IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 1, 2019, accepted December 16, 2019, date of publication December 24, 2019, date of current version January 2, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2961964

Text Detection and Recognition for Images of Medical Laboratory Reports With a Deep Learning Approach

WENYUAN XUE[1], QINGYONG LI[1], (Member, IEEE), AND QIYUAN XUE[2]
[1]Beijing Key Laboratory of Transportation Data Analysis and Mining, Beijing Jiaotong University, Beijing 100044, China
[2]Department of Burn and Plastic Surgery, The Fifth People's Hospital of Datong, Datong 037000, China

Corresponding authors: Wenyuan Xue (wyxue17@bjtu.edu.cn) and Qingyong Li (liqy@bjtu.edu.cn)

This work was supported by the CERNET Innovation Project under Grant NGII20170723.

ABSTRACT The adoption of electronic health records (EHRs) is an important step in the development of modern medicine. However, complete health records are not often available during treatment because of the functional problem of the EHR system or information barriers. This paper presents a deep-learning-based approach for textual information extraction from images of medical laboratory reports, which may help physicians solve the data-sharing problem. The approach consists of two modules: text detection and recognition. In text detection, a patch-based training strategy is applied, which can achieve the recall of 99.5% in the experiments. For text recognition, a concatenation structure is designed to combine the features from both shallow and deep layers in neural networks. The experimental results demonstrate that the text recognizer in our approach can improve the accuracy of multi-lingual text recognition. The approach will be beneficial for integrating historical health records and engaging patients in their own health care.

INDEX TERMS Medical laboratory reports, textual information extraction, text detection, text recognition.

I. INTRODUCTION

The medical laboratory report is one kind of important clinical data, which helps health care professionals with patient assessment, diagnosis, and long-term monitoring. The digitization process of healthcare services has been introduced into European countries under study during the last ten years. It has already reached excellent levels in some countries, especially in Northern Europe [1], [2]. In North America, the US government has also granted the substantial federal financial incentives to promote the adoption and use of electronic health records (EHRs) [3]–[5]. However, the situation may be different in developing countries, where paper documents are still common for health reports and records in hospitals. Taking China as an example, nearly 30% of tertiary hospitals still have paper-based or stand-alone computer systems, and another 30% have only basic systems that cannot share data among departments and hospitals [6]. Based on the above background, the purpose of our work is making papery medical laboratory reports digitalized for EHR system, which mainly relates to optical character recognition (OCR) techniques, especially text detection and recognition [7].

Though OCR is well-established for certain applications, text detection and recognition still face many challenges, such as the diversified requirements in different scenes (e.g., texts in street scene for robot navigation and receipts OCR for financial departments) and lower quality or degraded data (e.g., scanned legacy books in Google Books service) [7]. This work focuses on the digitization of documents in the medical scene. The most significant challenge to apply a text detection model to a documental image is that the image usually has a high resolution and many textual objects, while the single textual object occupies a very small region. It requires more memory to store the model's variables and takes more time to train and test the model when processing such a large image. A common operation [8], [9] for this problem is to resize the large image into a small scale. As shown in Fig. 1, generic objects can still keep saliency when they are resized twice or four times smaller. However, for a documental image, texts can be blurry and hard to be detected if they are resized into such small scales. Because a single text occupies a small region and can be recognized in a small

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

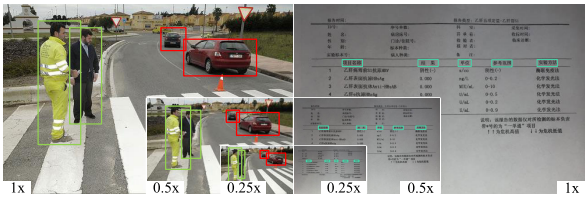

FIGURE 1. The outlines of pedestrians and cars on the left image can still be recognized when the image is resized twice or four times smaller. While each textual object occupies a small region on the right image, which becomes blurry and hard to be distinguished on the resized images.

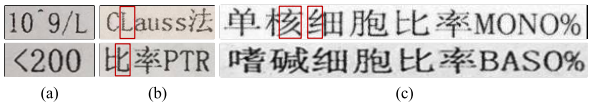

FIGURE 2. Images of text samples on medical laboratory report. The column *(a)* represents the texts that mix numbers and mathematics symbols. The column *(b)* contains Chinese and Latin characters. The last column *(c)* represents the typical long texts including multi-lingual characters and symbols. In the column *(b)* and *(c)*, two groups of strokes that have local similarity are bounded with red box.

image patch, this work proposes a patch-based strategy to cope with the challenge occurring during text detection.

Another challenge is brought by multi-lingual texts. The texts in a medical laboratory report often contain more than one kind of characters. In our experiments, besides numbers and symbols, the characters are mainly from Chinese and Latin. Chinese characters usually have a complex structure that consists of several parts. Some of these parts are basic strokes that are similar or identical, which bring difficulty for text recognition. Fig. 2 gives two groups of similar strokes that are bounded with red boxes. For text recognition, most existing approaches focuses on a single language, which is probably due to insufficient data [7], [10]. In this work, a concatenation structure is proposed to solve this problem, which can merge the features from both shallow and deep layers in the neural network.

In this work, a deep learning approach is presented to detect and recognize texts from a laboratory report image. In this approach, a patch-based strategy and a concatenation structure are proposed to handle the problems mentioned above. Specifically, an input documental image is cropped into patches firstly. Then a detector searches textual objects on each patch and outputs a set of predictions. The predictions from all patches are integrated as the final detection results. The module of text recognition is constructed based on CRNN (Convolutional Recurrent Neural Network [11]) and improved through a concatenation structure. For each detected textual object, the text recognizer outputs a text sequence directly. Because mobile devices have been more popular than before, we evaluate the proposed approach on a dataset with both scanned and phone-captured images. The results demonstrate that the proposed approach can effectively detect and recognize texts from medical laboratory reports. The contributions of this work are summarized as follows:

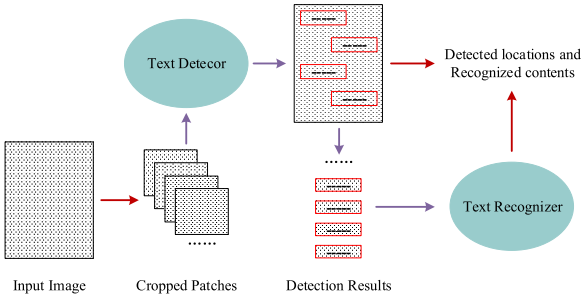

FIGURE 3. The pipeline of our approach. First, the input image of a medical laboratory report is cropped into small patches. Then the text detector searches textual objects on each patch and outputs their locations. According to detection results, the text recognizer takes each text area as input and predicts the contents within it.

- A patch-based strategy is used for text detection on documental images with high resolution and this strategy results in a high recall and precision.
- A concatenation structure is proposed that combines the features from two adjacent convolutional layers and brings a significant improvement in a multi-lingual scene.
- A deep learning approach is presented for text detection and recognition from images of medical laboratory reports.

II. RELATED WORK

A. TEXT DETECTION

The early approaches [13]–[16] mostly follow a bottom-up pipeline that applies artificial features [17]–[19] to detect strokes or characters. The individual character or combined strokes are directly classified in the recognition period or constructed into a line for text line verification in text detection. However, their performance relies on the results of character detection, and the extracted features are not robust to distinguish strokes or characters in different scenes (e.g., various fonts and degraded images).

Recent works in text detection are mostly inspired by scene object detection [20]–[22] and semantic segmentation [23], [24]. These methods can be categorized into bounding box regression based methods, segmentation based methods, and combined methods. Bounding box regression based methods [8], [9], [25], [26] treat each textual area as a kind of object and directly predict its bounding box and classification. Segmentation based methods [27]–[29] try to segment text regions from the background and output the final bounding boxes according to the segmented results. Combined methods [30] use a similar approach like Mask R-CNN [24], in which both segmentation and bounding box regression are used for better performance. However, combined methods are time consuming because more steps are involved. Among the three kinds of methods, bounding box regression based methods are the most popular in scene text detection and we also adopt this kind of method.

Bounding box regression based methods can be divided into one-stage methods and two-stage methods. One-stage

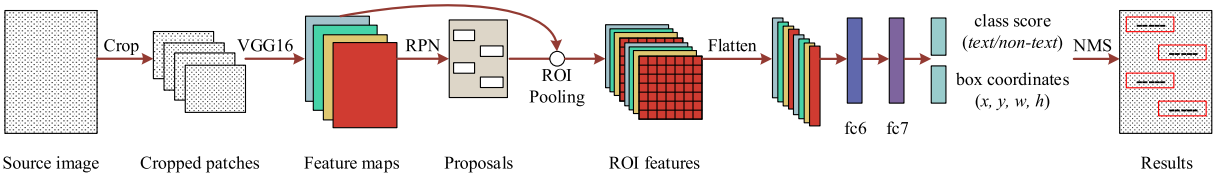

FIGURE 4. Overview of our detection pipeline. The network inherits from faster RCNN [12] architecture.

methods [9], [26] directly output detection results at several grids that correspond to the specific locations on feature maps. These methods often have faster speed but lower accuracy. Two-stage methods [8], [25] first apply CNN (Convolutional Neural Network) to extract features and generate a sparse set of candidate proposals that are supposed to contain all texts and filter out the majority of negative candidates. In the second stage, each candidate proposal is classified into one specific class and more accurate location is conducted through the learned bounding box regression. Considering the sequence characteristic of text, CTPN (Connectionist Text Proposal Network [8]) combines CNN and RNN (Recurrent Neural Network) to detect sequential features. EAST (An Efficient and Accurate Scene Text Detector [31]) is another two-stage detector, where a FCN-based (Fully Convolutional Networks [23]) pipeline is devised to merge the features from each convolutional layer. Besides class score and axis-aligned coordinates, EAST also outputs text rotation angle and quadrangle coordinates. Our text detector also adopts the two-stage design. But different from these methods that aim to natural scene images (COCO-Text [32], ICDAR 2013 [33] and ICDAR 2015 [34]), our method uses a patch-based strategy to address the challenge happened on images of medical laboratory reports, in which the image usually has a high resolution and many small textual objects.

B. TEXT RECOGNITION

Traditional methods [35]–[37] recognize characters individually and then group them into words. They explore low-level features, which are not robust to identify complex structures without context information. Then Wang, *et al.* [38] developed a CNN-based feature extraction framework for character recognition that achieved a better result than the methods [37], [39] with artificial features in individual character recognition. However, it is challenging to segment single characters because of the complicated background and inconsistent character spacing, e.g., Chinese character spacing is usually larger than Latin character spacing. As a natural characteristic of language, the relationship among characters or words is an important cue to make a prediction. CRNN [11] utilizes a sequential model to learn this relationship, which combines CNN and RNN for visual feature representation. Then, the CTC (Connectionist Temporal Classification [40]) loss is connected with the RNN outputs for calculating the conditional probability of the predictions. Most recent works [41], [42] take this approach as skeleton and introduce attention mechanism. However, few research aim at the multi-lingual scene [7], [43], [44]. For images of medical laboratory reports, we devise a concatenation structure to solve the recognition problem caused by multiple languages.

III. OUR APPROACH

The pipeline of our approach is illustrated in Fig. 3. Given an image of medical laboratory report, we first detect text blocks on it with the proposed patch-based strategy. Then each detected textual object is cropped from the source image and fed into a text recognizer. Due to the problem of multi-lingual texts and limited real data, the recognizer is enhanced with the proposed concatenation structure and trained on a synthetic dataset. The output for one source image contains the localizations and contents of all detected texts.

A. TEXT DETECTION IN DOCUMENT IMAGE

The supervised feature learning and end-to-end training procedure make it easy to transfer neural network methods to other applications. We adopt a two-stage architecture that is originally used for generic object detection. The patch-based strategy is applied to this architecture for text detection. Multiple optimizing methods are also adopted in this work to improve the performance.

1) NETWORK ARCHITECTURE

The detection module in our work is built based on Faster RCNN [12] architecture. As shown in Fig. 4, an input image first goes through a VGG16 [45] network to extract a group of feature maps. Second, a region proposal network (RPN) takes these feature maps as inputs and proposes axis-aligned bounding boxes that have more overlap areas with the ground-truth boxes. Then according to the locations of proposals, region-of-interest (ROI) pooling extracts the features from the previous feature maps and transforms them into fixed size (7×7 in our experiments). Third, these ROI features are flattened and pass through two fully connected (fc) layers. At last, the output layer, connected with fc layers, calculates the loss of text/non-text classification and bounding box regression.

2) TRAINING WITH PATCH-BASED STRATEGY

To begin with, the source image is cropped into small patches by a sliding window. The maximum length among textual objects is selected as the width of the sliding window.

The aspect ratio of the sliding window is set to 3:4. The horizontal and vertical strides are one-tenth of the width and height of the sliding window, respectively. These patches are randomly sampled and grouped into mini-batches, which are then sent to the detection network. During the training stage, the original coordinates of ground-truth bounding boxes are realigned to the axes of the corresponding patches. After getting feature maps, the RPN proposes a set of possible regions that may exist texts. In this period, the targets of these regions are a group of boxes named anchors. Every center of anchors is associated with one location in the feature maps, which can be calculated through the network architecture. The anchors are labeled according to their intersection-over-union (IoU[1]) with the ground-truth. We apply online hard negative mining [46] for the selection of anchors. Negative anchors with low IoU are abandoned and the ratio between negatives and positives is 1:1. The loss function is the same with [12]. At last, non-maximum suppression (NMS) [47] is used to post-process detection candidates to get the final results. During inference, all patch-based predictions will be aligned to the axes in the original image and NMS is used again to filter the overlapping bounding boxes.

B. MULTI-LINGUAL TEXT RECOGNITION

Deep convolutional networks can learn high-level features through successive convolutions. Recent studies [23], [48], [49] show that the features from shallow layers are also important in image classification, object detection, and semantic segmentation. Inspired by these works, we also take the strategy that combines the features from both deep and shallow layers to solve the local similarity problem between multi-lingual characters. In this section, we implement this idea by introducing a concatenation structure that learns more distinguishing features from shallow layers.

1) NETWORK ARCHITECTURE WITH CONCATENATION STRUCTURE

In our network, the features from two adjacent convolutional layers will be concatenated together as the input of the third layer. See Fig. 5 as an illustration. The network takes CRNN [11] as skeleton. The convolutional configuration from *conv_1* to *conv_6* is set to 3×3 kernel size, one stride, and one padding. Every convolution connects with ReLU (Rectified Linear Unit) function. From the third layer, the input of each convolutional layer is the concatenation of its previous two layers' outputs. Average pooling is used here to squeeze the feature maps so that they will have the same width and height before concatenation. It is noted that this operation does not bring too many extra parameters compared with convolution or deconvolution. Except for the last convolutional layer, all changes of width and height occur in max poolings or average poolings. After the seventh layer, two BLSTM (Bidirectional Long Short-Term Memory) layers [50] along the horizontal direction are involved to predict a label sequence of characters.

[1]For one anchor and one ground-truth, $IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union}$

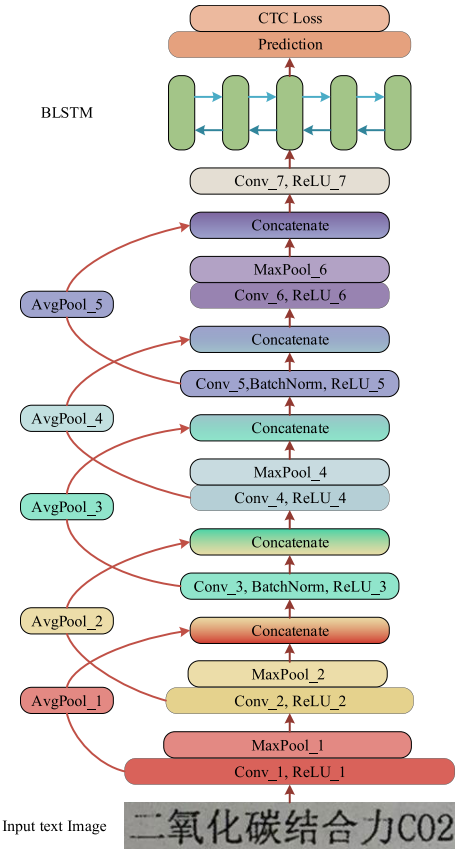

FIGURE 5. The recognition network in this work. The features from two adjacent convolutional layers are concatenated together as the input of the third layer. Before concatenation, the former feature map changes into the same width and height with the later by average pooling.

2) TRAINING

Given a batch of textual images, they are resized into $(h' \times w')$, where $h' = 32$, and w' is the maximum width among these images. Then the batch of images is fed into the network, which outputs a sequence of labels $\hat{y} = \hat{y}_1, \ldots, \hat{y}_m$. Each $\hat{y}_i \in \mathcal{D}$, where $\mathcal{D}$ is the dictionary that contains all characters in our task. Because the prediction may include incorrect labels, repeated labels, and 'blank's, we adopt the conditional probability defined in the Connectionist Temporal Classification (CTC) [40] layer to align the prediction and ground-truth. First, repeated labels and 'blank's are removed. Then, the conditional probability is defined as the sum of probabilities of all subsequence within $\hat{y}$:

$$p(\hat{y}|y) = \sum_{j=0}^{M} p(\hat{y}_{0:j}|y) \tag{1}$$

$$p(\hat{y}_{0:j}|y) = \prod_{i=0}^{J} p(\hat{y}_i^t) \tag{2}$$

where y is the label sequence of ground truth and $\hat{y}_{0:j}$ represents the subsequence of $\hat{y}$ from the 1st position to the jth position. In Eq. 2, $p(\hat{y}_i^t)$ is the probability that $\hat{y}_i$ has the same label t with y_i at the ith position, which is directly outputted from the network.

Denote the training set $\mathcal{X} = \{I_l, y_l\}_l$, where I_l is a training image and y_l is its ground truth label sequence. The objective is to minimize the negative log-likelihood of conditional probability for predictons when the corresponding ground truths are given:

$$\mathcal{L} = -\sum_{I_l, y_l \in \mathcal{X}} \log p(\hat{y}_l | y_l) \tag{3}$$

This objective function can directly calculate the cost value for one pair of prediction and ground truth so that the whole network can be trained by an end-to-end way.

IV. EXPERIMENTS AND RESULTS

The proposed approach is evaluated for both text detection and recognition. The experiments are conducted on an image dataset of medical laboratory reports. The details of metrics and implementation are presented in Section IV-A and Section IV-B, respectively. We give the experimental results and discussion in Section IV-C, Section IV-D, and Section IV-E, respectively.

A. DATASET AND METRICS

We conduct experiments on Chinese Medical Documents Dataset (CMDD) [51], which has three subsets ('scan', 'illu', and 'rota') classified by image capturing devices (i.e., scanners and smart phones) and conditions (i.e., illuminations and rotations). Because the 'rota' subset is not annotated, only the subsets of 'scan' and 'illu' are used in our work, and each of them contains 119 labeled documental images. Fig. 6 illustrates the layout of medical laboratory report in CMDD. The first line at the top of report lists the report time and test type. Next, a patient's private information about the medical test fills in the first table. The second table reports the details of test results. The physician's signature is at the bottom of report. The resolution of the images in CMDD is around 2500×3400. 50 to 150 textual objects distribute in such a documental image. There are totally 18402 textual instances and 351 different characters that contain Arabic numerals, mathematical symbols, Chinese, and Latin characters.

The performance of the detector is evaluated with **Recall**, **Precision**, **F1-measure**, and **Average Precision (AP)**. Recall reflects how many labeled texts are predicted correctly and precision tells us how many text predictions are correct. In experiments, one prediction is correct when both two criteria are satisfied: first, its class must be predicted as "text"; second, its IoU with one of the ground truths is over 0.6. F1-meansure and Average Precision (AP) are considered as tradeoff metrics.

For text recognition, both **Accuracy** and **mean Edit Distance (mED)** are taken into account for evaluation. In accuracy metric, the prediction of one test image is correct if

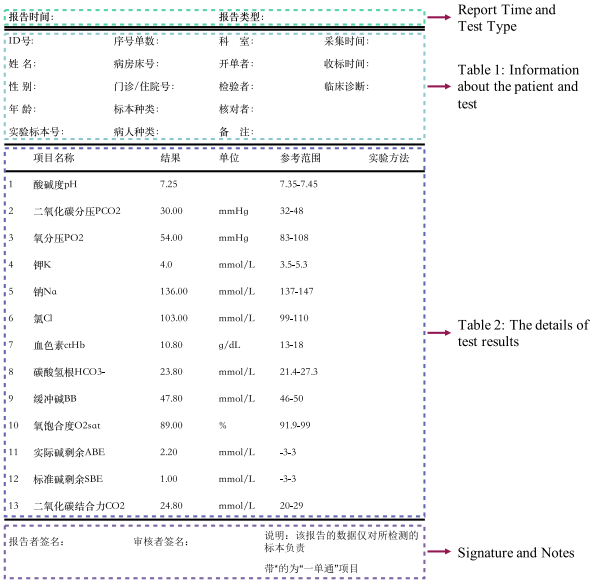

FIGURE 6. A template of medical laboratory report in CMDD.

and only if the predicted sequence is totally identical with the ground-truth label. However, this metric is not enough to evaluate the incorrect results. For example, if one ground truth is "hello", the model with prediction of "hel-o" has a better performance than the model with prediction of "he—". Therefore, we introduce mean edit distance to evaluate the models on incorrect predictions. Edit distance quantifies the similarity between one pair of prediction and ground truth by counting the minimum number of operations required to transform one string into the other. In addition, we also report the size of every model.

B. IMPLEMENTATION DETAILS

We fork the implementation in [52] to build our text detector. NMS is applied for RPN and the threshold is 0.7. Before NMS, the number of anchor boxes is set to 12,000 for training and 6,000 for testing. After NMS, 3,000 and 500 anchor boxes are preserved for training and testing, respectively. The Momentum optimizer is adopted for training. The learning rate and momentum are set to 0.001 and 0.9, respectively. The ratio between train set and test set is 8:2 and no extra data are added. We train the whole network for 30,000 iterations and divide the learning rate by a factor of 10 after the ten-thousandth iteration. The detection results on image patches are aligned to the source image and then evaluated with the ground truth on an intact image. All comparison methods in text detection experiments are pre-trained on ICDAR 2013 dataset [33] and then fine-tuned on CMDD.

In the experiments of text recognition, we use the Adam optimizer to train the whole network. The learning rate and beta1 are set to 0.0001 and 0.5, respectively. The network is initialized with values randomly drawn from a normal distribution having a mean of 0.0 and a standard deviation of 0.02. The comparison methods also adopt the same way

TABLE 1. Comparison results of text detection. The resolution of source image is around 2500 × 3400. The zoom scales are set to ×0.40, ×0.50, and ×1.00.

Method	Patch-based Strategy	Training Resoluiton ($\times n$)	Recall	Precision	F1-measure	AP
Faster RCNN [12]		×0.40	68.3%	98.4%	80.7%	63.6%
		×0.50	81.8%	**99.6%**	89.8%	81.7%
CTPN [8]		×0.40	70.9%	82.6%	76.3%	63.3%
		×0.50	84.5%	93.5%	88.8%	79.3%
	✓	×1.0	94.6%	56.8%	71.0%	85.8%
EAST [31]		×0.50	86.9%	99.4%	92.7%	81.4%
	✓	×0.50	95.8%	94.7%	95.2%	90.0%
	✓	×1.0	96.8%	98.8%	97.8%	**90.9%**
Ours	✓	×0.40	95.1%	96.4%	95.7%	90.9%
	✓	×0.50	97.4%	98.6%	98.0%	90.9%
	✓	×1.0	**99.5%**	98.6%	**99.1%**	**90.9%**

for initialization. Both the proposed recognizer and comparison methods use one million synthetic textual images for training due to the unbalanced distribution of the limited data in CMDD. The synthetic images are generated through FreeType library[2]. Because this method does not aim at a general text recognizer, the lexicon used for image generation has the same character set as CMDD. In order to make the distribution of synthetic data close to the real data, Gaussian noise is added according to the mean values and standard deviations of the image foreground and background in CMDD training set. We have released the source code[3] for synthetic images generation. The recognition model is trained for 6 epochs and all cropped textual images from CMDD are tested after every 5000 iterations. We compute the mean for every metric in the last ten epochs as the final results to report. The proposed approach is implemented on two GPUs (GeForce GTX TITAN Xp) and takes about ten hours for training. The source code and data[4] have been released for testing the proposed methods.

C. TEXT DETECTION

As mentioned in Section I, characters may be blurry when the documental image is resized into a small scale. The proposed strategy for detector training can preserve the original information as much as possible. In order to verify the influence of image resolution on text detection, we conduct the first experiment by resizing the input image into different resolutions. The resolution of source image is around 2500 × 3400. The zoom scales are set to ×0.4 and ×0.5. Faster RCNN [12] and CTPN [8] are taken as comparison methods in this experiment. The main difference between these two models is that RNN layers are connected with CNN in CTPN. RNN layers explore the sequence relationship among the CNN feature maps, which makes CTPN more suitable to detect horizontal texts in documental images. From the qualitative visualization in Fig. 7, it can be noticed that the performances for all of the methods improve significantly when the input image resolution becomes larger. The quantitative experimental results are reported in Table 1. CTPN achieves better performance in recall than the Faster RCNN (70.9%:68.3%, 84.5%:81.8%), which owes to the application of RNN layers. In addition, Faster RCNN achieves good results in precision (98.4%, 99.6%) while recall improves from 68.3% to 81.8%. This reflects that some characters become indistinguishable for text detector with the image resolution becoming small.

We also apply the patch-based strategy to CTPN and EAST [31] in the second experiment. The experimental results are listed in Table 1. When the zoom scale is set to 0.5, EAST improves 8.9% in recall with 4.7% decrease in precision after using patch-based strategy. When the cropped patches keep original resolution, EAST achieves 96.8% in recall and 98.8% in precision. The patch-based strategy does not bring improvement to CTPN in precision. According to the qualitative visualization in Fig. 7, the reason is that RNN is sensitive to character spacing and the patches cut off the original text sequence, which confuses CTPN to make accurate predictions. Among all experimental results, our detector has the best performance in recall (99.5%), F1-measure (99.1%), and AP (90.9%) when keeping the original resolution of cropped patches. Further more, our detector still has good results (recall: 95.1%, precision: 96.4%) even though the patch resolution is reduced to 40%. The visualization of results in Fig. 7 shows that our method can detect almost all textual objects with rare false positives.

The experiments in this section show that the resolution of input image affects the text detection results for comparison methods. The patch-based strategy can bring much more improvements, especially in recall.

D. TEXT RECOGNITION

The core of the proposed recognition model is merging two adjacent layers' features as the input of the next layer. To explore which layer's features conduce to the performance, we conduct the first experiment by deploying the concatenation structure in different layers. We make *ours_Li* denote the model that the concatenation structure is inserted from the *i*th layer to the sixth layer. As reported in Table 3, *ours_L5* has the highest accuracy that reaches to 95.8%. The second highest accuracy is from *ours_L2*, which is 95.4%. The mean edit distances of these two models are 3.29 and 3.30, respectively. With more

[2]www.freetype.org
[3]https://github.com/VisintLab-BJTU/GenTextImageBlocks
[4]https://github.com/xuewenyuan/OCR-for-Medical-Laboratory-Reports

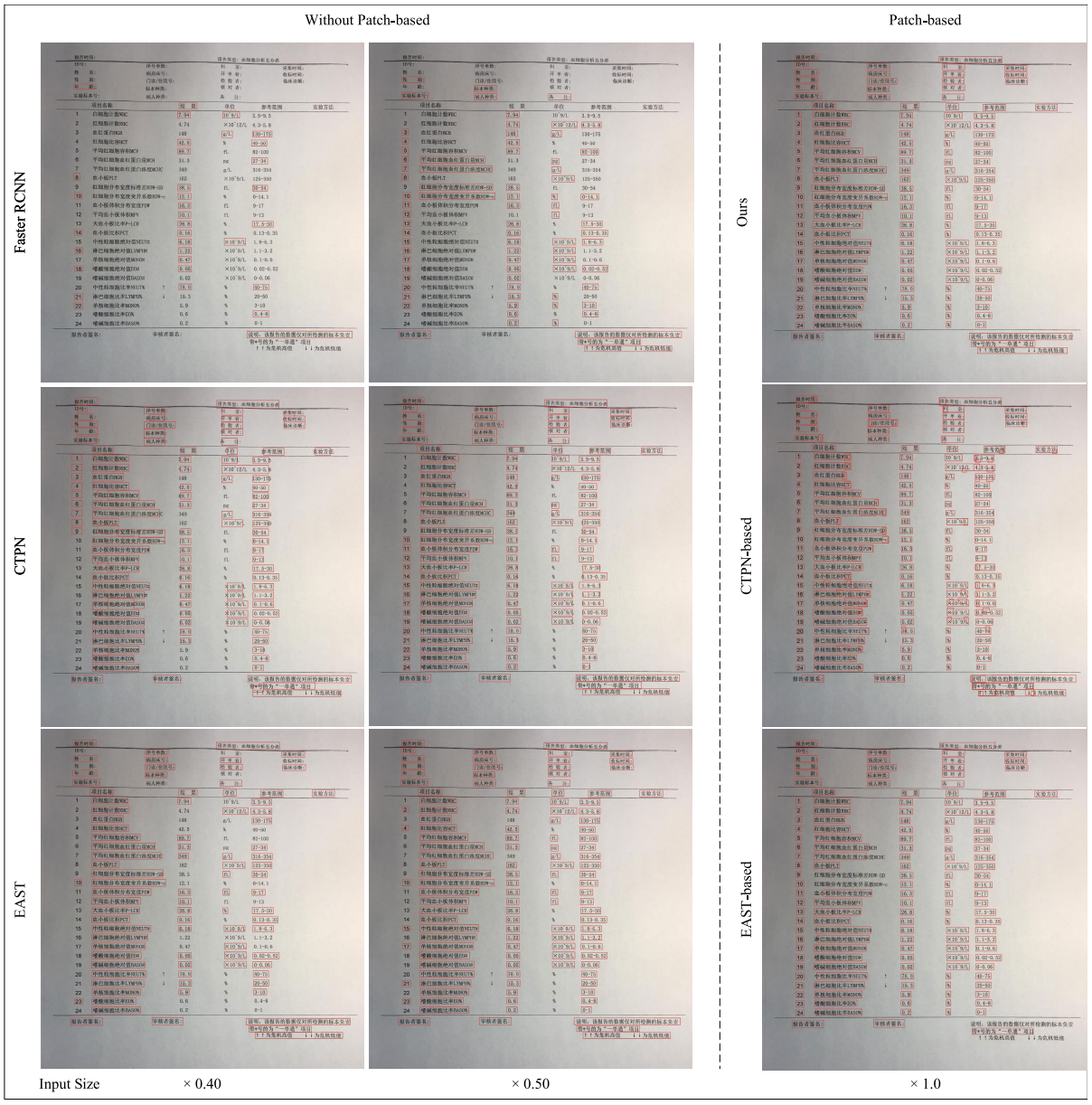

FIGURE 7. Some qualitative results on the experiments of text detection. The detection results are bounded with red boxes. In the experiment without patch-based strategy, all of the methods have a better performance with the input size increasing. CTPN is sensitive to character spacing and small characters are easily ignored. When applied the patch-based strategy, all of the methods get a better result except that many overlapping boxes occur in the result of CTPN. Zoom in the figure for more details.

TABLE 2. Text detection results on the multi-resolution test set where each original test image has five different resolutions. The results on the original test set are presented in parentheses.

Method	Patch-based Strategy	Training Resoluiton ($\times n$)	Recall	Precision	F1-measure	AP
Faster RCNN [12]		$\times$0.50	44.2%(81.8%)	39.4%(99.6%)	41.7%(89.8%)	38.4%(81.7%)
EAST [31]		$\times$0.50	87.0%(86.9%)	**99.4%**(99.4%)	92.7%(92.7%)	81.4%(81.4%)
	✓	$\times$0.40	92.8%(91.0%)	90.6%(88.8%)	91.7%(89.9%)	87.0%(84.6%)
	✓	$\times$0.50	96.8%(95.8%)	95.8%(94.7%)	96.3%(95.2%)	90.0%(90.0%)
	✓	$\times$1.0	97.6%(96.8%)	95.5%(98.8%)	96.5%(97.8%)	89.4%(90.9%)
Ours	✓	$\times$0.40	94.8%(95.1%)	95.4%(96.4%)	95.1%(95.7%)	90.8%(90.9%)
	✓	$\times$0.50	97.2%(97.4%)	96.8%(98.6%)	**97.0%**(98.0%)	**90.9%**(90.9%)
	✓	$\times$1.0	**98.8%**(99.5%)	95.3%(98.6%)	**97.0%**(99.1%)	90.8%(90.9%)

concatenation structure inserted into shallow layers, the model size increases gradually. *ours_L5* has a better tradeoff among the three metrics.

An easy way to add multi-scale features in CNN is concatenating the features from shallow layers to the last layers. We implement this method in the second experiment.

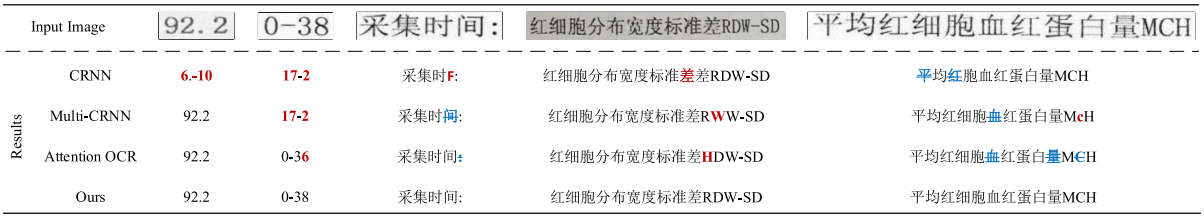

FIGURE 8. Sample qualitative results on the experiments of text recognition. The red characters are repeated or wrong predictions. The missing characters in prediction are marked as blue.

TABLE 3. Text recognition results. The top two results are highlighted.

Method	Accuracy	mED	Size (MB)
ours_L2	94.2%	**2.85**	49.4
ours_L3	**95.4%**	3.30	48.8
ours_L4	94.4%	3.75	47.6
ours_L5	**95.8%**	3.29	42.9
ours_L6	93.4%	3.22	**38.3**
multi-scale(2∼7)	94.2%	2.85	71.2
multi-scale(3∼7)	94.8%	3.50	59.2
multi-scale(4∼7)	93.7%	3.92	59.2
multi-scale(5∼7)	93.1%	3.76	53.9
multi-scale(6∼7)	92.8%	3.49	40.3
CRNN [11]	90.6%	3.79	**34.0**
Attention OCR [41]	83.8%	**2.51**	221.5

Because the aspect ratio of feature map in the last layer is different from that in previous layers, the previous feature map is resized through convolution so that the transformation can be learned during training period. Table 3 lists all the results in this experiment, in which *multi-scale(i∼j)* means that all feature maps from the *i*th layer to the *j*th layer are concatenated before they are delivered to the BLSTM layers. The experimental results also demonstrate that the features from shallow layers can improve the accuracy. However, the space occupation of this method is twice as large as that of our proposed model.

At last, we also compare the proposed model with CRNN [11] and the attention model [41]. The former is a classic recognition model from which many works developed. The later applies an attention model as a decoder on a "CNN+BLSTM" architecture. According to the results shown in Table 3, the accuracy of CRNN is 2% ∼ 5% lower than the models with fusion features. That means merging features is an effective way to improve the recognizer's performance. The attention model has the smallest mean edit distance while its accuracy is not good, which partly attributes to the attention mechanism.

Some qualitative results are presented in Fig. 8, where the wrong and missing predictions are marked as red and blue, respectively. In summary, all the three experiments show that the proposed recognition model can effectively utilize the shallow features and result in a higher accuracy and lower mean edit distance without too much space occupation.

E. TEST WITH MULTIPLE RESOLUTIONS

The image resolution in CMDD is around 2500 × 3400. The model trained on such a dataset may be overfitting. In order to verify the robustness of the proposed approach, we generate a multi-resolution test set where each original test image has five different resolutions. Each new image in the multi-resolution test set is obtained by resizing an original test image with a scale randomly drawn from (1.2 ∼ 0.7). This new multi-resolution set is tested with the models trained on the original train set. Table 2 and Table 4 report the text detection and recognition results on the multi-resolution set, where the results on the original test set are also presented in parentheses. For text detection, Faster RCNN and EAST are chosen as comparison methods for their good performance shown in Table 1. According to the results in Table 2, Faster RCNN is susceptible to the change of resolution. Our method and EAST perform better than Faster RCNN when facing different resolutions. Further, when the training images are resized 0.5 times, our method has the best result on this resolution. That means the training can be speeded up by resizing the image patch into a small scale with less performance loss. As for the text recognition on the multi-resolution test set, the results presented in Table 4 show that the accuracy of our method is 2% ∼ 3% lower than that tested on the original set, but higher than other comparison methods.

TABLE 4. Text recognition results on the multi-resolution test set. The results on the original test set are presented in parentheses.

Method	Accuracy	mED
ours_L3	**93.2%**(95.4%)	3.22(3.30)
ours_L5	**92.7%**(95.8%)	**3.12**(3.29)
multi-scale(2∼7)	89.92%(94.2%)	3.41(2.85)
multi-scale(3∼7)	91.41%(94.8%)	3.22(3.50)
CRNN [11]	88.97%(90.6%)	3.68(3.79)
Attention OCR [41]	85.98%(83.8%)	**2.07**(2.51)

V. CONCLUSION

This paper presents a deep learning approach for text detection and recognition from images of medical laboratory reports. Given an image of medical laboratory report, first, a patch-based training strategy is applied to a detector that outputs a set of bounding boxes containing texts. Then a concatenation structure is inserted into a recognizer, which takes the areas of bounding boxes in source image as inputs and outputs recognized texts.

In text detection experiments, image resolution can seriously affect the detection results. Our text detection module is enhanced through a patch-based strategy, which achieves

99.5% in recall and 98.6% in precision. The recognition experimental results demonstrate that the concatenation structure can effectively combine shallow and deep features and contribute to the recognition performance. In addition, the experiments on the multi-resolution test set show that the proposed approach has the ability to deal with images with different resolutions.

Although the presented approach can be further improved, it would benefit to reducing the cost of manual transcription for digitization of healthcare service in developing countries. The structured health records, which are recovered from document images, will be used for medical data mining to improve health services in our future works.

REFERENCES

[1] C. Rossignoli, A. Zardini, and P. Benetollo, "The process of digitalisation in radiology as a lever for organisational change: The case of the academic integrated hospital of verona," in *DSS 2.0—Supporting Decision Making With New Technologies*, vol. 261. Amsterdam, The Netherlands: IOS Press, 2014, pp. 24–35.

[2] S. Bonomi, "The electronic health record: A comparison of some European countries," in *Information and Communication Technologies in Organizations and Society*. Cham, Switzerland: Springer, Jan. 2016, pp. 33–50.

[3] A. K. Jha, C. M. DesRoches, P. D. Kralovec, and M. S. Joshi, "A progress report on electronic health records in US hospitals," *Health Affairs*, vol. 29, no. 10, pp. 1951–1957, 2010.

[4] M. B. Buntin, M. F. Burke, M. C. Hoaglin, and D. Blumenthal, "The benefits of health information technology: A review of the recent literature shows predominantly positive results," *Health Affairs*, vol. 30, no. 3, pp. 464–471, 2017.

[5] A. K. Jha, D. Doolan, D. Grandt, T. Scott, and D. W. Bates, "The use of health information technology in seven nations," *Int. J. Med. Inform.*, vol. 77, no. 12, pp. 848–854, 2008.

[6] T. Shu, H. Liu, F. R. Goss, W. Yang, L. Zhou, D. W. Bates, and M. Liang, "EHR adoption across China's tertiary hospitals: A cross-sectional observational study," *Int. J. Med. Informat.*, vol. 83, no. 2, pp. 113–121, 2014.

[7] Q. Ye and D. Doermann, "Text detection and recognition in imagery: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 7, pp. 1480–1500, Jul. 2015.

[8] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao, "Detecting text in natural image with connectionist text proposal network," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 56–72.

[9] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "Textboxes: A fast text detector with a single deep neural network," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 4161–4167.

[10] M. Bušta, Y. Patel, and J. Matas, "E2E-MLT—An unconstrained end-to-end method for multi-language scene text," Jan. 2018, *arXiv:1801.09919*. [Online]. Available: https://arxiv.org/abs/1801.09919

[11] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2298–2304, Nov. 2017.

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

[13] K. Subramanian, P. Natarajan, M. Decerbo, and D. Castanon, "Character-stroke detection for text-localization and extraction," in *Proc. 9th Int. Conf. Document Anal. Recognit. (ICDAR)*, vol. 1, 2007, pp. 33–37.

[14] H. Chen, S. S. Tsai, G. Schroth, D. M. Chen, R. Grzeszczuk, and B. Girod, "Robust text detection in natural images with edge-enhanced maximally stable extremal regions," in *Proc. 18th IEEE Int. Conf. Image Process.*, Sep. 2011, pp. 2609–2612.

[15] A. Mosleh, N. Bouguila, and A. B. Hamza, "Image text detection using a bandlet-based edge detector and stroke width transform," in *Proc. Brit. Mach. Vis. Conf.*, 2012, pp. 1–12.

[16] L. Neumann and J. Matas, "Text localization in real-world images using efficiently pruned exhaustive search," in *Proc. 11th Int. Conf. Document Anal. Recognit. (ICDAR)*, 2011, pp. 687–691.

[17] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 2963–2970.

[18] W. Huang, Y. Qiao, and X. Tang, "Robust scene text detection with convolution neural network induced MSER trees," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 497–511.

[19] S. Tian, Y. Pan, C. Huang, S. Lu, K. Yu, and C. Lim Tan, "Text flow: A unified text detection system in natural scene images," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4651–4659.

[20] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 580–587.

[21] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[22] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 21–37.

[23] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[24] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2961–2969.

[25] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2999–3007.

[26] M. Liao, B. Shi, and X. Bai, "TextBoxes++: A single-shot oriented scene text detector," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3676–3690, Aug. 2018.

[27] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai, "Multi-oriented text detection with fully convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4159–4167.

[28] D. Deng, H. Liu, X. Li, and D. Cai, "PixelLink: Detecting scene text via instance segmentation," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 6773–6780.

[29] S. Long, J. Ruan, W. Zhang, X. He, W. Wu, and C. Yao, "TextSnake: A flexible representation for detecting text of arbitrary shapes," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 20–36.

[30] P. Lyu, M. Liao, C. Yao, W. Wu, and X. Bai, "Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 67–83.

[31] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang, "EAST: An efficient and accurate scene text detector," in *Proc. IEEE Int. Conf. Pattern Recognit. (ICPR)*, Jul. 2017, pp. 2642–2651.

[32] A. Veit, T. Matera, L. Neumann, J. Matas, and S. Belongie, "COCO-Text: Dataset and benchmark for text detection and recognition in natural images," Jun. 2016, *arXiv:1601.07140*. [Online]. Available: https://arxiv.org/abs/1601.07140

[33] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. I. Bigorda, S. R. Mestre, J. Mas, D. F. Mota, J. A. Almazan, and L. P. De Las Heras, "ICDAR 2013 robust reading competition," in *Proc. 12th Int. Conf. Document Anal. Recognit. (ICDAR)*, Aug. 2013, pp. 1484–1493.

[34] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, and S. Lu, "ICDAR 2015 competition on robust reading," in *Proc. 13th Int. Conf. Document Anal. Recognit. (ICDAR)*, 2015, pp. 1156–1160.

[35] K. Sheshadri and S. K. Divvala, "Exemplar driven character recognition in the wild," in *Proc. Brit. Mach. Vis. Conf.*, 2012, pp. 1–10.

[36] A. Coates, B. Carpenter, C. Case, S. Satheesh, B. Suresh, T. Wang, D. J. Wu, and A. Y. Ng, "Text detection and character recognition in scene images with unsupervised feature learning," in *Proc. 11th Int. Conf. Document Anal. Recognit. (ICDAR)*, 2011, pp. 440–445.

[37] A. Mishra, K. Alahari, and C. Jawahar, "Top-down and bottom-up cues for scene text recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 2687–2694.

[38] T. Wang, D. J. Wu, A. Coates, and A. Y. Ng, "End-to-end text recognition with convolutional neural networks," in *Proc. IEEE Int. Conf. Pattern Recognit. (ICPR)*, Nov. 2012, pp. 3304–3308.

[39] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Nov. 2011, pp. 1457–1464.

[40] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proc. Int. Conf. Mach. Learn.(ICML)*, 2006, pp. 369–376.
[41] J. Poulos and R. Valle, "Character-based handwritten text transcription with attention networks," Dec. 2017, *arXiv:1712.04046*. [Online]. Available: https://arxiv.org/abs/1712.04046
[42] Z. Cheng, F. Bai, Y. Xu, G. Zheng, S. Pu, and S. Zhou, "Focusing attention: Towards accurate text recognition in natural images," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5086–5094.
[43] G. Nagy, "Twenty years of document image analysis in PAMI," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 1, pp. 38–62, Jan. 2000.
[44] 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.
[45] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," Sep. 2014, *arXiv:1409.1556*. [Online]. Available: https://arxiv.org/abs/1409.1556
[46] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
[47] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *Proc. Int. Conf. Pattern Recognit. (ICPR)*, vol. 3, Aug. 2006, pp. 850–855.
[48] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2261–2269.
[49] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.
[50] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, 2000.
[51] W. Xue, Q. Li, Z. Zhang, Y. Zhao, and H. Wang, "Table analysis and information extraction for medical laboratory reports," in *Proc. IEEE 4th Int. Conf. Cyber Sci. Technol.*, Oct. 2018, pp. 193–199.
[52] X. Chen and A. Gupta, "An implementation of faster RCNN with study for region sampling," Feb. 2017, *arXiv:1702.02138*. [Online]. Available: https://arxiv.org/abs/1702.02138

WENYUAN XUE received the B.E. degree in software engineering from Shanxi University, Taiyuan, China, in 2015. He is currently pursuing the Ph.D. degree with the School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China. His current research interests include OCR and document layout analysis, especially table recognition.

QINGYONG LI (M'09) received the B.Sc. degree in computer science and technology from Wuhan University, Wuhan, China, in 2001, and the Ph.D. degree in computer science and technology from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2006. He is currently a Professor with Beijing Jiaotong University, Beijing, China. His current research interests include computer vision and artificial intelligence.

QIYUAN XUE received the B.Med. degree in clinical medicine from Shanxi Medical University, Taiyuan, China, in 2008, and the M.Med. degree from Zunyi Medical University, Zunyi, China, in 2012. He is currently an Attending Physician with the Department of Burn and Plastic Surgery, The Fifth People's Hospital of Datong, Datong, China.