IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 17, 2018, accepted May 14, 2018, date of publication May 24, 2018, date of current version June 26, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2840218

Dense and Tight Detection of Chinese Characters in Historical Documents: Datasets and a Recognition Guided Detector

HAILIN YANG, LIANWEN JIN, (Member, IEEE), WEIGUO HUANG, ZHAOYANG YANG, SONGXUAN LAI, AND JIFENG SUN

School of Electronic and Information Engineering, South China University of Technology, Guangzhou 510640, China

Corresponding author: Lianwen Jin (eelwjin@scut.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 0 2016YFB1001405, in part by NSFC under Grant 61472144, Grant 61673182, and Grant 61771199, in part by GD-NSF under Grant 2017A030312006, in part by GDSTP under Grant 2015B010101004 and Grant 2017A030312006, and in part by GZSTP under Grant 201607010227.

ABSTRACT Characters in historical documents are typically densely distributed and are difficult to localize and segment by directly applying classic proposal and regression based methods. In this paper, we propose a novel method called recognition guided detector (RGD) that achieves tight Chinese character detection in historical documents. The proposed RGD consists of two simultaneously trained convolutional neural networks: a recognition guided proposal network that provides context information of the text and a detection network that applies this information to localize each of the characters accurately. To train and test the proposed method, we established two new datasets with character-level annotations, comprising ground truth character bounding boxes and ground truth characters in each of the boxes. The data in our datasets are scanned images collected from nine different versions of Tripitaka in Han. Experimental results show that, guided by a text recognition network with a test accuracy of 97.25%, the detection network in our proposed method achieves a much higher F-score with fewer parameters under a highly constrained evaluation criterion of intersection of union (IoU) $\geq$ 0.7, when comparing to several state-of-the-art object detection and text detection methods. The datasets are publicly available at https://github.com/HCIILAB/TKH_MTH_Datasets_Release for non-commercial use.

INDEX TERMS Historical documents, character detection, recognition guided detector, data sets.

I. INTRODUCTION

Historical documents are invaluable treasures that human ancestors have created in past centuries. An important and efficient way to understand and protect these documents is through digitization, in which the texts and graphic symbols in real documents are systematically transformed into digital records. And the quality of this process strongly relies on the accuracy of character detection and recognition in the document.

Unlike historical character recognition which has been extensively discussed in the literature as a branch of Optical Character Recognition (OCR) research, only a few studies have been conducted on historical character detection. However, it is believed that detection can significantly support research on historical documents in several different aspects. First, precise character detection can support research on understanding the evolution of ancient Chinese characters. Second, accurately spotting the position of the character can help in repairing the document in cases where it is damaged, blurred, or eroded. Third, once all characters in the documents have been accurately spotted and recognized, we can then easily retrieve each of the characters from even a raw image.

The methods widely used for character detection in historical documents are, in general, based on a combination of several independent components [20], [21]. These methods rely highly on domain-specific knowledge and careful tuning of hyper-parameters, which makes their application difficult to adapt on new datasets.

In recent years, deep learning methods have achieved substantial success in many applications. Tasks that focus on recognizing characters and texts in historical documents can also achieve better performance [3], [12], [28], [31] by combining

OCR techniques with different deep learning architectures such as Convolutional Neural Networks (CNNs) [11], Long Short-Term Memory (LSTM) [6] and Connectionist Temporal Classification (CTC) [7].

Several region-based methods [1], [16] and region-free methods [13], [15] exist for general detection tasks. These methods achieve state-of-the-art performance by incorporating deep learning models into their algorithms. However, even though these algorithms perform highly in general object detection and scene text detection tasks, for the following reasons, applying them to the detection of characters in historical documents is not a straightforward matter. Firstly, the amount of available well-labeled data associated with historical documents is insufficient. Secondly, characters are generally densely distributed in historical documents, which is different from the case in other detection tasks. Thirdly, characters need to be tightly bounded to allow further analysis, especially for Chinese characters. However, to date, only a few studies have attempted to fully address these challenges.

This paper aims to address the aforementioned three challenges. First, we propose two new datasets that contain scanned Tripitaka images with ground truth labels, including ground truth character bounding boxes and ground truth characters in each of the boxes. Secondly, inspired by the human cognition mechanism, we present a Recognition Guided Detector (RGD) that achieves dense and tight character-level detection in historical documents under the guidance of a text recognition network. RGD uses the pre-estimated position provided by a recognition network, which we call the Recognition Guided Proposal Network (RGPN), to find the precise position of the characters with another CNN-based regression network. The results of several experiments conducted to test the performance of the proposed detector on our datasets show that the detection network achieves better performance than most state-of-the-art object detection and text detection methods, while requiring fewer parameters.

The remainder of this paper is organized as follows. In Section II, we introduce related work on object detection, including text detection. Our proposed datasets are introduced in Section III, which is followed by details of our proposed detector in Section IV. Section V presents experimental results and discussions. Finally, Section VI presents our conclusion.

II. RELATED WORK

The methods widely used for character-level detection in historical documents are predominantly based on a combination of independent components. For example, Phan *et al.* [21] proposed an efficient method based on connected component analysis to extract characters from images, and Panichkriangkrai *et al.* [20] proposed a system that segments characters in Japanese historical books by applying region labeling of connected components followed by rule-based integration. Tseng and Chen [29] utilized stroke bounding boxes followed by a knowledge-based merging operation to segment characters. However, adapting these methods to new datasets is difficult because they rely much on hyper-parameters that may have different optimal values for different datasets.

Nowadays, object detection is an active research area. Its aim is to localize objects with bounding boxes and recognize them [2], [14], [18]. By using deep learning methods, the performance of algorithms in this area has significantly improved over other non-deep learning based methods. For example, Girshick proposed a two-stage approach called R-CNN [5], a combination of the Convolutional Networks (Conv-Nets) and region proposals generated by Selective Search (SS) [30] or Edgeboxes [36] methods. Subsequently, to speed up R-CNN, He *et al.* proposed the Spatial Pyramid Pooling Network (SPP-Net) [8] and Girshick proposed the Fast R-CNN algorithm with RoI-Pooling [4]. Both of these methods allow part of the network to share features computed by convolutional layers. Improvements have also been made to Fast R-CNN, resulting in Faster R-CNN [23], which has a more effective region proposal method, and R-FCN [1], which incorporates a technique known as position-sensitive score maps.

Unlike the above region-based methods that need to pre-generate region proposals (or candidate boxes) before refining them with a region-wise sub-network, region-free methods can skip the region proposal generation process and detect the objects directly. Single-Shot Detector (SSD) [15] and YOLO [22] are two representatives. They are both based on FCNs [17], in which the position information is preserved during convolutional calculation. These region-free methods can achieve performance comparable to that of region-based methods with a much faster detection speed.

However, although many deep learning methods have achieved substantial success in scene text detection tasks [13], [24], [34], [35], applying them to historical character detection tasks is difficult due to the lack of labeled data. At present, the most frequently used text detectors are enhanced versions of object detectors that have been applied on scene tasks. For example, Liao et al. proposed TextBoxes [13], which overcame the limitations of SSD in detecting words that have extreme aspect ratios, and has achieved promising text detection performance. Further, the FEN [33], proposed by Zhang *et al.*, extracts multi-scale features and then detects texts based on the combination of these features, which addresses the problem of detecting texts of different scale. To detect oriented scene text, Shi *et al.* proposed Segment Linking [26], which decomposes text into two locally detectable elements. Instead of detecting scene text with rectangular bounding boxes, Liu and Jin proposed a method that detects oriented text with tighter quadrangle boxes [16] and have achieved even better performance.

III. DATASETS

To facilitate the research of Chinese character detection in historical documents, we propose the Tripitaka Koreana in Han (TKH) Dataset and the Multiple Tripitaka in Han (MTH) Dataset.

TABLE 1. Details of the TKH and MTH datasets.

	TKH	MTH
Pages	1,000	500
Lines	23,471	17,178
Character instances	323,491	197,886
Character categories	1,471	3,664

Tripitaka Koreana in the Han language has been playing an important role in connecting Buddhism to the public since its first release in AD 11 [19]. We downloaded the TKH images, which were originally released by the research institute of Tripitaka Koreana [19], from the internet. The first row of Fig.1 shows typical sample pages of Tripitaka Koreana images. Although these scanned images have noise such as spots, tears, ink fading, and transparent backside, they can be easily read without any preprocessing.

However, labeling the image from scratch without any preprocessing is very time-consuming and laborious because there are too many (usually more than 300) characters per image. Therefore, we used a vertical projection method to separate the images into text lines, which are then further over-segmented by a beam-search algorithm to obtain the initial rectangular bounding boxes of characters. Then, we adjusted these bounding boxes to ensure they bound the characters accurately. During the adjustment, we also annotated the ground truth character in each of the boxes. Examples of the ground truth bounding boxes are shown in Fig.3.

Our TKH Dataset contains 1,000 images, comprising approximately 320,000 character instances and 23,000 lines. The details of the dataset are given in Table 1. As the layout of the images in the TKH Dataset are relatively regular and the characters in them are mostly uniform, it can be a suitable baseline dataset for research on character detection and recognition in historical documents.

For more complicated situations, we propose the MTH Dataset, which contains scanned images from eight different Tripitaka versions in China. The data in this dataset induces more complicated situations, which makes the MTH Dataset more challenging than the TKH Dataset. For example, some images contain drawings, and some have more than one text area. The most challenging case is where characters in the same line appear in different sizes. In this case, smaller characters are very blurred and close to each other. Some examples are shown in Fig 1. These challenges also make it infeasible to generate initial bounding boxes in the same manner as that used on the TKH Dataset. Instead, we used a detector trained with the TKH Dataset to generate these bounding boxes. Examples of ground truth bounding boxes of the MTH Dataset are shown in Fig. 3.

Our MTH Dataset contains approximately 500 images that are more complicated and more representative than those in the TKH Dataset. Thus we believe that the MTH Dataset has a unique value and can further support research on historical

FIGURE 1. Sample images from the TKH Dataset (first row) and the MTH Dataset (second and third rows).

document images. The complete TKH and MTH datasets are publicly available for non-commercial use.

IV. APPROACH

Humans can accurately tell the positions of characters in dense documents, even when some of them are confusing. This is in large part because of our ability to recognize the characters. Inspired by this, we propose a recognition guided character detection method called Recognition Guided Detector (RGD) that makes use of the context information to help detect characters in historical document images. As shown in Fig. 2, our proposed method consists of three main parts: a text line segmentation process, a proposal generation network called Recognition Guided Proposal Network (RGPN), and a character-level detector. The text line segmentation process first segments the input image into text lines. Then, a RGPN is used to generate the context information, which the character-level detector then uses to infer the positions of characters in the text lines.

A. TEXT LINE SEGMENTATION

In most documents in the TKH Dataset, characters are clearly written and neatly arranged to facilitate reading and spreading. Therefore, they generally have clear text line patterns. We exploit this pattern to segment the whole image into text lines using some classical approaches.

In this paper, we use vertical projection [25] of the grey scale image to segment text lines by finding the dividing points on the image. Let TW denote the average text width

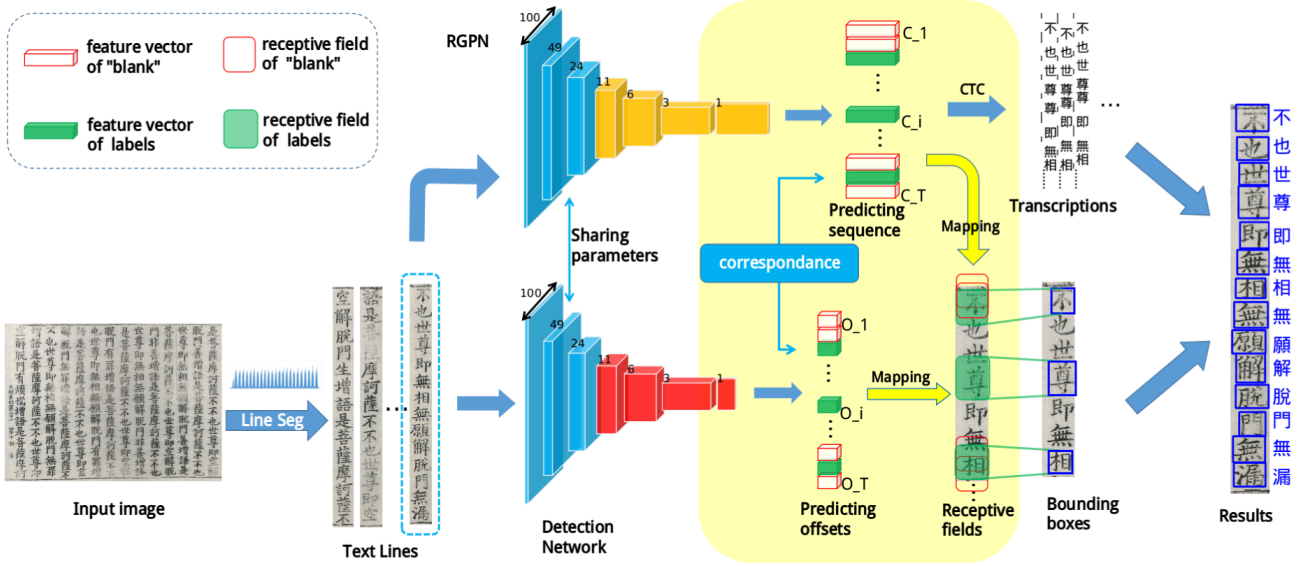

FIGURE 2. The architecture of our RGD. It comprises of three main parts: text line segmentation, proposal generation, and character-level detection. The system first segments the text lines from the input image by vertical projection. Then, it generates proposals of characters in text lines via a recognition guided proposal network (RGPN). Finally, precise bounding boxes of characters are obtained via our new method called Recognition Guided Detector(RGD). The yellow part of the image illustrates the connection between RGPN and the detection network, through which the detection network can be promoted by the context information from RGPN.

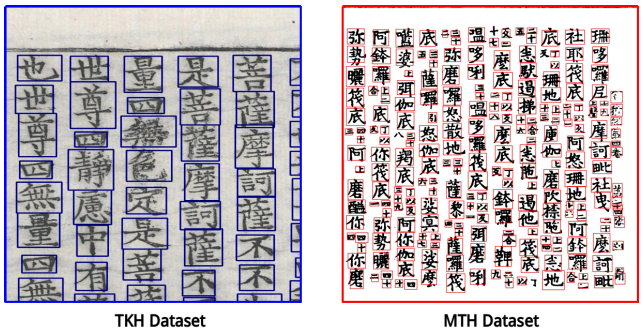

FIGURE 3. Examples of ground truth bounding boxes in the TKH and MTH Datasets.

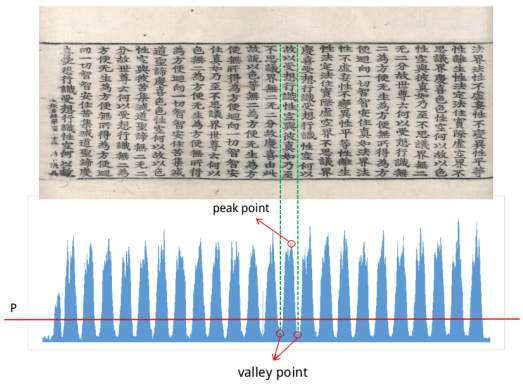

FIGURE 4. Text line segmentation by vertical projection. We consider a line to be valid only when its peak is higher than the threshold P.

and R denote the text lines search region. Then, each text line area can be determined by finding two adjacent valley points with width, greater than a threshold W and the peak projection data greater than P. The details are shown in Fig.4. In this paper, we set R at 0.8 W at 0.5, and P at 2.

B. RECOGNITION GUIDED CHARACTER DETECTOR

We conducted character-level detection on the segmented text line. In classical object (scene text) detection frameworks, classification and detection of objects (or texts) are independently carried out based on proposals, which are either predefined (region-free methods) or generated by region proposal networks (region-based methods). The only connection between these two sub-tasks is through parameter sharing of some layers. However, in addition to parameter sharing, we believe that the context information around a character can also improve the detection performance, especially for dense and tight Chinese character detection in historical document. Therefore, we propose a Recognition Guided Proposal Network (RGPN), and use it beforehand to encode the context information into proposals. Then, in the remaining detection process, we only need to consider proposals that contain the most context information.

1) RECOGNITION GUIDED PROPOSAL NETWORK

Yin *et al.* [32] proposed a CNN-based scene text recognition method that first recognizes the text based on the sliding window method, and then decodes the sequence using CTC [7] to remove redundant predictions. Inspired by their work, we designed RGPN with an architecture that contains convolutional layers followed by multiple fully connected layers and a CTC layer. The merits of RGPN are manifold. Firstly, it can recognize the input text lines through pre-training in a manner similar to that in [32]; therefore, it can provide

context information. Secondly, because the movement of the receptive fields of CNNs can be regarded as an implicit sliding window, RGPN can also provide context information at each receptive field in the input image. Thirdly, we can also share parameterized layers between RGPN and the detection network, such that the two networks can cooperate with each other to achieve better performance.

CTC [7] applies a softmax function to the input sequence for every time step, which provides a probability at that time step for emitting each label from an extended label set including all the character labels, plus an extra "*blank*" symbol that represents null emission. We designed RGPN with the input size of its CTC layer as $K \times 1$, in which K denotes the number of labels in the extended label set and T is the length of the predicting sequence. We assume that each unit on the T axis is a "*time step*" and the feature vector at a "*time step*" represents the context information in its corresponding receptive field. The output of RBPN with a size of $K \times 1$ indicates the possibilities of characters over K classes at each "*time step*". A "*time step*" with a non-blank prediction result indicates that its corresponding receptive field contains a character, which can be regarded as a positive proposal for the detection network. Otherwise, the detection network will simply skip that "*time step*".

The receptive field corresponding to each "*time step*" is a rectangular local region in the input image that can be properly represented by the feature vector at that "*time step*". Let r_i denote the local region size (width/height) of the i-th layer, and let (x_i, y_i) denote the center position of this local region. Then, the relationship of r_i and (x_i, y_i) between adjacent layers can be formulated as follows:

$$\begin{aligned} r_i &= (r_{i+1} - 1) \times S_i + K_i \\ x_i &= S_i \times x_{i+1} + (\frac{K_i - 1}{2} - P_i) \\ y_i &= S_i \times y_{i+1} + (\frac{K_i - 1}{2} - P_i) \end{aligned} \tag{1}$$

where K is the kernel size, S is the stride size, and P is the padding size of a particular layer. The areas corresponding to the receptive fields can be obtained by applying (1) recursively to adjacent layers in the convolutional network from the output down to the input images.

In summary, on one hand, our RGPN can recognize line texts in a sliding window mechanism along the line of text. A well-performing text recognizer can capture all characters in a text line; thus, it can find the rough positions of all characters. On the other hand, unlike the traditional region proposal network (RPN) used in previous region-based object detectors such as Faster R-CNN [23] and R-FCN [1], RGPN utilizes the context information to tell exactly whether a proposal is positive (from where it can recognize a character) or not.

2) CHARACTER-LEVEL DETECTION

The precise location of each character can be obtained by regressing the proposals provided by RGPN. For this task, we propose a CNN-based detection network. Each of the outputs from this detection network is coupled with a corresponding output in RGPN. We call our detection method Recognition Guided Detector (RGD), which achieves character-level detection with fewer parameters.

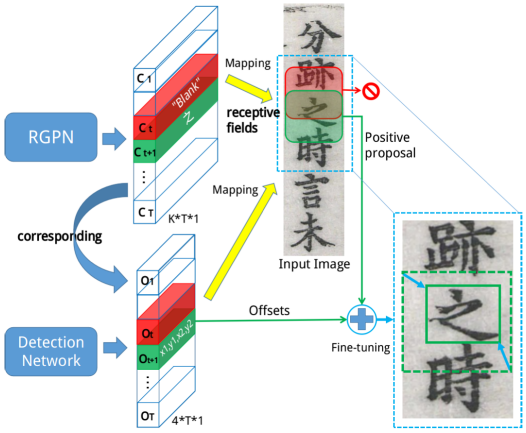

FIGURE 5. Detection with the proposals from RGPN. Feature vectors of the same position in the output of two networks are computed from the same receptive field. The receptive field with a non-blank prediction result (the green one) is regarded as a positive proposal, whereas the red one with "*blank*" prediction result is discarded. The prediction bounding boxes are obtained by fine-tuning the corresponding positive proposals using the offsets provided by the detection network.

Fig.5 illustrates the process of character detection with proposals from RGPN. The output of RGPN consists of probabilities over the extended label set at every "*time step*", which indicates whether the corresponding receptive field is a positive proposal or not. To maintain consistency with the RGPN's output shape of $K \times 1$, the output shape of our detection network is set to 4×1, where 4 signifies four coordinate offsets to the corresponding receptive fields. To avoid losing position information in the output of RGPN, we continuously match the shape of the feature maps of both RGPN and the detection network, such that feature vectors at the same position in the output of the two networks are extracted from the same receptive field. For example, in Fig.5, the red vectors of the two different feature maps (C_t and O_t) are extracted from the same red receptive field in the input image. In our detection pipeline, the number of positive proposals is approximately equal to the number of characters recognized by RGPN from the text line image, which is much smaller than the case in other detection methods.

When calculating the loss of the detection network during training, only the loss at the valid position decided by RGPN, which are receptive fields with a non-blank prediction, is counted. Note that the detection network does not score or decide a valid position itself. For these valid positions, we chose the smooth_L1 loss [23] to improve the performance in the detection task. Therefore, the loss function is formulated as (2),(3), in which P_i and G_i denote the predictions of the detection network and the ground truth at the i_{th} position, respectively, and $\mathbb{I}(\cdot)$ denotes the guidance

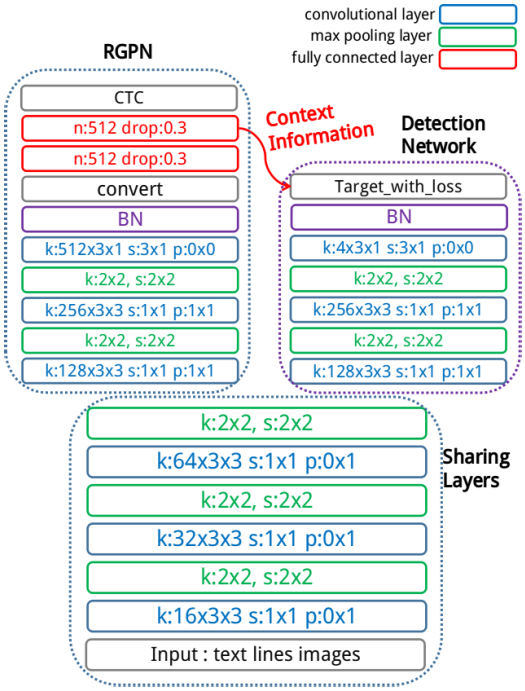

FIGURE 6. Architecture of the proposed Recognition Guided Detector (RGD).

from RGPN:

$$\$_g = \sum_{i=0}^{T} smooth\_L1(P_i, G_i) \times \mathbb{I}(P_i) \tag{2}$$

where

$$\mathbb{I}(P_i) = \begin{cases} 0, & i_{th}\ position\ is\ invalid \\ 1, & i_{th}\ position\ is\ valid \end{cases} \tag{3}$$

In general, the number of output bounding boxes of most other object detectors, regardless of whether they are region-based or region-free, is much greater than the number of characters in the image. The common solution is to process and select them with a non-maximum suppression (NMS) process that requires predicted scores for every detected bounding box to indicate their quality. As proposals from RGPN are more precise, the NMS process can be replaced by a simpler suppression process: if the overlap of two final bounding boxes with the same recognition result is higher than a threshold t, then the bounding box with the lower recognition confidence is discarded.

V. EXPERIMENTS

A. EXPERIMENTAL SETUP

The detailed architecture of our framework is shown in Fig.6. The inputs to the network are binary images that have been resized to a fixed shape of 1000 × 100. The output size is $N \times 1472 \times 32$ in the RBPN and $N \times 4 \times 32$ in the detection network (N denotes the batch-size). We applied a batch normalization [10] operation after the last convolutional layer to accelerate training and avoid over-fitting.

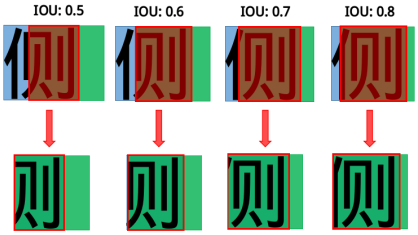

FIGURE 7. Examples of different IoUs. The Blue boxes denote ground truth, green boxes denote the detection results.

The conversion operation in RGPN after the batch normalization layer is applied to modify the output shape of ConvNets to suit the CTC layer. Finally, we designed the final layer of the detection network (Target_with_loss layer) to select valid positions according to the proposals and calculate the Smooth-L1 loss at these positions.

To evaluate the performance, we calculated the correct rate (CR) and accuracy rate (AR) [32] of the recognition network, and the precision (P), recall (R), and F-score (F) for the detection network. In most detection tasks, it is widely accepted that prediction bounding boxes are considered positive when their Intersection-over-Union (IoU) with ground truth boxes are greater than 0.5. However, as shown in Fig.7, for Chinese characters, $IoU \geq 0.5$ is not sufficient for recognizing the characters because it may result in strokes being missed. Therefore, we evaluated the performance of different methods with criteria $IoU \geq 0.5, 0.6, 0.7, and 0.8$, respectively.

We randomly selected 300 images from the TKH Dataset as a testing set, and used the remaining 700 images as a training set. Because the recognition network needs to provide context information to guide the detection, it should be pre-trained before training the detection network. Our RGPN was pre-trained for 20 epochs with text lines obtained from the TKH Dataset by using the segmentation method in Section IV-A. The final AR and CR of our RGPN were 95.44% and 97.34%, respectively.

B. RESULTS AND ANALYSIS

1) EFFECTIVENESS OF USING CONTEXT INFORMATION

The context information provided by RGPN is the probabilities over the extended label set at each position. Such information can be used during the training phase of the detection network for calculating gradients and during the testing phase to help the detection network to find the final bounding boxes. To evaluate the effectiveness of context information, we conducted a set of experiments in which the detection network was trained and tested with and without using any information from RGPN. The results are shown in Table 2. Note that, when detecting without the recognition network, the detection network produces an extra output, which is the score of the bounding box.

It is clear that incorporating the context information from RGPN into the detection network during training leads to much better performance compared to the other two imple-

TABLE 2. Results of our proposed method with different context information usage based on precision(P), recall(R) and F-score(F) in different level of IoU levels. All the experiments were performed on the TKH Dataset.

		Training with Context Info	Post-Processing with Context Info	Without Context Info
IoU0.5	P	98.58	96.69	97.73
	R	96.96	95.48	93.21
	F	97.76	96.08	95.42
IoU0.6	P	97.23	93.83	93.72
	R	96.50	93.16	90.22
	F	96.86	93.49	91.94
IoU0.7	P	94.93	67.92	74.75
	R	94.27	67.57	72.44
	F	94.60	67.74	73.58
IoU0.8	P	84.08	35.24	40.17
	R	83.56	35.97	39.44
	F	83.82	35.60	39.80

TABLE 3. Comparison of results obtained by sharing different layer parameters between RBPN and the detection network. When no layer was shared, we trained them independently. All the detection results were obtained under the criterion of *IoU* ≥ 0.7.

Shared Layers	AR	CR	P	R	F
Not sharing	96.83	97.90	94.40	93.55	93.97
First 2 layers	96.80	97.78	94.89	93.22	94.05
First 3 layers	**97.25**	98.61	**94.93**	**94.27**	**94.60**
First 4 layers	97.24	**98.63**	94.09	93.50	93.79
First 5 layers	97.12	98.55	92.96	92.39	92.67

mentations. It is also clear that the results in the second column deteriorate when larger IoUs are required. This may be because the recognition network is only used in post-processing and no ground truth about the content in the box is given during training.

2) EFFECTIVENESS OF LAYER PARAMETER SHARING

As the feature extraction layers of RGPN and the character-level detection network are convolutional layers, it is natural to share some of the parameters between them to enhance the connection between the two networks as well as to save parameter storage. We conducted experiments in which we compared the performance of the networks when sharing different numbers of layers in the first five layers. For example, the network in Fig.6 has shared parameters for the first three convolutional layers. Note that, on sharing a specific layer, that layer was trained jointly by both networks; unshared layers were trained independently. Table 3 shows the results of these experiments.

It can be seen from Table 3 that sharing parameters generally improves the performance of both RGPN and the detection network. However, when the number of shared layers increases, the level of improvements decreases. As indicated by Table 3, our method achieves the best performance when the first three convolutional layers are shared.

3) COMPARISON WITH STATE-OF-THE-ART METHODS

We compared our proposed method with several state-of-the-art general object detection and scene text detection methods. The general object detection methods were two typical region-based methods, specifically, R-FCN [1] and Faster R-CNN [23], and two typical region-free ones, specifically, SSD [15] and YOLO [22]. The compared scene text detection methods were TextBoxes [13], DMP-Nets [16] and FEN [33]. FEN and DMP-Nets use the Resnet-101 network [9] for feature extraction, whereas all the other methods use the VGG-16 network [27]. We tested these methods on our TKH Dataset using the entire image as input. The results are shown in the top half of Table 4. The results for traditional methods are not shown because they are less competitive than deep learning-based methods (for example, histogram projection has only an F-score of 33% when $IoU \geq 0.7$).

It can be seen that, with at least seven times fewer parameters, our Recognition Guided Detector (RGD) method outperforms other methods on F-score under the evaluated criteria of $IoU \geq 0.7$ and $IoU \geq 0.8$. When the criterion becomes less constrained (e.g. $IoU \geq 0.6$ or $IoU \geq 0.5$), our method still achieves performance comparable to that of the state-of-the-art methods. Note that our proposed RGD method predicts a bounding box only when RGPN can recognize a character from the specific area. Therefore, the performance, especially the recall of our detection network, is limited. This becomes more obvious as the evaluation criterion becomes less constrained. Thus, our proposed method is more suitable for tight detection scenarios, such as this tight character detection application.

As the detection network in our proposed method uses text lines as input, for fair comparison, we also tested the performance of other detection methods using text lines as input. The results are given in the bottom half of Table 4, in which previous methods with the preprocessing we introduced in Section IV-A are denoted by *method*-Line. The experimental results show that previous methods achieved better performance when taking text lines as input compared to their performance when taking the entire image as input. This may be because the process of text line segmentation reduced the detection difficulty by splitting a whole dense character area into more sparse separated lines.

4) TESTING ON THE MTH DATASET

As stated in Section III, data in the MTH Dataset comprise more diverse and complex situations than the TKH Dataset. This makes the MTH Dataset very suitable for testing the robustness and generalization of the models trained with each of the methods. Therefore, we conducted a set of experiments to compare the testing performance of the well-trained models in Section V-B3 on the MTH Dataset. Table 5 shows the results of the experiments.

Note that as the proposed RGD method uses text lines as input, for comparison purpose, only the results for experiments that used text lines as input are shown. In addition, to simplify the implementation, when segmenting the lines, we guaranteed that each segmented image contained only one line of characters. Nonetheless, it is worth mentioning that when the entire image was used as input, the performance of

TABLE 4. Comparison between RGD and other detection methods on the TKH dataset.

	Param	IoU:0.5			IoU:0.6			IoU:0.7			IoU:0.8		
		P	R	F	P	R	F	P	R	F	P	R	F
RFCN [1]	70.52M	99.69	90.96	94.98	99.34	90.37	94.64	97.32	88.54	92.72	78.22	71.15	74.52
Faster-RCNN [23]	130.07M	90.70	99.79	95.03	99.53	90.46	94.78	97.89	88.97	93.22	82.16	74.67	78.24
YOLO [22]	232.19M	-	-	-	-	-	-	-	-	-	-	-	-
SSD [15]	87.30M	99.92	66.03	75.23	99.78	60.22	75.10	98.54	59.56	74.24	86.60	52.31	65.23
TextBoxes [13]	90.64M	99.92	57.27	72.81	99.77	57.18	72.70	98.51	56.46	71.78	84.77	48.58	61.77
DMP-Nets [16]	178.56M	99.43	89.54	94.23	98.63	88.82	93.47	95.29	85.82	90.31	71.59	64.48	67.85
FEN [33]	176.85M	99.34	97.57	**98.44**	98.18	95.86	97.00	89.66	86.74	88.18	62.28	59.41	60.81
RGD[ours]	**9.29M**	98.32	97.52	97.92	97.23	96.50	96.86	94.93	94.27	94.60	84.08	83.56	83.82
RGD-VGG16[ours]	64.02M	98.58	96.96	97.76	97.64	96.39	**97.01**	95.40	94.56	**94.98**	85.97	85.72	**85.85**
RFCN-Line	70.52M	99.54	99.58	**99.56**	98.88	98.92	**98.90**	95.92	95.49	95.70	83.22	83.19	83.21
Faster-RCNN-Line	130.07M	99.44	99.33	99.38	98.35	98.46	98.40	94.11	94.32	94.22	79.73	79.74	79.74
YOLO-Line	232.19M	92.28	96.09	94.15	91.19	95.22	93.16	83.26	86.94	85.06	56.73	59.24	57.95
SSD-Line	87.30M	99.56	96.44	97.98	98.54	95.81	97.16	94.65	92.64	93.63	79.07	78.00	78.53
TextBoxes-Line	90.64M	98.49	98.49	98.66	97.89	98.23	98.06	95.75	96.08	95.91	86.82	87.12	**86.97**
DMP-Nets-Line	178.56M	99.56	99.46	99.51	98.98	98.67	98.82	96.37	96.06	**96.22**	81.19	80.93	81.06
FEN-Line	176.85M	99.62	99.46	99.54	99.01	98.65	98.83	96.52	94.91	95.71	77.15	74.48	75.79

The bottom half are the results of other methods when text lines are used as input. RGD-VGG16 is our RGD method using the VGG-16 network as the backbone detection network. YOLO cannot be trained when using the entire image as input because it is not suitable for detection of densely distributed objects, as also mentioned by the authors [22].

TABLE 5. Comparison of RGD with other detection methods using text lines from the MTH Dataset as input.

	IoU:0.5			IoU:0.6			IoU:0.7			IoU:0.8		
	P	R	F	P	R	F	P	R	F	P	R	F
RFCN-Line	96.30	97.68	**96.98**	94.36	95.72	95.04	84.50	85.72	85.11	46.94	47.61	47.27
Faster-RCNN-Line	96.15	97.44	96.79	93.26	94.52	93.89	81.60	82.70	82.15	43.97	44.57	44.27
YOLO-Line	92.09	88.51	90.26	88.96	85.50	87.20	77.14	74.14	75.61	42.16	40.52	41.33
SSD-Line	98.85	90.41	94.44	97.48	89.17	93.14	89.81	82.14	85.81	58.39	53.40	55.78
TextBoxes-Line	81.75	93.88	87.38	80.11	91.99	85.64	75.87	87.13	81.11	58.33	66.98	62.36
DMP-Nets-Line	96.45	95.91	96.18	94.65	94.12	94.38	85.13	84.66	84.89	46.05	45.79	45.92
FEN-Line	96.37	91.40	93.82	95.17	88.98	91.97	83.91	77.23	80.44	42.52	38.00	40.14
RGD[ours]	97.35	95.97	96.65	95.34	94.00	94.67	88.81	87.55	88.17	61.98	61.10	61.54
RGD-VGG16[ours]	97.71	95.86	96.78	96.44	94.61	**95.52**	92.17	90.42	**91.29**	73.72	72.31	**73.01**

all methods dropped dramatically. This is mainly due to the complex layout of the image pages in the MTK Dataset.

It can be clearly seen in Table 5 that our RGD method significantly outperformed other methods when the VGG-16 network was used as the backbone detection network. This is primarily as a result of the context information provided by RGPN, which guides our detector to focus more on the text area, without being confused by noise in these more complex situations.

VI. CONCLUSION

In this paper, we proposed a novel detection method (a recognition guided detector) that utilizes the context information provided by a proposed recognition network, called RGPN, to boost its detection performance. Two new datasets that can aid research into dense Chinese character detection and recognition in historical documents were also presented. The two datasets, the TKH Dataset and the MTH Dataset, comprise images taken from Chinese historical documents with character-level annotations (and are publicly available for non-commercial use).

Comparative performance evaluations were conducted with several typical text detection and object detection methods on the two proposed datasets. The results show that for the TKH Dataset, our proposed method, which involves fewer parameters, achieves comparable performance against state-of-the-art methods when using text lines as input, and outperforms them when the entire image is used as input. On the MTH Dataset, our method was much better than the other methods for IoU $\geq$ 0.6 or higher. This demonstrates that our model is more robust than those methods.

However, the MTH Dataset comprises more challenging but common situations in historical documents than the TKH Dataset. These challenges are yet to be overcome in the area of historical document analysis.

Nonetheless, in general, our proposed datasets and RGD method, as well as its promising performance, provide a good benchmark that can aid investigations of intelligent heritage in historical document images, such as Tripitaka in Han.

ACKNOWLEDGMENT

The authors sincerely thank Beijing Longquan Monastery for organizing volunteers to help building the datasets.

REFERENCES

[1] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 379–387.

[2] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes challenge: A retrospective," *Int. J. Comput. Vis.*, vol. 111, no. 1, pp. 98–136, Jan. 2014.

[3] V. Frinken, A. Fischer, and C.-D. Martínez-Hinarejos, "Handwriting recognition in historical documents using very large vocabularies," in *Proc. Int. Workshop Historical Document Imag. Process.*, 2013, pp. 67–72.
[4] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1440–1448.
[5] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. CVPR*, Jun. 2014, pp. 580–587.
[6] A. Graves, *Long Short-Term Memory*. Berlin, Germany: Springer, 2012.
[7] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 855–868, May 2009.
[8] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.
[9] 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.
[10] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. ICML*, 2015, pp. 448–456.
[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[12] B. Li, L. Peng, and J. Ji, "Historical Chinese character recognition method based on style transfer mapping," in *Proc. IAPR Int. Workshop Document Anal. Syst.*, Apr. 2013, pp. 96–100.
[13] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "Textboxes: A fast text detector with a single deep neural network," in *Proc. AAAI*, 2017, pp. 4161–4167.
[14] T. Y. Lin *et al.*, "Microsoft coco: Common objects in context," in *Proc. ECCV*, 2014, pp. 740–755.
[15] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. ECCV*, 2016, pp. 21–37.
[16] Y. Liu and L. Jin, "Deep matching prior network: Toward tighter multi-oriented text detection," in *Proc. CVPR*, Jul. 2017, pp. 3454–3461.
[17] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.
[18] Q. Lu, C. Liu, Z. Jiang, A. Men, and B. Yang, "G-CNN: Object detection via grid convolutional neural network," *IEEE Access*, vol. 5, pp. 24023–24031, 2017.
[19] *Tripitaka Koreana Knowledgebase*. Accessed: Nov. 10, 2017. [Online]. Available: http://kb.sutra.re.kr/ritk/index.do
[20] C. Panichkriangkrai, L. Li, and K. Hachimura, "Character segmentation and retrieval for learning support system of japanese historical books," in *Proc. Int. Workshop Historical Document Imag. Process.*, 2013, pp. 118–122.
[21] T. V. Phan, B. Zhu, and M. Nakagawa, "Development of Nom character segmentation for collecting patterns from historical document pages," in *Proc. Int. Workshop Historical Document Imag. Process.*, 2011, pp. 133–139.
[22] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. CVPR*, Jun. 2016, pp. 779–788.
[23] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
[24] X. Ren, Y. Zhou, Z. Huang, J. Sun, X. Yang, and K. Chen, "A novel text structure feature extractor for Chinese scene text detection and recognition," *IEEE Access*, vol. 5, pp. 3193–3204, 2017.
[25] 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. ICDAR*, Jul. 2009, pp. 651–655.
[26] B. Shi, X. Bai, and S. J. Belongie, "Detecting oriented text in natural images by linking segments," in *Proc. CVPR*, Jul. 2017, pp. 3482–3490.
[27] K. Simonyan and A. Zisserman. (2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556
[28] Y. Tang, L. Peng, Q. Xu, Y. Wang, and A. Furuhata, "Cnn based transfer learning for historical Chinese character recognition," in *Proc. Document Anal. Syst.*, Apr. 2016, pp. 25–29.
[29] L. Y. Tseng and R. C. Chen, "Segmenting handwritten Chinese characters based on heuristic merging of stroke bounding boxes and dynamic programming," *Pattern Recognit. Lett.*, vol. 19, no. 10, pp. 963–973, 1998.
[30] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *Int. J. Comput. Vis.*, vol. 104, no. 2, pp. 154–171, Apr. 2013.
[31] Y.-C. Wu, F. Yin, and C.-L. Liu, "Improving handwritten Chinese text recognition using neural network language models and convolutional neural network shape models," *Pattern Recognit.*, vol. 65, pp. 251–264, May 2017.
[32] F. Yin, Y.-C. Wu, X.-Y. Zhang, and C.-L. Liu. (2017). "Scene text recognition with sliding convolutional character models." [Online]. Available: https://arxiv.org/abs/1709.01727
[33] S. Zhang, Y. Liu, L. Jin, and C. Luo. (2017). "Feature enhancement network: A refined scene text detector." [Online]. Available: https://arxiv.org/abs/1711.04249
[34] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai, "Multi-oriented text detection with fully convolutional networks," in *Proc. CVPR*, Jun. 2016, pp. 4159–4167.
[35] Y. Zhu, C. Yao, and X. Bai, "Scene text detection and recognition: Recent advances and future trends," *Frontiers Comput. Sci.*, vol. 10, no. 1, pp. 19–36, 2016.
[36] C. L. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *Proc. ECCV*, 2014, pp. 391–405.

HAILIN YANG received the B.S. degree in engineering from the College of Electronic Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2016. He is currently pursuing the master's degree with the Deep Learning and Vision Calculations Laboratory, South China University of Technology, Guangzhou, China. His current research interests include deep learning, computer vision, and optical character recognition.

LIANWEN JIN received the B.S. degree from the University of Science and Technology of China, Anhui, 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 has authored over 100 scientific papers. His research interests include computer vision, optical character recognition, handwriting analysis and recognition, machine learning, deep learning, and intelligent systems. He was a recipient of the award of New Century Excellent Talent Program of MOE in 2006 and the Guangdong Pearl River Distinguished Professor Award in 2011.

WEIGUO HUANG received the B.S. degree in electronics and information engineering from the South China University of Technology in 2016, where he is currently pursuing the master's degree in information and communication engineering. His research interests include unsupervised learning, software development, and web design.

ZHAOYANG YANG received the bachelor's degree in engineering from the School of Electronic and Information Engineering, South China University of Technology (SCUT), Guangzhou, China, in 2015, and the M.Phil. degree from the University of New South Wales, Canberra, ACT, Australia, as a part of a double master's degree program. He is currently pursuing the master's degree with SCUT. His current research interests include deep learning, computer vision, and robotics.

SONGXUAN LAI received the B.S. degree in electronics and information engineering from the South China University of Technology in 2016, where he is currently pursuing the Ph.D. degree in information and communication engineering. His research interests include machine learning, handwriting analysis and recognition, signal processing, and computer vision.

JIFENG SUN received the B.S. degree in machine building and automation and the M.S. degree in precision instruments and testing from Tsinghua University, Beijing, China, in 1983 and 1986, respectively, and the Ph.D. degree in electrical engineering from Kyushu University, Fukuoka, Japan, in 1995. From 2005 to 2009, he was a Chief Professor with the Computer Vision and Intelligent Information Processing Laboratory. He is currently a Professor and Ph.D. Supervisor with the Department of Electronic and Information Engineering. His current research interests include computer vision, machine learning, and self-organizing communication networks.