Recognition of Handwritten Chinese Text by Segmentation: A Segment-Annotation-Free Approach

Dezhi Peng, Lianwen Jin, Member, IEEE, Weihong Ma, Canyu Xie, Hesuo Zhang, Shenggao Zhu, and Jing Li

Abstract—Online and offline handwritten Chinese text recognition (HTCR) has been studied for decades. Early methods adopted oversegmentation-based strategies but suffered from low speed, insufficient accuracy, and high cost of character segmentation annotations. Recently, segmentation-free methods based on connectionist temporal classification (CTC) and attention mechanism, have dominated the field of HCTR. However, people actually read text character by character, especially for ideograms such as Chinese. This raises the question: are segmentation-free strategies really the best solution to HCTR? To explore this issue, we propose a new segmentation-based method for recognizing handwritten Chinese text that is implemented using a simple yet efficient fully convolutional network. A novel weakly supervised learning method is proposed to enable the network to be trained using only transcript annotations; thus, the expensive character segmentation annotations required by previous segmentation-based methods can be avoided. Owing to the lack of context modeling in fully convolutional networks, we propose a contextual regularization method to integrate contextual information into the network during the training stage, which can further improve the recognition performance. Extensive experiments conducted on four widely used benchmarks, including online and offline HCTR, show that our method significantly surpasses existing methods on both online and offline HCTR, and exhibits a considerably higher inference speed than CTC/attention-based approaches.

Index Terms—Handwritten Chinese text recognition, online and offline text recognition, segmentation-based text recognition, weakly supervised learning.

I. INTRODUCTION

HANDWRITTEN Chinese text recognition (HCTR), including online and offline HCTR, is a challenging research topic that has been intensively studied for decades. However, owing to the large vocabulary (tens of thousands of character categories), diverse writing styles, and the character-touching problem, satisfactory recognition performance has not yet been achieved.

Early methods of HCTR are based on oversegmentation [1]–[8], which first oversegment the text line and then search for the best segmentation-recognition path by integrating classifier outputs, geometric context, and linguistic context. Oversegmentation-based methods used to be the most successful approaches for HCTR; however, they can be easily affected by touching or overlapping characters and require annotations to provide the boundary of characters. Recently, segmentation-free methods [9]–[24] based on hidden Markov model (HMM), connectionist temporal classification (CTC) [25], and attention mechanism [26], have dominated the field of HCTR. They surpass oversegmentation-based methods and only require transcripts to be annotated. We further illustrate the performance of typical methods on the widely used offline subset of the ICDAR2013 competition dataset [4] in Fig. 1. Although the segmentation-based method achieved considerable improvement before 2016, segmentation-free methods have become mainstream. However, are segmentation-free approaches really the best solutions to the HCTR problem? Fig. 2 shows two example images of handwritten English text from the IAM dataset [27] and handwritten Chinese text from the SCUT-HCCDoc dataset [28]. In contrast to English texts, the basic elements of Chinese texts are characters rather than words. Intuitively, for a native Chinese speaker, each character is first
We formulate a new segmentation-based text recognition framework for HCTR which end-to-end segments and recognizes characters through a fully convolutional network. Compared with previous oversegmentation-based methods, the proposed framework is end-to-end trainable with high efficiency. Moreover, compared with most existing segmentation-free methods that utilize recurrent neural networks (RNNs) or auto-regressive models, our method runs in parallel and exhibits a higher inference speed. To summarize, the main contributions of this paper are as follows:

- We formulate a new segmentation-based text recognition framework for online and offline HCTR. During inference, the framework is implemented using a fully convolutional network, and thus exhibits high efficiency.
- We propose a weakly supervised learning method to enable the network to be trained using only transcript annotations, thereby greatly reducing the cost of manual annotations. However, character segmentation results can still be produced using our method.
- We design a contextual regularization method to integrate contextual information into the training of the fully convolutional network without slowing down the inference speed.
- Extensive experiments demonstrate the state-of-the-art performance of our method on both online and offline HCTR, in terms of both recognition accuracy and inference speed.

II. RELATED WORK

A. Offline Handwritten Chinese Text Recognition

Offline HCTR aims to transcribe text-line images into Chinese texts. In general, existing methods can be categorized into segmentation-based and segmentation-free approaches.

Most previous segmentation-based (also called explicit-segmentation) approaches adopt oversegmentation-based strategies. Specifically, these methods [1], [6] first obtain primitive segments through oversegmentation and then search for the best segmentation-recognition path. Although the character classifier, language model, and geometric model are integrated for path evaluation, satisfactory performance cannot be achieved, especially for touching or overlapping characters. Therefore, Wu et al. [7] explored neural network language models, yielding a significant improvement in the recognition performance. Furthermore, Wang et al. [8] proposed a weakly supervised method...
for string-level training of oversegmentation-based systems. In addition to the strategy using oversegmentation, Peng et al. [31] designed a fully convolutional network for end-to-end handwritten Chinese text segmentation and recognition.

Although enormous progress has been achieved by previous segmentation-based methods, most of them require expensive character segmentation annotations. To address this issue, segmentation-free (also called implicit-segmentation) approaches have recently received considerable interest. One category of segmentation-free methods [9]–[12] addresses the offline HCTR problem using hidden Markov model (HMM), where the handwritten text lines are modeled by a series of cascading HMMs based on the feature sequence extracted in a sliding-window manner. The method based on CTC [25] is another popular category of solutions to offline HCTR. Messina et al. [15] solved the offline HCTR problem by combining multi-dimensional long short-term memory (MDLSTM) and CTC. Wu et al. [17] further proposed separable MDLSTM to improve efficiency. Xie et al. [21] improved the CTC-based methods by exploring data augmentation and preprocessing. Liu et al. [65] explored residual and squeeze-and-excitation structures for feature extraction and proposed context beam search to integrate the Transformer-based [63] language model into CTC-based methods. The attention mechanism, which has been widely adopted in scene text recognition [30], [35], [67], action recognition [37], [38], and video processing [39], can also be applied to offline HCTR. Xiu et al. [40] improved the attention-based decoder by a multi-level multi-modal fusion network. In addition to the above methods that adopt a single strategy, Zhu et al. [22] proposed a convolutional combination strategy to combine the segmentation-based and segmentation-free approaches for better performance.

### III. METHODOLOGY

#### A. Overview

The overall design of the proposed method is illustrated in Fig. 4. First, a new segmentation-based handwritten Chinese text recognition framework that segments and recognizes characters is formulated in an end-to-end manner and is implemented using an efficient fully convolutional network during inference. Second, a novel weakly supervised learning method is proposed to effectively train the model using only transcript annotations, which avoids costly manual segmentation annotations. Nevertheless, our model can still accurately output character segmentation results. Third, as the fully convolutional architecture cannot capture contextual dependencies, a new contextual regularization (ConR) is proposed to integrate contextual information into the model during the training stage. The proposed ConR can improve the performance without reducing the inference speed.

#### B. Segmentation-Based Text Recognition

Online HCTR is aimed at recognizing Chinese text from pen-tip trajectories. In contrast to offline HCTR, the input of online HCTR contains sequential information and does not have background noise. However, the acquisition of pen-tip trajectories requires specific hardware. Owing to the rise of pen-based devices, online HCTR is also a very important research topic with wide applications in many fields such as finance and education.

The oversegmentation-based strategy can also be used to address online HCTR [2], [3], [5], following the same pipeline as for offline HCTR. However, these methods rely heavily on the evaluation of candidate segmentation-recognition paths and have difficulty in recognizing touching or overlapping characters. Therefore, some studies [13], [19], [41] proposed to directly handle the pen-tip trajectory using RNNs and CTC. These methods extract features using long short-term memory (LSTM) or gated recurrent unit (GRU) and optimize the model with CTC loss. Liu et al. [19] further proposed distilling GRU to accelerate model training and handle handwritten texts with various styles. Instead of using RNNs, Peng et al. [64] designed a global and local relation network (GLRNet) that uses self-attention [63] for feature extraction and jointly trained the combination of GLRNet and a Transformer-based language model to achieve optimal overall performance. With the prevalence of convolutional neural networks (CNNs), there exist methods [16], [18], [32] that transform the online pen-tip trajectory into offline feature maps that CNNs can process. Such methods first obtain image-like representations through path signature or eight-directional feature maps. Thereafter, an integrated CNN-LSTM network is trained using CTC loss. Xie et al. [32] further improved this pipeline using an implicit language model and multi-spatial context.
Character in the $n$-th region is classified as each of the $n_{cls}$ categories.

We implement this framework using a fully convolutional network during inference. The detailed network architecture is illustrated in Fig. 5. Inspired by ResNet [42], the residual connection is adopted in the ConvBlock. Instead of extracting a single feature map $f_{enc}$, the encoder outputs three feature maps, $f_{loc}$, $f_{bbox}$, and $f_{cls}$, to predict the character location, character bounding boxes, and character classification, respectively.

During inference, after removing redundant predictions using non-maximum suppression (NMS) [43] and sorting the remaining predictions from left to right, we can obtain segmentation and recognition results as shown in the “Prediction” part of Fig. 4(c).

Specifically, the network is first pretrained using synthetic data with segmentation annotations. As described in Section IV-B, the samples for different scenarios are synthesized in the same manner, by placing isolated characters on a white background without any complicated synthesis techniques. Although the synthetic data may differ significantly from the text lines from real scenes, the model can still learn the fundamental ability to localize and recognize characters.

Then, the real data with only transcript annotations is also used to train the model. The procedure of the proposed weakly supervised learning method for real samples is shown in Fig. 4(c). Similar to the learning process of humans, the model is taught which prediction is correct and trained based on past successful experiences. Specifically, for a real input, the network predicts the segmentation result $R_{seg}$ and recognition result $R_{rec}$ as

$$R_{seg} = \{ (r_{1,seg}, r_{1,scro}), (r_{2,seg}, r_{2,scro}), \ldots, (r_{lpr,seg}, r_{lpr,scro}) \},$$

$$R_{rec} = \{ (r_{1,rec}, r_{1,عروض}), r_{1,عروض}, \ldots, r_{lpr,rec} \},$$

where $r_{i,seg}$, $r_{i,scro}$, and $r_{i,عروض}$ are the coordinates of the bounding box, score, and category of the $i$-th predicted character, respectively, and $l_{pr}$ is the total number of predicted characters.
Thereafter, we compute the edit distance between transcript annotation $S_{gt}$ and recognition result $R_{rec}$, where $S_{gt}$ is defined as

$$S_{gt} = \{s^1_{gt}, s^2_{gt}, \ldots, s^l_{gt}\},$$

where $s^j_{gt}$ is the category of the $j$-th character in the transcript, and $l_{gt}$ is the total number of annotated characters. The characters, which are matched as “equal” in computing edit distance, are marked using red arrows between the annotation and prediction in Fig. 4(c). Because character recognition and segmentation tasks are highly related, the “equal” characters in the prediction are very likely to have accurate bounding box predictions.

Next, the bounding boxes corresponding to “equal” characters (red bounding boxes in Fig. 4(c)) are used to update the pseudo bounding boxes. The pseudo bounding boxes $B_{pse}$ are defined as follows:

$$B_{pse} = \{(b^j_{pse}, b^i_{sco}), (b^2_{pse}, b^2_{sco}), \ldots, (b^l_{pse}, b^l_{sco})\},$$

where $b^j_{pse}$ and $b^i_{sco}$ are the coordinates and score of the pseudo bounding box of character $s^j_{gt}$, respectively. All the pseudo bounding boxes are initialized as $\emptyset$. If the character $r^i_{rec}$ is matched to be “equal” to the character $s^j_{gt}$, the segmentation result $r^i_{seg}$ is used to update the pseudo bounding box as follows:

$$b^j_{pse} = \begin{cases} r^i_{seg}, & b^j_{pse} = \emptyset, \\ \lambda_{pse} \cdot b^j_{pse} + (1 - \lambda_{pse}) \cdot r^i_{seg}, & \text{otherwise}, \end{cases}$$

$$b^i_{sco} = \begin{cases} r^i_{sco}, & b^i_{sco} = \emptyset, \\ \lambda_{pse} \cdot b^i_{sco} + (1 - \lambda_{pse}) \cdot r^i_{sco}, & \text{otherwise}, \end{cases}$$

where $\lambda_{pse}$ is calculated as:

$$\lambda_{pse} = \frac{e^{10 \cdot b^i_{sco}}}{e^{10 \cdot b^i_{sco}} + e^{10 \cdot r^i_{sco}}}.$$ 

The weight $\lambda_{pse}$ is a function of $b^i_{sco}$ and $r^i_{sco}$, so as to make the bounding box with a higher score have a much higher weight.

Subsequently, the pseudo bounding boxes $B_{pse}$ and transcript annotation $S_{gt}$ are used to optimize the network. However, there may exist pseudo bounding boxes equal to $\emptyset$, which implies that the loss cannot be computed in a normal manner. Therefore, a special method for loss calculation is designed.

Specifically, we first project the pseudo bounding boxes that are not equal to $\emptyset$ onto their corresponding regions of the model input, yielding a mapping $M_{ptr}$. The element $(j, n) \in M_{ptr}$ indicates that the center point of the pseudo bounding box $b^j_{pse}$ is within the $n$-th region. The regions that contain the center points of pseudo bounding boxes are represented by the blue squares in the “Regions” part of Fig. 4(c). Then the loss of $p_{bbox}$ and $p_{cls}$ are calculated as:

$$l_{bbox} = \frac{1}{|M_{ptr}|} \sum_{(j, n) \in M_{ptr}} SE(b^j_{pse}, p^n_{bbox}),$$

$$l_{cls} = -\frac{1}{|M_{ptr}|} \sum_{(j, n) \in M_{ptr}} \log(p^n_{cls}(y^n_{bbox})),$$

where the function $SE$ calculates the square error between two inputs.

Regarding the loss of character location $p_{loc}$, we only know that the blue regions in Fig. 4(c) contain characters. Owing to the existence of pseudo bounding boxes equal to $\emptyset$, the difficulty lies in determining the negative samples, i.e., the regions that do not contain characters. Fortunately, although we cannot find all the negative samples, it can be confirmed that there is no character in the regions between two consecutive pseudo bounding boxes. Specifically, if both $b^j_{pse}$ and $b^{j+1}_{pse}$ are not equal to $\emptyset$, the indices of the regions, which are between the two regions corresponding to $b^j_{pse}$ and $b^{j+1}_{pse}$, are added to the set $N_{loc}$. The regions in $N_{loc}$ contain no characters and are represented by the green squares in Fig. 4(c). However, whether the yellow regions contain characters cannot be determined because of missing pseudo bounding boxes. Thus, these regions are not considered in the loss calculation. Consequently, the loss of $p_{loc}$ is formulated as:

$$l_{loc} = -\frac{0.5}{|T_{loc}|} \sum_{n \in T_{loc}} \log(p^n_{loc}) - \frac{0.5}{|N_{loc}|} \sum_{n \in N_{loc}} \log(1 - p^n_{loc}),$$

where $T_{loc}$ is the set of the indices of the blue regions that are supposed to contain characters.

Finally, the total loss is given by:

$$l_{total} = l_{bbox} + l_{cls} + l_{loc}.$$ 

D. Contextual Regularization

The fully convolutional architecture results in high efficiency and parallel computing, but it is difficult to capture contextual dependencies. The prediction for each region of a text line depends only on the corresponding receptive field. For example, the character classification $p_{cls}$ can be reformulated as:

$$p_{cls} = \{p^1_{cls}, \ldots, p^{w_{enc}}_{cls}\} = FCN(I_n) = \{FCN(F^1), \ldots, FCN(F^{w_{enc}})\},$$

where $FCN$ is our network and $F^n$ is the receptive field of the $n$-th character classification prediction. In the above formula, the predictions in $p_{cls}$ are independent without considering the relationship between them.

However, the task of HCTR, especially the character classification part, relies heavily on contextual information. Even for native speakers, it is difficult to recognize some confusing curvaceous handwritten characters without context. Previous methods adopted recurrent layers, such as LSTM and GRU, for context modeling. However, these recurrent layers can not run in parallel, which greatly reduces the inference speed especially when handling long texts. To this end, we propose to solve context modeling from a new perspective, by guiding the feature extraction using a novel contextual regularization (ConR) only in the training stage, as illustrated in Fig. 6.

During training, two bidirectional long short-term memory (BLSTM) layers are added on top of the feature map $f_{cls}$. Subsequently, an extra character classification is predicted based on
the output of the BLSTM layers. The loss $l_{conr}$ of this new character classification is calculated in the same way as for $l_{cls}$ (11). Then, the loss $l_{conr}$ is added to the total loss $l_{total}$ as a regularization term. Through the backpropagation of the gradient, ConR can guide the feature $f_{cls}$ to capture contextual information with the help of the context modeling ability of BLSTM layers.

During inference, the BLSTM layers and subsequent character classification are removed to maintain the high processing speed of the fully convolutional architecture. Experiments show that ConR results in a consistent improvement on all datasets. Even if the BLSTM layers and subsequent character classification are used during inference, the performance remains nearly unchanged, which proves that the feature map $f_{cls}$ really learns contextual information.

IV. EXPERIMENTS

A. Datasets

CASIA-HWDB [33] is a large-scale offline handwriting database, including CASIA-HWDB1.0-1.2 and CASIA-HWDB2.0-2.2. CASIA-HWDB1.0-1.2 contain 3,895,135 isolated characters from 1,020 writers, while CASIA-HWDB2.0-2.2 include 52,230 text lines from 1,019 writers. Note that the isolated character samples of CASIA-HWDB1.0-1.2 are not the characters cropped from the text lines of CASIA-HWDB2.0-2.2.

CASIA-OLHWDB [33] is the online version of CASIA-HWDB, which consists of CASIA-OLHWDB1.0-1.2 and CASIA-OLHWDB2.0-2.2. CASIA-OLHWDB1.0-1.2 contain 3,912,017 isolated characters from 1,020 writers while CASIA-OLHWDB2.0-2.2 contain 52,220 text lines from 1,019 writers. The CASIA-OLHWDB2.0-2.2 are further divided into 41,710 text lines for training and 10,510 text lines for testing.

ICDAR2013 competition dataset [4] contains 3,432 online and offline handwritten Chinese text lines from 60 writers. For convenience, the online and offline subsets are denoted as ICDAR2013-Online and ICDAR2013-Offline, respectively.

SCUT-HCCDoc [28] contains 12,253 offline camera-captured document images with 116,629 text lines. The training and testing sets comprise 93,411 text lines and 23,218 text lines, respectively.

B. Data Synthesis

For all real datasets, synthetic text lines are synthesized by simply placing offline characters on a white background or concatenating the pen-tip trajectories of online characters. The synthetic samples for CASIA-HWDB and CASIA-OLHWDB use isolated characters from CASIA-HWDB1.0-1.2 and CASIA-OLHWDB1.0-1.2, respectively, while the synthetic samples for SCUT-HCCDoc use the character images generated from 101 font files. The categories of characters are randomly selected from the vocabulary when synthesizing data without semantic information, while the corpora described in Section IV-D are adopted when synthesizing data with semantic information. Fig. 7 shows examples of synthetic samples (without semantic information) for CASIA-HWDB, CASIA-OLHWDB, and SCUT-HCCDoc.

C. Implementation Details

We conduct our experiments using an NVIDIA GTX 1080ti GPU with 11 GB of memory and implement our method using PyTorch. First, in the pretraining stage, the model is pretrained using synthetic data for 150,000 iterations. The weakly supervised learning method and ConR are not adopted in this stage. Then, in the training stage, the model is trained using both real and synthetic data for 1,200,000 iterations. The synthetic data is synthesized on the fly. The network is optimized using stochastic gradient descent (SGD) with a batch size of 8 and an initial learning rate of 0.01. The learning rate is multiplied by 0.1 at 25%, 50%, and 75% of the total number of iterations. During inference, the batch size is set to 1.

D. Transcription

During inference, the transcription without language model is described in the last paragraph of Section III-B, which first removes redundant predictions through NMS and then rearranges the remaining characters from left to right. In the following sections, if not specified, the results are obtained by the transcription without language model.

Moreover, the transcription process can be combined with n-gram language models. Specifically, a tri-gram language model generated from the same corpora as the conference version [31] is adopted. The corpora consist of the PFR corpus [48] (news text of 2,199,492 characters from the 1998 People’s Daily corpus),
the PH corpus [49] (news text of 3,697,028 characters from the People’s Republic of China’s Xinhua news recorded between January 1990 and March 1991), and the CLDC corpus [50] (50 million characters collected by the Institute of Applied Linguistics). Because the character classification prediction $p_{cls}$ contains the probability of each character category and the blank probability can be calculated as $1 - p_{loc}$, the CTC beam search algorithm [51] can be used for the transcription with the tri-gram language model based on the CTC-style predictions formed by $p_{cls}$ and $1 - p_{loc}$.

Recently, the Transformer-based [63] language model has emerged in the field of HCTR [40], [64], [65]. After preparing CTC-style predictions as mentioned above, the transcription process can also be integrated with the Transformer-based language model through the context beam search algorithm [65]. The Transformer-based language model follows the architecture specified in [65] and is trained with the same corpora as the tri-gram language model using the Fairseq [66] toolkit.

### E. Evaluation Metrics

Following previous studies on online and offline HCTR, the accurate rate (AR) and correct rate (CR) are adopted to evaluate the performance of methods, which are calculated as

$$AR = \frac{(N_t - D_e - S_e - I_e)}{N_t},$$

$$CR = \frac{(N_t - D_e - S_e)}{N_t},$$

(15)

where $D_e$, $S_e$, and $I_e$ represent the total number of deletion, substitution, and insertion errors, respectively, and $N_t$ is the total number of characters in the annotations. The errors are calculated between the recognition result and transcript annotation. The AR and CR are presented as percentages in the tables of the following sections.

### F. Experiments on ICDAR2013-Offline

1) **Experimental Settings**: The model is first pretrained using the synthetic data (without semantic information) for CASIA-HWDB. Then, the 52,230 real samples from CASIA-HWDB2.0-2.2 and the synthetic data (with semantic information) for CASIA-HWDB are used to train the model. Both the ratios of the real and synthetic samples in a batch are 0.5. Finally, the model is evaluated on the 3,432 samples from ICDAR2013-Offline. Following the setting of most previous methods, the number of character categories is set to 7,356.

2) **Data Preprocessing**: For the text-line images from CASIA-HWDB2.0-2.2 and ICDAR2013-Offline, the data preprocessing is illustrated in Fig. 8. Given a text-line image (Fig. 8(a)), we first estimate the tilt angle of the text by conducting linear regression using the coordinates of black pixels (Fig. 8(b)). Then, the text-line image is rotated to make the text horizontal (Fig. 8(c)). Next, we remove the white padding at the top and bottom of the image to highlight the text (Fig. 8(d)). Finally, the height of the text-line image is normalized to 128 pixels while maintaining the aspect ratio (Fig. 8(e)).

For the synthetic text-line images, only size normalization (Fig. 8(e)) is performed.

3) **Experimental Results**: Table I shows the comparison of existing methods and ours on ICDAR2013-Offline. It can be seen that our approach achieves state-of-the-art performance with and without language model. Specifically, taking advantage of the Transformer-based language model, the method in [65] outperforms our approach that uses the traditional tri-gram language model. However, state-of-the-art performance can still be achieved when our method is also equipped with the Transformer-based language model.

Moreover, in addition to the outstanding recognition performance, our method can also output character segmentation results. Fig. 9 shows the visualization results of ICDAR2013-Offline. It can be seen that the characters can be segmented accurately from the text-line image. Particularly in the fourth example, the character “清” is written very similar to “消” but is still correctly recognized (marked by blue square), demonstrating the ability to distinguish similar characters. This ability may be attributed to the contextual information integrated by ConR and the large-scale synthetic datasets. Nonetheless, there are still some misrecognized similar characters, e.g., “张” is recognized as “张” in the second example, indicating the recognition of similar handwritten Chinese characters is still an important issue worth studying in the future.
G. Experiments on ICDAR2013-Online

1) Experimental Settings: First, the synthetic data (without semantic information) for CASIA-OLHWDB is used to pre-train the model. Then, the model is trained using the 41,710 real samples from the training set of CASIA-OLHWDB2.0-2.2 and the synthetic data (with semantic information) for CASIA-OLHWDB. Both the ratios of the real and synthetic samples in a batch are 0.5. After training, the 3,432 samples from ICDAR2013-Online are used for evaluation. Following previous studies, the number of character categories is set to 7,356.

2) Data Preprocessing: The pipeline of data preprocessing for pen-tip trajectories from CASIA-OLHWDB2.0-2.2 and ICDAR2013-Online is depicted in Fig. 10. Given a pen-tip trajectory containing a sequence of points belonging to multiple strokes (Fig. 10(a)), we first perform linear fitting based on the \((x, y)\) coordinates of the points and obtain the tilt angle of the text line, as shown in Fig. 10(b). Then, the text line is rotated to be horizontal, as depicted in Fig. 10(c). Next, the \((x, y)\) coordinates of the points are rescaled to normalize the height of the text line to 128 while maintaining the aspect ratios, as shown in Fig. 10(d). Because the raw data adopts uniform-time sampling, the density of the points is related to the writing speed and sampling rate. Therefore, we resample the trajectory in an equal-distance manner as shown in Fig. 10(e). Specifically, the points are sampled at intervals of a Euclidean distance of 1.

3) Path Signature: For online HCTR, it is crucial to translate the pen-tip trajectory into offline feature maps while retaining most of the online information. The path signature [52]–[55] has been verified to be effective in both online Chinese character recognition [56]–[58] and online HCTR [16], [18], [32]. Therefore, following previous methods [32], the truncated path signature feature maps up to 2nd integrated integral are calculated in a sliding-window fashion with a window size of 9. Fig. 11 visualizes the generated path signature feature maps.

4) Experimental Results: In Table II, we compare our methods with existing approaches on ICDAR2013-Online. When language models are not used, the performance of our method is comparable to state-of-the-art performance. Although the method proposed by [64] performs slightly better, the advantage of our method is that it can produce character segmentation results. When using language models, our approach equipped with the Transformer-based language model outperforms existing methods, including the method in [64] which also adopts a Transformer-based language model.

| Method | Without LM | With LM |
|--------|------------|---------|
|        | AR | CR | AR | CR |
| Zhou et al. [2] | - | - | 94.06 | 94.76 |
| Zhou et al. [5] | - | - | 94.22 | 94.76 |
| Sun et al. [59] | 89.12 | 90.18 | 93.40 | 94.43 |
| 2C-FCRN+impLM [32] | 88.88 | 90.17 | 95.46 | 96.01 |
| 2C-FCRN+impLM&staLM [32] | 88.88 | 90.17 | 96.06 | 96.58 |
| VGG-DBLSTM [18] | 87.49 | 87.98 | 97.03 | 97.29 |
| CharNet: DBLSTM [18] | 87.10 | 87.71 | 96.87 | 97.15 |
| Liu et al. [19] | 91.36 | 92.37 | 94.89 | 95.70 |
| Peng et al. [64] | 95.05 | 95.46 | 97.36 | 97.63 |
| Ours (tri-gram LM) | 94.46 | 94.67 | 96.64 | 97.28 |
| Ours (Transformer-based LM) | 94.46 | 94.67 | 97.89 | 98.06 |

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Fig. 12. Visualization results of ICDAR2013-Online. The recognition results without language model are presented.

Table III

| Method            | AR  | CR  |
|-------------------|-----|-----|
| CTC-based [14]    | 87.46 | 88.83 |
| Attention-based [60] | 83.30 | 84.81 |
| DAN [30]          | 83.53 | 85.41 |
| Lin et al. [65]   | 89.06 | 90.12 |
| Ours              | 98.71 | 92.01 |

Fig. 12 illustrates the visualization results of ICDAR2013-Online. According to the predicted bounding boxes, we can divide the pen-tip trajectory into several segments, each of which corresponds to one character. Specifically, the points within a bounding box belong to the corresponding character. For the points that are not within any bounding boxes or are within multiple bounding boxes, they belong to the character corresponding to the closest bounding box. As shown in Fig. 12, both the segmentation and recognition results are very accurate.

H. Experiments on SCUT-HCCDoc

1) Experimental Settings: First, the model is pretrained using the synthetic data (without semantic information) for SCUT-HCCDoc. Then, both the 93,411 real samples from the training set of SCUT-HCCDoc and the synthetic samples (without semantic information) for SCUT-HCCDoc are utilized to train the model. Because the vocabularies of SCUT-HCCDoc and the corpora described in Section IV-D are very different, the synthetic samples during the training stage are also synthesized without semantic information. Moreover, the ratios of the real and synthetic samples in a batch are 0.7 and 0.3, respectively. After the model training, the 23,218 samples from the testing set of SCUT-HCCDoc are adopted to evaluate the method. The number of character categories is set to 6,109.

2) Data Preprocessing: Both the real and synthetic text-line images are resized to a height of 128 pixels while maintaining their aspect ratios.

3) Experimental Results: In Table III, we compare our method with existing approaches on SCUT-HCCDoc. Specifically, the latest results of the CTC/attention-based approaches [14], [60], which were updated by the authors of [28] at their website, are presented in Table III. The performances of the other two methods [30], [65] are obtained from our reimplementation based on their official codes. It can be observed that our method achieves state-of-the-art performance on SCUT-HCCDoc with an AR of 90.71% and a CR of 92.01%.

The visualization results of SCUT-HCCDoc are shown in Fig. 13. Although the synthetic samples for SCUT-HCCDoc are composed of simple characters generated from font files and white backgrounds as illustrated in Fig. 7(c), our method can still make full use of them and learn to segment and recognize characters of complex real samples. From the visualizations in Fig. 13, we can observe that our method can handle various writing styles including illegible handwriting and is unaffected by the noises such as illumination and the interference from the background. Especially for the sample at the top-right of Fig. 13, where the text is crossed out, the characters can still be successfully segmented and recognized, which verifies the robustness of our approach.

I. Ablation Studies

In Table IV, we conduct ablation analysis to demonstrate the effectiveness of each component of our method.

The baseline method represents the pretrained model that uses only synthetic data. It can be seen that training the network using only synthetic data leads to very poor performance. Especially for SCUT-HCCDoc where the real images are very different from the synthetic samples, the baseline can only achieve an extremely low accuracy with an AR of 1.14% and a CR of 1.25%.

When the proposed weakly supervised learning method is adopted, real samples with only transcript annotations can also be utilized to train the model. Taking advantage of the effective design of the weakly supervised learning method, the model can make full use of the general knowledge acquired from simple synthetic data and be adapted to handle the real sample by learning from its past successful predictions. Table IV shows that performance can be significantly improved. Even though the pretrained model for SCUT-HCCDoc has extremely poor performance (Online), the model can still achieve state-of-the-art results with an AR of 90.71% and a CR of 92.01%.

Fig. 13. Visualization results of SCUT-HCCDoc.

Table IV

| Method                | AR  | CR  |
|-----------------------|-----|-----|
| Baseline (synthetic)  | 1.14 | 1.25 |
| Weakly supervised     | 90.71 | 92.01 |

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
performance, our weakly supervised learning method can also work very well. ConR is aimed at integrating contextual information into the feature maps for character classification. The results in Table IV demonstrate that ConR can bring remarkable improvement.

### J. Effectiveness of Weakly Supervised Learning

We further analyze the effectiveness of the proposed weakly supervised learning method. In our method, we distinguish the correct prediction by computing the edit distance between the recognition result and transcript annotation. Moreover, the pseudo bounding boxes are updated in a weighted-sum manner, considering all the correct predictions in previous iterations and their scores.

However, some existing approaches [61], [62] that involve weakly supervised learning follow a very simple pipeline. If the lengths of the recognition result and transcript annotation are equal, the predicted bounding boxes are directly viewed as the ground truth. Following their ideas, we replace our weakly supervised learning method with a new one named Text Length, where the pseudo bounding boxes $B_{pse}$ are updated as

$$B_{pse} = \begin{cases} R_{seg}, & l_{pr} = l_{gt}, \\ B_{pse}, & \text{otherwise}, \end{cases}$$

(16)

where $R_{seg}, l_{pr},$ and $l_{gt}$ are the segmentation result, the length of the recognition result, and the length of the transcript annotation, respectively (as specified in Section III-C).

As presented in Table V, our method outperforms the Text length by a large margin, especially for ICDAR2013-Online and ICDAR2013-Offline that contain long texts. All models in Table V are trained without contextual regularization. The criterion of Text length (i.e., the length of predictions and annotations are equal) cannot guarantee the quality of the predictions used for updating the pseudo bounding boxes, and is rarely satisfied for long texts and pretrained models with low accuracy. Moreover, the pseudo bounding boxes are updated by copying new predictions, which may make the training easily be interfered with by poor predictions. However, our method is based on the observation that correctly recognized characters are likely to have accurate bounding boxes. Thus only the characters that are matched as “equal” in computing edit distance are selected to ensure the quality of bounding boxes. Furthermore, the updating of pseudo bounding boxes is also carefully designed to suppress the impact of potential poor predictions.

### K. Effectiveness of Contextual Regularization

In Table VII, experiments are conducted for further analysis of ConR.
First, we investigate the performance of the model if the BLSTM layers and the subsequent character classification results are adopted during inference, instead of using the character classification results from feature map \( f_{cls} \). As shown in Table VII, the performances with and without the BLSTM layers are nearly the same. Specifically, compared with the counterpart with BLSTM layers, the AR and CR without BLSTM layers can be slightly higher on ICDAR2013-Online and ICDAR2013-Offline, and only drop a little on SCUT-HCCDoc. Based on the above results, we can conclude that ConR can really integrate the contextual information into the feature \( f_{cls} \) before BLSTM layers.

As for the inference speed, the two BLSTM layers can consume more than 40% of the inference time of the entire network if they are used for context modeling, as demonstrated by the “Ratio” column of Table VII. This is attributed to the non-parallel running of the BLSTM layer and the long texts that are common in Chinese documents. With the help of ConR, the recurrent layers are not required during inference, thus significantly improving the speed.

### L. Comparison With CTC and Attention

Table VIII compares our method with existing widely used CTC/attention-based approaches.

| Dataset          | Method                  | AR       | CR       | Speed  |
|------------------|-------------------------|----------|----------|--------|
| ICDAR2013-Online | CTC-based [19]          | 91.36    | 92.37    | 65fps  |
|                  | Attention-based [36]    | 85.35    | 85.84    | 16fps  |
|                  | Ours                    | 94.46    | 94.67    | 70fps  |
| ICDAR2013-Offline| CTC-based [21]          | 91.53    | 92.13    | 64fps  |
|                  | Attention-based [36]    | 84.79    | 85.90    | 16fps  |
|                  | Ours                    | 94.50    | 94.70    | 70fps  |
| SCUT-HCCDoc      | CTC-based [14]          | 87.64    | 88.83    | 75fps  |
|                  | Attention-based [60]    | 83.30    | 84.81    | 52fps  |
|                  | Ours                    | 90.71    | 92.01    | 97fps  |

The results in Table VIII show that our method outperforms existing CTC/attention-based approaches in terms of AR and CR. Moreover, owing to the parallel computing characteristic of the fully convolutional architecture, our method exhibits a higher inference speed compared with the CTC/attention-based methods that use recurrent layers for contextual modeling. In addition, our method can produce character segmentation results, whereas CTC/attention-based approaches cannot.

### M. Reading Chinese Text in the Wild

In this section, we further explore the potential of our method to be extended to the recognition of Chinese text in the wild. The experiments are conducted using ReCTS-25k [34], which contains 25,000 signboard images (20,000 for training and 5,000 for testing). There are 108,963 and 10,789 text lines cropped from the training and testing sets, respectively. We further divide the text lines from the training set into 90,763 samples for training and 18,200 samples for validating. The synthetic data is synthesized in the same way as in Section IV-B, using white background and characters from font files. The number of character categories is 4,134. Other details follow the experiments on SCUT-HCCDoc and Section IV-C.

As shown in Table IX, our method achieves a normalized edit distance (NED) [34] of 86.66% on the validating set and 90.70% on the testing set. Some visualization results are shown in Fig. 14. When tested using an NVIDIA GTX 1080ti GPU, our method performs better with a higher inference speed. However, our approach is inferior to the winning method (SNAHL_v1 with 95.55% NED) of the competition [34], which uses an ensemble of three types of models and a large amount of external data. To conclude, our method has the potential of being extended to other applications besides online and offline HCTR.

### N. Error Analysis

The weakness of our method lies in the segmentation of punctuations. Fig. 15 presents the failure cases of punctuation segmentation (indicated by red arrows). Compared with Chinese characters, punctuations are much smaller and have more flexible locations that could be the top or bottom of the text line.
Our method is basically for offline text recognition because online texts are also expressed as offline representations. In previous literature, there are few studies which translate the online pen-tip trajectories into image-like representations. Thus, we illustrate the AR versus parameter size on ICDAR2013-Offline.

However, punctuations are overwhelmed by Chinese characters in the training data, which may be the major cause of the poor segmentation of punctuations. Nevertheless, owing to the decoupled design of character segmentation and classification, the recognition results can still be correctly predicted in the first two cases of Fig. 15. Moreover, the punctuation could be very close to its previous character especially for handwritten texts, which makes it difficult to distinguish. For example, the comma in the ground truth consists of multiple same characters, such as “AAA,” but only “AA” is recognized, we can not determine which two “A”s in the ground truth are correctly recognized by calculating edit distance. This issue could lead to inaccurate pseudo bounding boxes because they may be updated using unsuitable predictions. However, such a situation rarely occurs in real data that contains natural texts. Thus, the model training is almost unaffected. Nevertheless, this problem can be explored by incorporating spatial constraints or feature similarities in the future.

O. Discussion

1) Parameter Size: Our method is basically for offline text recognition because online texts are also expressed as offline representations. Thus, we illustrate the AR versus parameter size of existing methods on ICDAR2013-Offline in Fig. 16. Note that only methods whose parameter sizes were reported in previous literature are presented. It can be observed that our model has a good trade-off between AR and parameter size. Especially compared with the method in [65], our method achieves better performance with approximately half of the parameters.

2) Training Time: As described in Section IV-C, there are two stages of our model training, i.e., the pretraining and training stages. Using an NVIDIA RTX 2080ti GPU, it takes approximately 125 hours in total to train a model for ICDAR2013-Offline. Specifically, the pretraining and training stages take 14 hours and 111 hours, respectively. The time required to train a model for ICDAR2013-Online is nearly the same because we translate the online pen-tip trajectories into image-like representations. In previous literature, there are few studies which reported their training time. For example, the method in [65] for ICDAR2013-Offline requires approximately 80 hours, and the method in [19] for ICDAR2013-Online requires 102 hours. Compared with these methods, the training time of our method is slightly longer but still acceptable. Because most procedures of the weakly supervised learning run on CPU in our implementation, the training speed could be accelerated by migrating them to GPU in the future.

3) Failure Case of Weakly Supervised Learning: If the ground truth consists of multiple same characters, such as “AAA,” but only “AA” is recognized, we can not determine which two “A”s in the ground truth are correctly recognized by calculating edit distance. This issue could lead to inaccurate pseudo bounding boxes because they may be updated using unsuitable predictions. However, such a situation rarely occurs in real data that contains natural texts. Thus, the model training is almost unaffected. Nevertheless, this problem can be explored by incorporating spatial constraints or feature similarities in the future.

V. CONCLUSION

In this paper, we propose a novel segmentation-based method for online and offline HCTR. In contrast to previous oversegmentation-based approaches, we formulate a brand-new segmentation-based text recognition framework that end-to-end segments and recognizes characters through fully convolutional networks with high efficiency and accuracy. To address the high cost of character segmentation annotations, a new weakly supervised learning method is proposed to enable the network to be trained using only transcript annotations. Owing to the absence of context modeling in the fully convolutional architecture, we design a contextual regularization method to integrate contextual information into extracted features without affecting the inference speed. Extensive experiments on CASIA-HWDB, CASIA-OLHWDB, ICDAR2013, and SCUT-HCCDoc, demonstrate the superiority of our method over existing approaches. To the best of our knowledge, our method may be the first to achieve state-of-the-art performance on both online and offline HCTR. An additional trial on ReCTS-25k demonstrates the potential of our method out of the field of HCTR. We hope this work will spark further research beyond the realms of prevalent CTC/attention-based methods.

REFERENCES

[1] Q. Wang, F. Yin, and C. Liu, “Handwritten Chinese text recognition by integrating multiple contexts,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no. 8, pp. 1469–1481, Aug. 2012.
[2] X. Zhou, D. Wang, F. Tian, C. Liu, and M. Nakagawa, “Handwritten Chinese/Japanese text recognition using semi-Markov conditional random fields,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 35, no. 10, pp. 2413–2426, Oct. 2013.
[3] D.-H. Wang, C.-L. Liu, and X.-D. Zhou, “An approach for real-time recognition of online Chinese handwritten sentences,” Pattern Recognit., vol. 45, no. 10, pp. 3661–3675, 2012.
[4] F. Yin, Q. Wang, X. Zhang, and C. Liu, “ICDAR 2013 Chinese handwriting recognition competition,” in Proc. 12th Int. Conf. Document Anal. Recognit., 2013, pp. 1464–1470.
[5] X.-D. Zhou, Y.-M. Zhang, F. Tian, H.-A. Wang, and C.-L. Liu, “Minimum-risk training for semi-Markov conditional random fields with application to handwritten Chinese/Japanese text recognition,” Pattern Recognit., vol. 47, no. 5, pp. 1904–1916, 2014.
A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks,” in Proc. IAPR Int. Workshop Document Anal. Synt., 2020, pp. 45–59.

V. J. Zhang, F. Yin, Y. Zhang, C. 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.

K. Ke, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 770–778.

A. Neubeck and L. Van Gool, “Efficient non-maximum suppression,” in Proc. Int. Conf. Pattern Recognit., 2006, pp. 850–855.

M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, “Synthetic data and artificial neural networks for natural scene text recognition,” in Proc. Adv. Neural Inform. Process. Syst. Deep Learn. Workshop, 2014.

A. Gupta, A. Vedaldi, and A. Zisserman, “Synthetic data for text localisation in natural images,” in Proc. IEEE Comput. Vis. Pattern Recognit., 2016, pp. 2315–2324.

F. Zhan, S. Lu, and C. Xue, “Verisimilar image synthesis for accurate detection and recognition of texts in scenes,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 249–266.

S. Fogel, H. Averbach-Elor, S. Cohen, S. Mazor, and R. Litman, “ScrabbleGAN: Semi-supervised varying length handwritten text generation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2020, pp. 4324–4333.

The People’s daily corpus,” The People’s daily news and information center, The Peking University Institute of Computational Linguistics and Fujitsu Research and Development Center Limited, Accessed on: Mar. 25, 2016. [Online]. Available: http://icl.pku.edu.cn/icl_groups/corpus/wwd/wwd/00001l100014115957515961596045434356535349494953495649560495553534343594956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604955534956495604
[54] T. Lyons, “Rough paths, signatures and the modelling of functions on streams,” in Proc. Int. Congress Math., 2014, vol. IV, pp. 163–184.
[55] B. Hambly and T. Lyons, “Uniqueness for the signature of a path of bounded variation and the reduced path group,” Ann. Math., vol. 171, pp. 109–167, 2010.
[56] B. Graham, “Sparse arrays of signatures for online character recognition,” 2013, arXiv:1308.0371.
[57] 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. Int. Conf. Doc. Anal. Recognit., 2015, pp. 551–555.
[58] 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, 2016.

[59] L. Sun, T. Su, C. Liu, and R. Wang, “Deep LSTM networks for online Chinese handwriting recognition,” in Proc. Int. Conf. Front. Handwriting Recognit., 2016, pp. 271–276.
[60] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, “Attention-based models for speech recognition,” in Proc. Adv. Neural Inform. Process. Syst., 2015, pp. 577–585.
[61] L. Xing, Z. Tian, W. Huang, and M. R. Scott, “Convolutional character networks,” in Proc. Int. Conf. Comput. Vis., 2019, pp. 9126–9136.
[62] Y. Baek, B. Lee, D. Han, S. Yan, and H. Lee, “Character region awareness for text detection,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2019, pp. 9365–9374.
[63] A. Vaswani et al., “Attention is all you need,” in Proc. Adv. Neural Inform. Process. Syst., 2017, pp. 5998–6008.
[64] D. Peng et al., “Towards fast, accurate and compact online handwritten Chinese text recognition,” in Proc. Int. Conf. Front. Handwriting Recognit., 2021, pp. 157–171.
[65] B. Liu, W. Sun, W. Kang, and X. Xu, “Searching from the prediction of visual and language model for handwritten Chinese text recognition,” in Proc. Int. Conf. Doc. Anal. Recognit., 2021, pp. 274–288.
[66] M. Ott et al., “Fairspeech: A fast, extensible toolkit for sequence modeling,” in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol., 2019, pp. 48–53.
[67] C. Luo, L. Jin, and Z. Sun, “MORAN: A multi-object rectified attention network for scene text recognition,” Pattern Recognit., vol. 90, pp. 109–118, 2019.

Dezhi Peng received the B.S. degree in information engineering in 2019 from the South China University of Technology, Guangzhou, China, where he is currently working toward the Ph.D. degree in information and communication engineering. His research interests include optical character recognition, document analysis and recognition, and handwriting text recognition.

Lianwen Jin (Member, IEEE) received the B.S. degree from the University of Science and Technology of China, Hefei, China, in 1991, and the Ph.D. degree from the South China University of Technology, Guangzhou, China, in 1996. He is currently a Professor with the School of Electronic and Information Engineering, South China University of Technology. He is the author of more than 100 scientific papers. His research interests include computer vision, optical character recognition, handwriting analysis and recognition, machine learning, deep learning, and intelligent systems. Dr. Jin was the recipient of the Award of New Century Excellent Talent Program of MOE in 2006 and Guangdong Pearl River Distinguished Professor Award in 2011.

Weihong Ma received the B.S. degree from the School of Electronic and Information Engineering, South China University of Technology, Guangzhou, China, in 2019. He is currently working toward the master’s degree in information and communication engineering with the South China University of Technology. His current research interests include deep learning, scene text detection, and document analysis.

Canyu Xie received the B.S. degree in electronic science and technology in 2019 from the South China University of Technology, Guangzhou, China, where he is currently working toward the master’s degree in electronic and communication engineering. His current research interests include computer vision, model compression, and acceleration.

Hesuo Zhang received the B.S. degree from the School of Electronic and Information Engineering, South China University of Technology, Guangzhou, China, in 2019. He is currently working toward the master’s degree in information and communication engineering with the South China University of Technology. His current research interests include machine learning, deep learning, handwritten text segmentation, and recognition.

Shenggao Zhu received the bachelor’s degree in electronic engineering and information science from the University of Science and Technology of China, Hefei, China, in 2011, and the Ph.D. degree in computer science from the National University of Singapore, Singapore, in 2017. In 2017, he joined Huawei Cloud, where he is currently a Technical Expert. His research interests include computer vision and AI applications.

Jing Li received the B.S. degree from the University of Science and Technology of China, Hefei, China, in 2013, and the Ph.D. degree from the National University of Singapore, Singapore, in 2019. She is currently an AI engineer with Huawei Cloud Computing Technologies. Her research interests include face recognition, optical character recognition, and image generation.