conditional generative adversarial network for handwritten character generation,'' *J. Supercomput.*, vol. 75, no. 4, pp. 1922–1940, Apr. 2019.
[13] D. Impedovo and G. Pirlo, ''Dynamic handwriting analysis for the assessment of neurodegenerative diseases: A pattern recognition perspective,'' *IEEE Rev. Biomed. Eng.*, vol. 12, pp. 209–220, 2019.
[14] R. P. dos Santos, G. S. Clemente, T. I. Ren, and G. D. C. Cavalcanti, ''Text line segmentation based on morphology and histogram projection,'' in *Proc. 10th Int. Conf. Document Anal. Recognit.*, Jul. 2009, pp. 651–655.
[15] Y. LeCun, Y. Bengio, and G. Hinton, ''Deep learning,'' *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.
[16] B. Moysset, C. Kermorvant, C. Wolf, and J. Louradour, ''Paragraph text segmentation into lines with recurrent neural networks,'' in *Proc. 13th Int. Conf. Document Anal. Recognit. (ICDAR)*, Aug. 2015, pp. 456–460.
[17] B. Moysset, J. Louradour, C. Kermorvant, and C. Wolf, ''Learning text-line localization with shared and local regression neural networks,'' in *Proc. 15th Int. Conf. Frontiers Handwriting Recognit. (ICFHR)*, Oct. 2016, pp. 1–6.
[18] B. Moysset, P. Adam, C. Wolf, and J. Louradour, ''Space displacement localization neural networks to locate origin points of handwritten text lines in historical documents,'' in *Proc. 3rd Int. Workshop Historical Document Imag. Process. (HIP)*, 2015, pp. 1–8.
[19] B. Moysset, C. Kermorvant, and C. Wolf, ''Full-page text recognition: Learning where to start and when to stop,'' in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Nov. 2017, pp. 871–876.
[20] J. Long, E. Shelhamer, and T. Darrell, ''Fully convolutional networks for semantic segmentation,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.
[21] Y. Liu, Y. Luo, F. Liu, and Z. Qiu, ''A novel approach of segmenting touching and kerned characters,'' in *Proc. 8th Int. Conf. Neural Inf. Process.*, 2001, pp. 1603–1606.
[22] R. G. Casey and E. Lecolinet, ''A survey of methods and strategies in character segmentation,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 7, pp. 690–706, Jul. 1996.
[23] G. E. Hinton and R. R. Salakhutdinov, ''Reducing the dimensionality of data with neural networks,'' *Science*, vol. 313, no. 5876, pp. 504–507 2006.
[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ''ImageNet classification with deep convolutional neural networks,'' in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[25] M. D. Zeiler and R. Fergus, ''Visualizing and understanding convolutional networks,'' in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 818–833.
[26] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, ''Going deeper with convolutions,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
[27] K. He, X. Zhang, S. Ren, and J. Sun, ''Deep residual learning for image recognition,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[28] R. Girshick, J. Donahue, T. Darrell, and J. Malik, ''Rich feature hierarchies for accurate object detection and semantic segmentation,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.
[29] R. Girshick, ''Fast R-CNN,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Dec. 2015, pp. 1440–1448.

[30] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[31] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Conf. Comput. Vis.*, Oct. 2017, pp. 2980–2988.

[32] L. C. Chen, G. Papandreou, I. M. K. Kokkinos, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," in *Proc. Int. Conf. Learn. Representations (ICLR)*, San Diego, CA, USA, May 2015, pp. 1–14.

[33] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. S. Torr, "Conditional random fields as recurrent neural networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 529–1537.

[34] K. Kamnitsas, C. Ledig, V. F. J. Newcombe, J. P. Simpson, A. D. Kane, D. K. Menon, D. Rueckert, and B. Glocker, "Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation," *Med. Image Anal.*, vol. 36, pp. 61–78, Feb. 2017.

[35] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, Atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[36] L. O'Gorman, "The document spectrum for page layout analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 11, pp. 1162–1173, Nov. 1993.

[37] C. Rother, "GrabCut: Interactive foreground extraction using iterated graph cuts," in *Proc. SIGGRAPH*, 2004, p. 23.

[38] L. Xu, F. Yin, and C.-L. Liu, "Touching character splitting of Chinese handwriting using contour analysis and DTW," in *Proc. Chin. Conf. Pattern Recognit. (CCPR)*, Oct. 2010, pp. 814–818.

[39] J. Tian, R. Wang, G. Wang, J. Liu, and Y. Xia, "A two-stage character segmentation method for Chinese license plate," *Comput. Electr. Eng.*, vol. 46, pp. 539–553, Aug. 2015.

[40] L. Xu, F. Yin, Q.-F. Wang, and C.-L. Liu, "An over-segmentation method for single-touching Chinese handwriting with learning-based filtering," *Int. J. Document Anal. Recognit.*, vol. 17, pp. 91–104, Jun. 2014.

[41] K. He, G. Gkioxari, P. Dollár, and R. Girshick "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 386–397, Feb. 2020.

[42] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2015, pp. 234–241.

[43] C.-L. Liu, M. Koga, and H. Fujisawa, "Lexicon-driven segmentation and recognition of handwritten character strings for Japanese address reading," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 11, pp. 1425–1437, Nov. 2002.

[44] L. Xu, F. Yin, Q.-F. Wang, and C.-L. Liu, "Touching character separation in Chinese handwriting using visibility-based foreground analysis," in *Proc. 11th Int. Conf. Document Anal. Recognit.*, Sep. 2011, pp. 859–863.

[45] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[46] S. Zhao, Z. Chi, P. Shi, and H. Yan, "Two-stage segmentation of unconstrained handwritten Chinese characters," *Pattern Recognit.*, vol. 36, no. 1, pp. 145–156, Jan. 2003.

[47] C.-L. Liu, F. Yin, D.-H. Wang, and Q.-F. Wang, "CASIA online and offline Chinese handwriting databases," in *Proc. Int. Conf. Document Anal. Recognit.*, Beijing, China, Sep. 2011, pp. 37–41, doi: 10.1109/ICDAR.2011.17.

ZHENYU GAO received the B.S. degree in software engineering from Nantong University, Nantong, in 2018. She is currently pursuing the M.E. degree with Shanghai Maritime University. Her current research interests are deep learning, as well as object and text detection.

JIN LIU (Member, IEEE) received the B.S. degree from Lanzhou University, the M.S. degree from the University of Electrical Science and Technology of China, and the Ph.D. degree from Washington State University. He is currently a Professor at Shanghai Maritime University. His research interests include deep learning, nature language processing, and computer vision. He is a member of the CAAI and CCF.

YIYAO LI received the B.S. degree in computer science and technology from the Shanghai Lixin University of Accounting and Finance in 2019. She is currently pursuing the M.S. degree with Shanghai Maritime University. Her current research interests include deep learning and affective computing.

YIHE YANG received the B.E. degree from Shanghai Maritime University, in 2018, where he is currently pursuing the M.E. degree. His current research interests are data mining, natural language processing, and machine learning.

HUIHUA HE received the Ph.D. degree from Washington State University, in 2007. She is currently an Associate Professor with the College of Education, Shanghai Normal University. Her research interests include ICT uses in education, affective science, social-emotional learning, and related topics. She is a member of the CSE.