Original Article

DeepSSR: a deep learning system for structured recognition of text images from unstructured paper-based medical reports

Hao Liu[1], Huijin Wang[1], Jieyun Bai[2,3]^, Yaosheng Lu[2,3], Shun Long[1]

[1]Department of Computer Science, College of Information Science and Technology, Jinan University, Guangzhou, China; [2]Department of Electronic Engineering, College of Information Science and Technology, Jinan University, Guangzhou, China; [3]Guangdong Provincial Key Laboratory of Traditional Chinese Medicine Information Technology, Jinan University, Guangzhou, China

Contributions: (I) Conception and design: H Liu, J Bai; (II) Administrative support: H Wang; (III) Provision of study materials or patients: Y Lu; (IV) Collection and assembly of data: H Liu; (V) Data analysis and interpretation: H Liu; (VI) Manuscript writing: All authors; (VII) Final approval of manuscript: All authors.

Correspondence to: Jieyun Bai. Department of Electronic Engineering, College of Information Science and Technology, Jinan University, No. 601 West Huangpu Road, Tianhe District, Guangzhou 510632, China. Email: baijieyun@jnu.edu.cn; Huijin Wang. Department of Computer Science, College of Information Science and Technology, Jinan University, No. 601 West Huangpu Road, Tianhe District, Guangzhou 510632, China. Email: twanghj@jnu.edu.cn.

Background: Complete electronic health records (EHRs) are not often available, because information barriers are caused by differences in the level of informatization and the type of the EHR system. Therefore, we aimed to develop a deep learning system [deep learning system for structured recognition of text images from unstructured paper-based medical reports (DeepSSR)] for structured recognition of text images from unstructured paper-based medical reports (UPBMRs) to help physicians solve the data-sharing problem.

Methods: UPBMR images were firstly preprocessed through binarization, image correction, and image segmentation. Next, the table area was detected with a lightweight network (i.e., the proposed YOLOv3-MobileNet model). In addition, the text of the table area was detected and recognized with the model based on differentiable binarization (DB) and convolutional recurrent neural network (CRNN). Finally, the recognized text was structured according to its row and column coordinates. DeepSSR was trained and validated on our dataset with 4,221 UPBMR images which were randomly split into training, validation, and testing sets in a ratio of 8:1:1.

Results: DeepSSR achieved a high accuracy of 91.10% and a speed of 0.668 s per image. In the system, the proposed YOLOv3-MobileNet model for table detection achieved a precision of 97.8% and a speed of 0.006 s per image.

Conclusions: DeepSSR has high accuracy and fast speed in structured recognition of text based on UPBMR images. This system may help solve the data-sharing problem due to information barriers between hospitals with different EHR systems.

Keywords: Deep learning; paper-based medical reports; table detection; text detection; text recognition

Submitted Dec 13, 2021. Accepted for publication Apr 29, 2022.
doi: 10.21037/atm-21-6672
View this article at: https://dx.doi.org/10.21037/atm-21-6672

^ ORCID: 0000-0002-2847-350X.

Introduction

Paper-based medical reports are widely used in the medical system, and they vary in data, structures, and layouts (1). This is particularly true in China because hospitals have used different technologies to develop their medical information systems, making data exchange and information sharing very difficult (2). It is a common practice for patients to carry paper-based medical reports if they are to transfer to or seek help from another hospital, where the information is to be re-processed (read and manually input) by the staff before use. In addition, to collect big data required for medical research and practice, paper-based medical reports need to be entered into the system. This manual process is time-consuming, inefficient, and costly. There is a growing demand for the automatic processing of paper-based medical reports. To realize automatic processing of unstructured paper-based medical reports (UPBMRs), table detection, text detection, text recognition, and text box assignment should be done in sequence, and various methods are used to solve the sub-problems.

For table detection, heuristic-based methods and deep learning-based methods are mostly used. The former adopted a set of characteristic-based rules to analyze a given image to identify the table areas that meet specific criteria. The T-Recs system proposed by Kieninger *et al.* used a bottom-up clustering approach to detect the word segments within the image and then combined them according to some predefined rules to obtain the conceptual text blocks (3). Yildiz *et al.* developed the pdf2table system, which employs multiple heuristics to identify tables in PDF files (4). Koci *et al.* adopted a graphic model to represent the layout and spatial features of the potential forms within a page and then identified the form as a subgraph using a genetic algorithm (5). Overall, these above heuristic-based approaches have difficulties in identifying tables in practice, and their robustness remains a doubt (6). Alternatively, deep learning-based approaches have been proposed. Siddiqui *et al.* (7) applied deformable convolution to Faster Region-based Convolutional Neural Networks (Faster RCNN) (8) and Feature Pyramid Networks (9) to detect tables with arbitrary layouts. Sun *et al.* (10) proposed the concept of the corner (the table vertex used as the center of an area of a certain radius) and combined corner location into table detection based on Faster RCNN. Huang *et al.* (11) added anchor optimization strategy and post-processing method to the table detection model You Only Look Once, Version 3 (YOLOv3) (12). For anchor optimization, K-means clustering is used to find the exact location of the table, before additional blank and noise pages are removed from the prediction results. The table structure in the medical report is complicated with frame and frameless tables, making it difficult for rule- or heuristic-based table detection approaches. Deep learning-based approaches usually yield better performance (13).

For text detection, traditional text detection methods are to locate the text by designed features. For example, Mayan *et al.* compared the pixels of the original image against those of the template (14), Epshtein *et al.* extracted image edge features to generate a stroke width diagram for text detection (15), Yin *et al.* extracted the maximum stable extremum region (MSER) from the image as a candidate character region (16). However, the traditional text detection methods are prone to natural factors. In recent years, deep learning approaches have widely been adopted in text detection, which is mainly divided into two categories, namely bounding box regression and image segmentation based. In bounding box regression-based methods, text regions are treated as objects and their locations and categories are predicted. The Connectionist Text Proposal Network proposed by Tian used a Long Short-Term Memory network (17) to predict the text area and generate suggestions (18). Zhou *et al.* (19) proposed an efficient and accurate scene text detector (EAST) based on fully convolutional networks (20), which directly generates text regions, eliminating those redundant and time-consuming intermediate steps. In segmentation-based methods, text detection is treated as a classification problem of text and background. Kong *et al.* introduced a cyclic grouping model to map pixel embedding into n-sphere space, and segment each instance at the same time by predicting the embedding of all pixels at one time (21). Wang *et al.* predicted to predict the text region under different shrinkage scales and enlarged the detected text region iteratively until it collides with other instances (22). Liao *et al.* proposed a segmentation network with differentiable binary modules (DB), which improves both the accuracy and speed of segmentation (23). Despite yielding good performance in text detection of regular shapes, the bounding box regression-based approaches show difficulties in dealing with medical reports, which usually vary in size and shape in practice.

For text recognition, traditional text recognition

(24-26) recognizes separate characters before grouping them into words. They explore low-level features that cannot recognize complex structures without context information. At present, most text recognition approaches adopt deep learning algorithms. Wang *et al.* (27) proposed a feature extraction framework for text recognition, which performed well in single character recognition. However, due to the variance in the background and character spacing, the segmentation of single characters remains a challenge. The Convolutional Recurrent Neural Network (CRNN) model proposed by Shi *et al.* (28) uses the sequence model to learn the relationship between multiple characters. It combines Convolutional Neural Network (CNN) with Recurrent Neural Network (RNN) for visual feature representation and uses connectionist temporal classification (CTC) (29) to calculate the conditional probability for prediction. Cheng *et al.* proposed a string recognition model based on an attention mechanism to solve the problem of target character deviation in complex images (30). The end-to-end fast text location network proposed by Liu *et al.* shares features between text detection and text recognition (31). CRNN is an end-to-end trainable algorithm, which can recognize entire text sequences without splitting characters. In addition, it is not restricted by predefined dictionaries. Therefore, we decided to use the CRNN network for character recognition.

This paper presents a deep learning system [deep learning system for structured recognition of text images from unstructured paper-based medical reports (DeepSSR)] to extract and structure table information in medical reports. The system is mainly divided into four parts: image preprocessing, table detection, character recognition, and text box assignment. First, medical reports were preprocessed by graying, binarization, and image correction, so that the processed image was more suitable for subsequent table detection and character recognition than the original image. Next, a YOLOv3-MobileNet network model was used as the baseline network to detect the table area. Then, a model based on DB and a model based on CRNN were used for text detection and text recognition in the table area respectively. Finally, in the text box assignment phase, we associated the identified text boxes according to the row and column coordinates to obtain structured data. The contributions of this paper are the following:

- We proposed a framework based on deep learning to recognize and structure tabular information from medical reports;
- We proposed a YOLOv3-MobileNet for table detection in medical reports;
- We built a dataset of UPBMR images for table detection algorithms.

We present the following article in accordance with the TRIPOD reporting checklist (available at https://atm.amegroups.com/article/view/10.21037/atm-21-6672/rc).

Methods

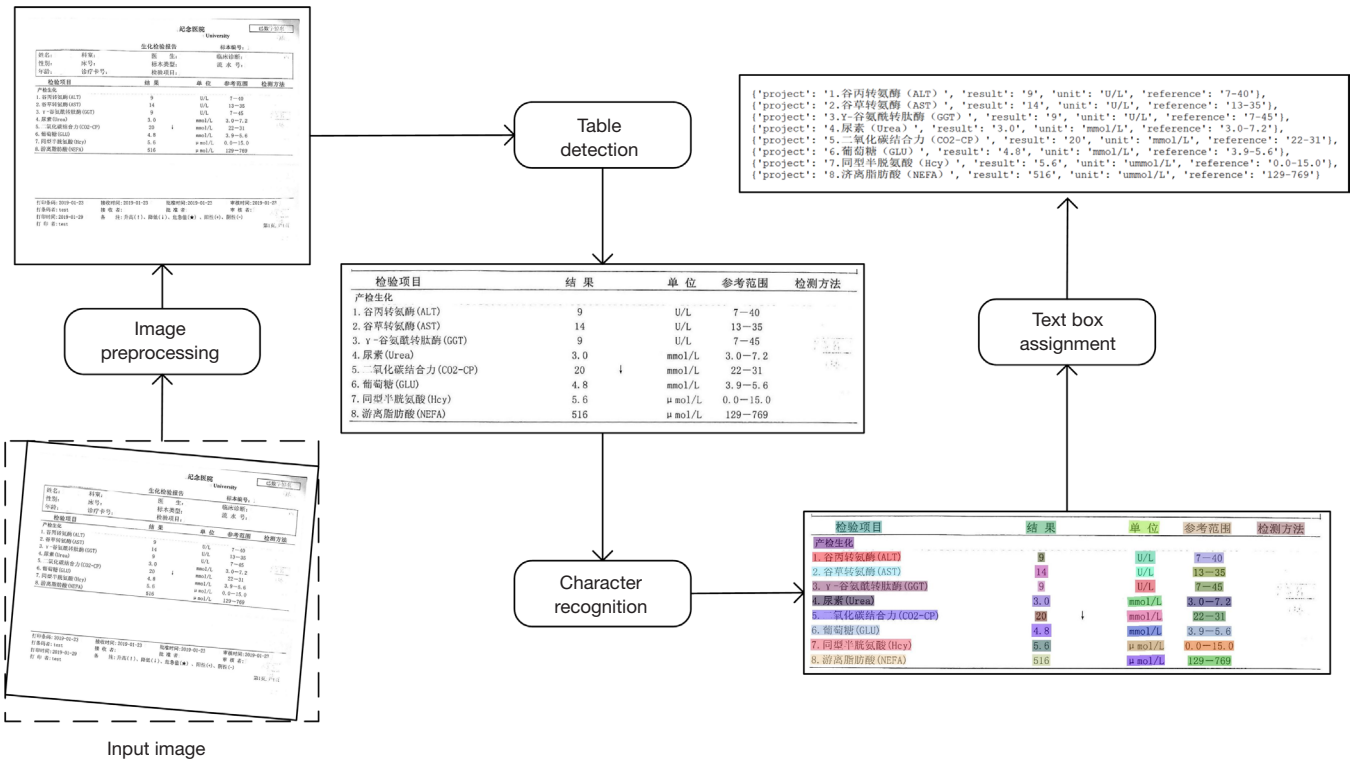
The pipeline of our steps in DeepSSR is illustrated in *Figure 1*. Given an image of paper medical reports, image preprocessing, which includes gray processing, binarization, and image correction, was first carried out. Second, the table area of the processed image was detected via a YOLOv3-MobileNet network model. Third, the text within the detected table was detected and recognized with the model based on the DB and the CRNN. Finally, the detected text aligned with row and column coordinates was structured according to the key-value format. The output for one image of paper medical reports was an electronic form with structured data records.

Image preprocessing

The quality of UPBMR images is usually affected by various factors such as brightness, shooting angle, and so on. To minimize these influences on subsequent processing (including feature extraction, table recognition, text recognition, and text structuring), preprocessing was applied to enhance and correct images. Image enhancement is to strengthen the table and text information, while image correction is to correct the image tilt angle for subsequent table detection and text structuring. More specifically, we adopted grayscale transformation of color images, binarization and image correction.

Grayscale transformation of color images

Photos taken by mobile phones, cameras, and other devices are usually color images, which require more computing resources than their grayscale counterparts for lightweight processing do. In addition, colors in these UPBMR images are usually affected by natural light and cannot provide important information for subsequent detection and



检验项目	结 果		单 位	参考范围	检测方法
产检生化					
1.谷丙转氨酶(ALT)	9		U/L	7—40	
2.谷草转氨酶(AST)	14		U/L	13—35	
3.γ—谷氨酰转肽酶(GGT)	9		U/L	7—45	
4.尿素(Urea)	3.0		mmol/L	3.0—7.2	
5.二氧化碳结合力(CO2-CP)	20	↓	mmol/L	22—31	
6.葡萄糖(GLU)	4.8		mmol/L	3.9—5.6	
7.同型半胱氨酸(Hcy)	5.6		μmol/L	0.0—15.0	
8.游离脂肪酸(NEFA)	516		μmol/L	129—769	

Figure 1 The pipeline of DeepSSR. DeepSSR, deep learning system for structured recognition of text images from unstructured paper-based medical reports.

recognition. To minimize the interference and promote the calculation speed (32), the grayscale transformation was applied. We focused on images composed of images of three channels in Red-Green-Blue, with different values from 0 to 255 respectively. The grayscale transformation was applied to convert these color images into gray ones composed of only one channel. These transformation methods are usually based on component, maximum, average, and weighted average. Our approach adopted the weighted average method, that is, the values of the three RGB (red, green, and blue) channels are manipulated as the following formula (33):

$$Y=0.333\times Fr+0.5\times Fg+0.1666\times Fb \quad [1]$$

where Y is the gray value of the transformed image, Fr, Fg, and Fb are the gray values of R, G, and B of the original color image, respectively.

Binarization of grayscale images

The quality of the captured images may vary because of unclear ink and inconsistent definition. They must be first binarized to obtain a black-and-white image with only 0 (black) and 255 (white) image pixels so that the foreground (including the text and form structure) and background can be divided in a light-weighted manner. To achieve this, a threshold should be determined for image binarization (i.e., the pixels greater than the threshold are set to be 255, whereas the pixels less than the threshold are set to be 0) in either a global or local method (34). Our approach adopted a maximum interclass variance method [i.e., Otsu algorithm (35)] which maximizes the square root of the average gray levels of the whole image and its foreground and background regions.

Image correction based on the inclination angle

The inclination may be introduced when the medical reports are digitized manually, which affects the detection of table area and in turn the accuracy of character recognition. The tile angle of the image must be calculated to correct the image. Popular image correction algorithms include Hough transform (36), side horizontal projection (37), straight-line

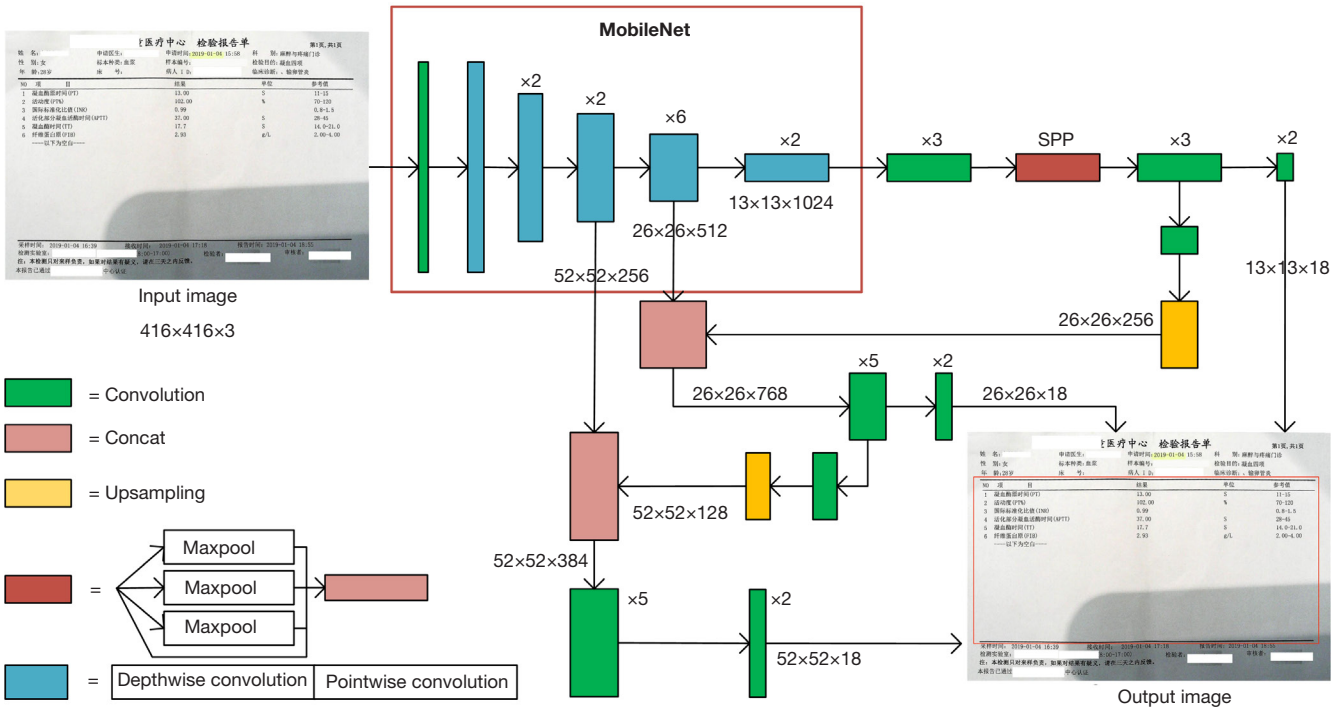


Figure 2 The architecture of YOLOv3-MobileNet. MobileNet, Efficient Convolutional Neural Network for Mobile Vision Application; SPP, spatial pyramid pooling.

fitting (38), and Fourier transform (39). Hough transform was adopted in our approach because it could provide good results even in the presence of noise and occlusion.

Hough transform was applied to the center of the images to reduce the computational complexity and improve accuracy. More specifically, the Hough transform was used to detect the lines in the image before the inclination angles of these lines were calculated. Note that the inclination angle of the image did not exceed 45 degrees. Therefore, those lines with an inclination angle of fewer than 45 degrees were retained, and the overall tilt angle of the image was their average inclination angle. Finally, the image was rotated accordingly.

Table detection

After preprocessing, an image with information enhanced could be obtained. The table area was detected and its position was extracted via YOLOv3 which is one of the most widely used anchor-based one-stage architectures. YOLOv3 turns the target detection problem into a regression problem to detect the boundary box and category in the image. It is mainly composed of the feature extraction network DarkNet-53 and multi-scale prediction (12).

Although YOLOv3 has high accuracy on small objects, it still has shortcomings. First, the deployment speed of DarkNet-53 networks on embedded devices (e.g., NVIDIA® Jetson AGX Xavier) is slow. Second, the multi-scale prediction has poor extraction of local features on a single convolutional layer.

We proposed a model named YOLOv3-MobileNet for table detection. The Efficient Convolutional Neural Network for Mobile Vision Application (MobileNet) has shown its advantages in small size, low computation cost, and high speed (40). Therefore, YOLOv3-MobileNet replaced DarkNet-53 with MobileNet to achieve lightweight. Spatial pyramid pooling (SPP) uses multiple pooling windows to process the same image at different scales to achieve multi-scale local area feature fusion (41). Therefore, YOLOv3-MobileNet introduced SPP to improve the accuracy of big objects detection by combining global and local multi-scale features.

Figure 2 shows the structure of the YOLOv3-MobileNet-based detection module. First, images are scaled to a unified

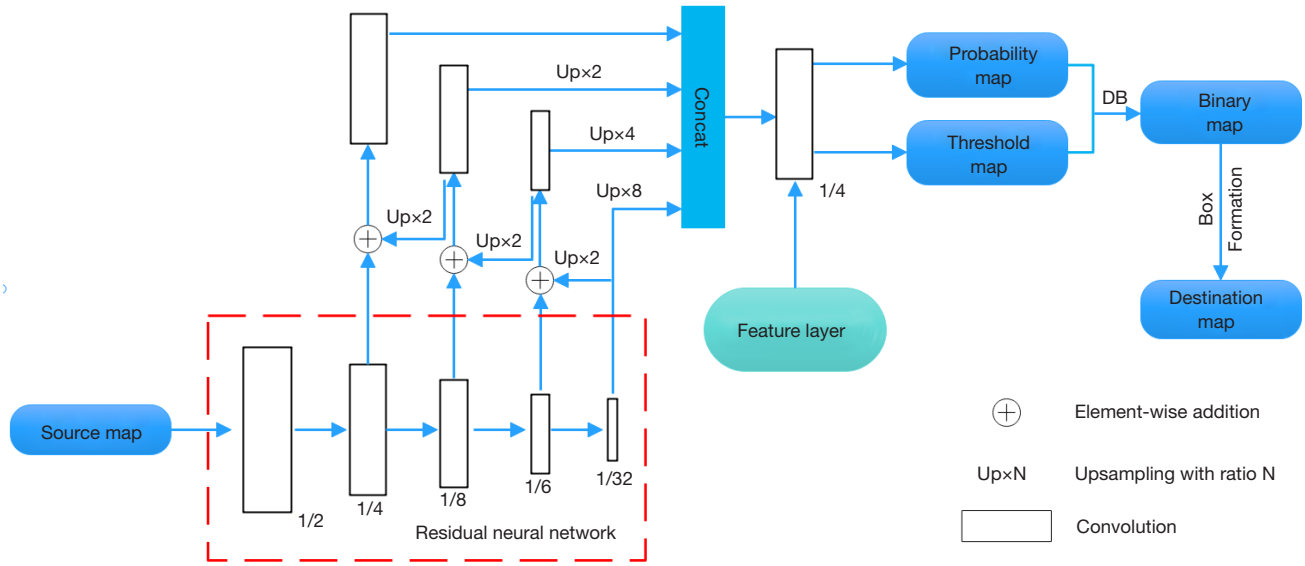


Figure 3 The architecture of the text detector DB. DB, differentiable binarization network; N, the upsampling factor.

size (416×416). Then, the features were extracted via a MobileNet which decomposed the standard convolution into depthwise and pointwise convolutions to reduce the computational cost, without sacrificing accuracy and speed. The depthwise convolution was responsible for single filtering every input channel, and the pointwise convolution was responsible for combining the output of the depthwise convolution. Finally, the multi-scale detection network extracted the last three feature layers of MobileNet for convolution prediction. The first layer extracted the 52×52 feature map for detecting small targets, the second layer extracted the 26×26 feature map for detecting medium targets, and the last layer extracted the 13×13 feature map for detecting large targets. SPP was doped after the third feature layer, which used multiple windows to pool feature maps to obtain fixed-size feature vectors. The final prediction was obtained by splitting the output of all three different feature maps.

Richer semantic information is usually obtained via more layers and smaller feature maps. A low-level feature map has greater resolution and unveils more details, which helps to locate objects. A multi-scale detection network integrates the high-level and low-level features and predicts them on a multi-scale feature map, which improves the ability of the model to detect the surface area.

We used the anchor box mechanism proposed by Faster RCNN for target detection from the feature map and used the K-means algorithm to cluster the size of the real bounding boxes in the training set. The obtained anchor box size has better a priori than the manually selected size does. The model obtained 9 sizes of bounding boxes through the K-means algorithm, of which 3 with the largest size were allocated to the feature graph with the size of 13×13, 3 with the medium size were allocated to the feature graph with the size of 26×26, and the remaining 3 with the smallest size were allocated to the feature graph with the size of 52×52. For the output results, non-maximum suppression was used to filter the invalid or redundant bounding boxes, and the final detection results were therefore obtained.

Character recognition

Text detection

Once the table was detected, the content of the table must be recognized, and the key step of content recognition was the detection of the text area. Here, we used the DB algorithm to detect the text region of the image.

Figure 3 shows the architecture of the text detector DB. First, the DB model used a residual neural network to extract image features, converted the feature output to the same size through upsampling, and cascaded to generate a feature layer. Then, the text probability map and dynamic threshold map were calculated through the feature layer. Finally, the binary map was generated by the text probability

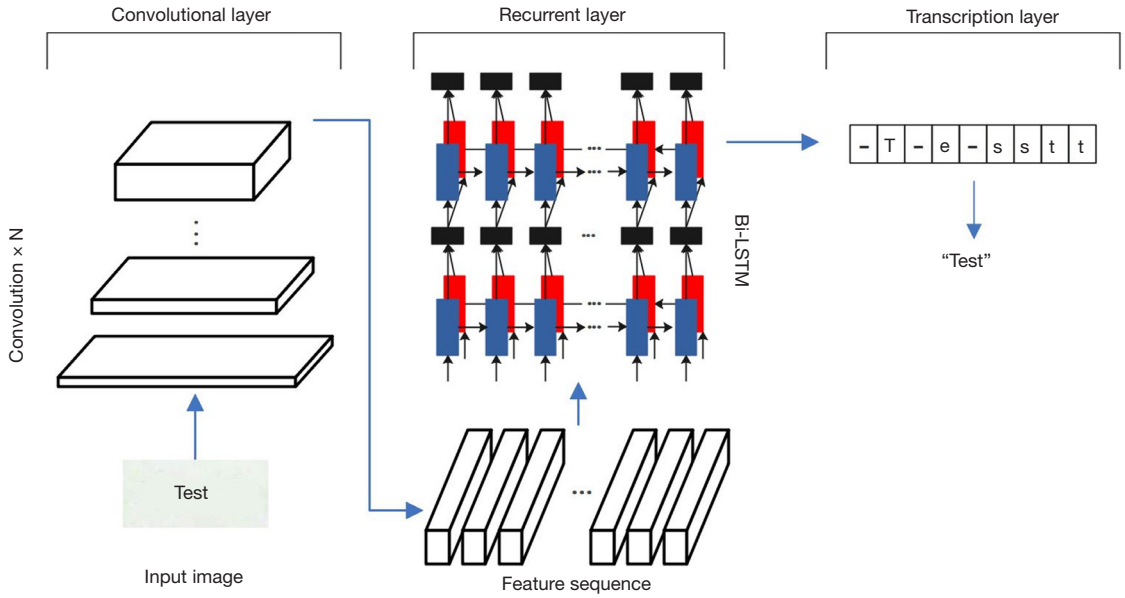


Figure 4 The architecture of the text recognizer CRNN. CRNN, convolutional recurrent neural network; Bi-LSTM, bi-directional long short-term memory network; N, the number of convolutions.

map and dynamic threshold map, and converted into text boxes using heuristic techniques.

After obtaining the probability map, the traditional method calculated the binary map through a fixed threshold. The DB algorithm predicted the threshold of each position in the image through the network to generate a dynamic threshold map. The results calculated according to the dynamic threshold were more suitable for complex and changeable detection scenarios.

Text recognition

After the text area was identified, the texts within could be recognized. CRNN was adopted for text recognition.

The network structure of CRNN, as shown in *Figure 4*, consists of three parts, including a convolutional layer, a recurrent layer, and a transcription layer. In the convolutional layer, a CNN is used to extract feature sequences from each input image. In the recurrent layer, a RNN is used to learn and predict the label distribution from the feature sequence. In the transcription layer, a CTC is used to convert the label distribution obtained by the recurrent layer into the final recognition result.

Text box assignment

Text box assignment was needed to structure the not-related-yet data obtained from the previous step.

First, the text boxes were sorted by their coordinates (as shown in *Figure 5A*). The abscissas and ordinates were sorted from top to bottom and then from left to right because text boxes might tilt with the paper. A fixed threshold of 1/3 of the text box height was set according to a priori experience. If the difference between the vertical coordinates of the text box was less than the threshold, the text boxes were considered of the same row, or they were divided into different rows if otherwise.

Similarly, we strung the text information into columns (as shown in *Figure 5B*). The range of the abscissa of the text box in the header row was used as the range of the current column. If there was an intersection between the abscissa of the text box and the header column, the text boxes were considered of the same column, or they were divided into different columns if otherwise.

Finally, the text information aligned with row and column coordinates was structured in a key-value manner. By analyzing the header information of medical reports, a dictionary with keys was generated. Multiple header information was compared with the dictionary to get keys, and other text information on the same column was considered as values. In this manner, the conceptual connections were created between text information, which was more convenient for later storage and analysis.

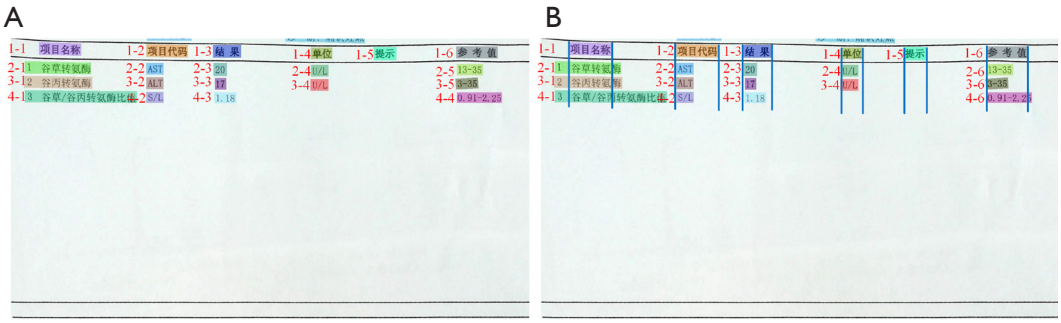


Figure 5 Text box assignment. (A) 1-1 represents the first text box of the first line. (B) The blue line represents the range of the current column and 1-1 represents the text box in the first row and first column.

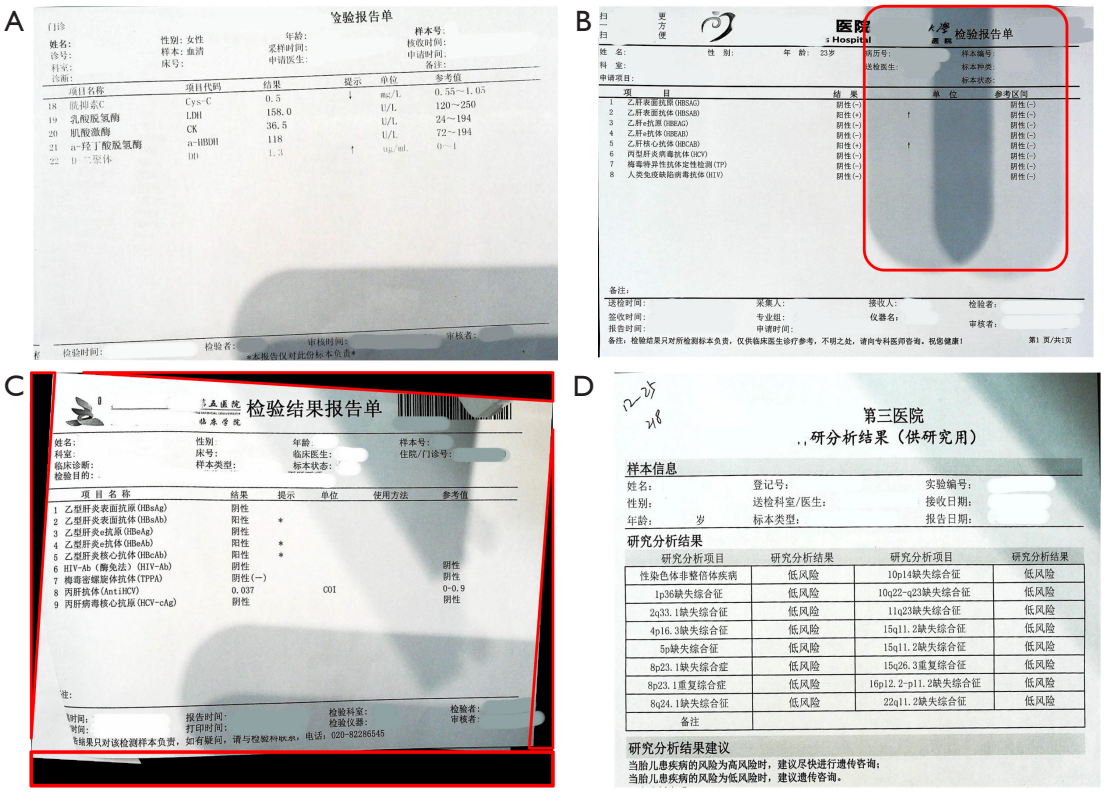


Figure 6 UPBMR images. (A) has a slope, (B) has a shadow, (C) has a black border. The structures of (A), (B), (C), and (D) are different. UPBMR, unstructured paper-based medical reports.

Dataset

A set of 4,221 UPBMR images between January 2020 and June 2021 were collected and used for table detection and structure recognition because the public dataset was not available because of ethical concerns. Some of these UPBMR images are shown in *Figure 6*.

For table detection, the dataset was first annotated via LabelImg and then divided into three subsets for training (80%), verification (10%), and testing (10%) purposes. A rectangular box was used to label a table within the image and the corresponding information such as its position and size was recorded in an XML file.

For structured data, we annotated the test set by manual input. In the annotation process, we extracted information in the image table according to the actual application scenario and stored the annotation information in the file in the form of key/value pairs.

Training

The hardware for model training is based on an Intel Xeon E5-2698 v4 CPU @ 2.20 GHz, 256 GB RAM, and NVIDIA Tesla V100 DGXS GPU with 32 GB memory.

For table detection, we put the training set into the training and used the verification set for verification in the training process to avoid overfitting. A max iterator of 50,000, a batch size of 32, a learning rate of 0.001, and a learning decay rate of 10 were set for training (42).

The Chinese and English ultra-lightweight text detection model and text recognition model trained by Du (43) were used in our training for text detection and recognition.

Evaluation standard

In the field of object detection, the average precision (AP) is mostly used as the evaluation metric of models. AP is calculated from precision (P) and recall (R). P refers to the probability of correct detection among all detected objects. R refers to the probability of correct identification among all positive samples. P and R are defined as follows:

$$P=\frac{TP}{TP+FP} \quad [2]$$

$$R=\frac{TP}{TP+FN} \quad [3]$$

where *TP*, *FP*, and *FN* denote true positive, false positive, and false negative. To get *TP* and *FP*, we need to use Intersection-over-Union (IoU), which is the ratio of the intersection and union of the prediction box and ground truth. If the IoU is greater than a threshold, it is considered *TP*, otherwise, it is considered *FP*.

Based on the P and R, the *AP* is defined in:

$$AP=\int_0^1 P(R)\,dR \quad [4]$$

For table detection, we evaluated models by using AP50 (AP at IoU =0.5), which is the standard PASCAL Visual Object Classes (VOC) metric (44).

Statistical analysis

Wilcoxon rank-sum tests were employed to compare the proposed YOLOv3-MobileNet and other models. A P value less than 0.05 is considered significant. Statistical analysis was performed using SPSS 22.0 (SPSS Inc., Chicago, IL, USA).

Results

The performance comparison between different models is shown in *Table 1*, where our YOLOv3-MobileNet is compared against Faster RCNN and original YOLOv3.

Table 1 suggests that the YOLOv3-MobileNet model has the best performance. In terms of AP50, the YOLOv3-MobileNet is higher than the Faster RCNN and the original YOLOv3. In terms of AP75 (AP at IoU =0.75), the YOLOv3-MobileNet is only 0.2% lower than the original YOLOv3. However, the YOLOv3-MobileNet model is several times faster than the other two. In conclusion, the YOLOv3-MobileNet model has high recognition accuracy and fast detection speed, which can meet the needs of medical report detection in the actual scene.

We also tested the performance of the DeepSSR. In terms of accuracy, the Tree-Edit-Distance-based Similarity (45) is used as the evaluation index, and the recognition accuracy is 91.10%. In terms of speed, the DeepSSR takes an average of 0.668 s to process an image. In addition, it can be deployed on NVIDIA Jetson AGX Xavier with a speed of 1.5 s.

To explore the impact of different table detection models on the performance of the frame, we conducted comparative experiments, and each experiment only changed the table detection model. The experimental results show that the accuracy and speed of the DeepSSR are improved by using the YOLOv3-MobileNet model. The system using YOLOv3-MobileNet has little difference in accuracy compared with the system using the other two models (*Table 2*).

DeepSSR shows excellent performance in the structured recognition of most UPBMR images (*Figure 7*). However, DeepSSR does not perform well for recognizing images with multiple lines of text in a cell (*Figure 8*).

Table 1 Comparison of table detection algorithms

Detection algorithm	AP50 (%)	AP75 (%)	Test time of a single image per second
Faster RCNN	96.5	94.1	0.030
YOLOv3	97.5	94.9	0.014
YOLOv3-MobileNet	97.8	94.7	0.006

AP50, average precision at Intersection-over-Union =0.5; AP75, average precision at Intersection-over-Union =0.75; Faster RCNN, Faster Region-based Convolutional Neural Network; YOLOv3, You Only Look Once, Version 3 (a real-time object detection algorithm); MobileNet, Efficient Convolutional Neural Network for Mobile Vision Application.

Table 2 Comparison results of experiments

Table detection model	Accuracy (%)	Test time of a single image per second
Faster RCNN	90.85	0.986
YOLOv3	89.51	0.670
YOLOv3-MobileNet	91.10	0.668

Faster RCNN, Faster Region-based Convolutional Neural Network; YOLOv3, You Only Look Once, Version 3 (a real-time object detection algorithm); MobileNet, Efficient Convolutional Neural Network for Mobile Vision Application.

Discussion

Different from the standard Natural Language Processing (NLP) tasks for text recognition from regular documents or paragraphs, it is not feasible to rely on Computer Vision (CV) and NLP alone to recognize text from unstructured documents. In the text recognition from unstructured documents, using image segmentation technology to extract text ignores semantic information, whereas the text in tables with complex structures cannot be effectively recognized with the standard NLP methods (46). In the present study, we successfully developed a deep learning system by integrating a series of methods (including image processing, table detection, text detection text recognition, and text structurization) to structurize text from UPBMR images. The system with high accuracy of 91.10% and a fast speed of 0.668 s per image exhibited remarkable performance in structurizing text from UPBMR images. These results indicate that this system could be used as a potential processing tool for automatically extracting text information. To our knowledge, this study was the first deep learning system that realizes the text structurization from Chinese UPBMR images.

The high accuracy of our system could be attributed to the appropriate processing methods adopted in each stage. These key processing methods include image correction based on the inclination angle in the image pre-processing stage, the proposed YOLOv3-MobileNet model in the table detection stage, the DB-CRNN model in the character recognition stage, and the text box assignment in the text structurization stage. (I) The results of image correction directly affect the accuracy of the table detection and the text structurization. On the one hand, corrected images ensure that the table area is approximately rectangular, promoting the increase in the accuracy of table detection and character recognition. On the other hand, the corrected image makes recognized text close to being arranged in rows and columns, facilitating the text box assignment according to its coordinates and thereby increasing the accuracy of the text structurization. (II) A lightweight deep learning model with YOLOv3 and MobileNet is designed for table detection from our UPBMR images. The model achieved an AP of 97.8% while ensuring a high processing speed of 0.006 s. The results of table detection. The accurate table detection helps text detection and recognition in the character recognition stage. (III) For text detection, the DB model trained on a public dataset with 1,670 images (17,548 annotated regions) at the 13th International Conference on Document Analysis and Recognition (ICDAR) is used in our deep learning system since this model is the best among 44 methods summited to the ICDAR 2015 Robust Reading Competition and its detection accuracy reached to 83.79%. For text recognition, the CRNN model is used in our deep learning system, since it is suitable for multi-lingual text recognition and its average accuracy reached 98.57% for Chinese text recognition (47). These excellent methods suitable for processing our UPBMR images are used in our system to improve the accuracy of character recognition. (IV) Given the prior knowledge (i.e., the structured text is arranged in rows and columns of the table) of the data

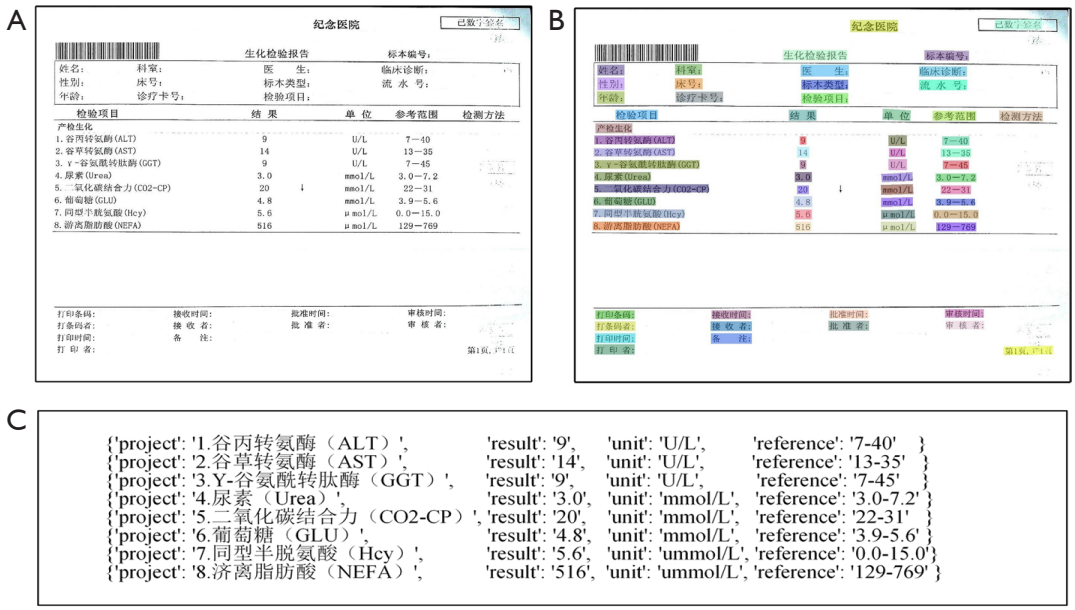


Figure 7 An example of a well-processed document. (A) The input image. (B) The character recognition result. (C) Structured data.

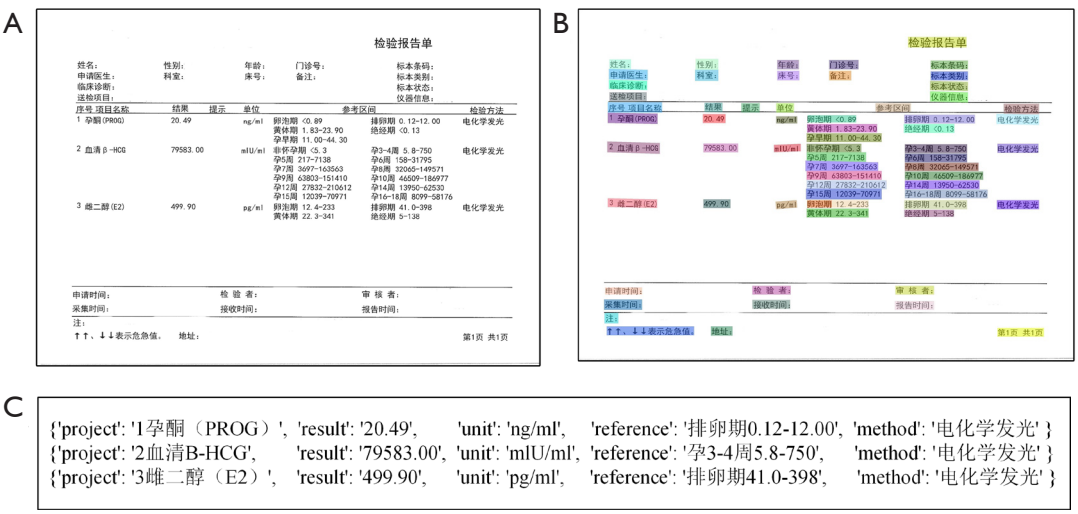


Figure 8 An example of a poorly processed document. (A) The input image. (B) The character recognition result. (C) Structured data.

structure in UPBMR images, the corresponding post-processing method is designed to ensure the high accuracy of the text structurization. Therefore, these methods designed for these UPBMR images allow our system to obtain good accuracy.

To enable our system to be used in future products, the use of various methods needs to increase the processing speed as much as possible without losing a lot of accuracy. In our deep learning system, the YOLOv3-MobileNet model for table detection and the DB-CRNN model for character recognition are mainly used for ensuring the accuracy of the system, while the lightweight network structure for table detection and the traditional methods for the pre-processing and post-processing of images are designed to reduce the calculation time. Especially, the number of processed images of the YOLOv3-MobileNet model in the

table detection stage is increased by 140% compared with that of the YOLOv3 model. These approaches used in our system make the processing of images at a speed of 0.668 s per frame. The system is also deployed on NVIDIA Jetson AGX Xavier that can be used in our future products and its speed reaches 1.5 s. These results indicate that the deep learning system may be able to quickly structurize text from UPBMR images in our future products.

For text recognition from UPBMR images, Xue *et al.* (48) introduced a dataset that contains 357 images with a resolution of 2,500×3,400 and proposed a method based on deep learning to extract text information from medical reports. This approach consists of two modules: the module of text detection is built based on Faster RCNN architecture, while the module of text recognition is constructed based on and improved through a concatenation structure. The concatenation structure can merge the features from both shallow and deep layers. Different from the approach of Xue *et al.*, our system includes four parts (i.e., image pre-preprocessing, table detection, character recognition, and text structurization) and thereby realize automatic text structurization. Moreover, DB and CRNN are used for text detection and text recognition, respectively. The performance of the DB model, which is better than Faster RCNN, is the second-best model for text detection (*Table 3*). Although the accuracy of CRNN is slightly lower than that of Xue *et al.* (90.6 *vs.* 95.8), its lower complexity compared to that of Xue *et al.* (34.0 *vs.* 42.9) can promote the speed of our system (*Table 4*). Therefore, our system tries its best to increase the speed of calculation under the accuracy of text recognition and automatically realizes the process from unstructured data to structured data.

White-Dzuro *et al.* (51) developed an optical mark recognition/optical character recognition system to extract medical information from paper assessment forms. The system scans a fixed area in the image by aligning the template to identify textual information, which only works for a single template. Different from the approach of White-Dzuro *et al.*, our system uses a deep learning approach to identify important information areas in the report. Therefore, our system is more robust and suitable for recognizing medical report images of various structures.

Despite good performance, the proposed approach still has pitfalls for future improvement: (I) due to the lack of annotated corpus for text detection and recognition, the trained models (i.e., DB and CRNN) on public datasets are used in our system. The annotated corpus of our UPBMR images will be completed in further work. In this case, the accuracy of our system will be further improved when these models are trained on various datasets including our annotated UPBMR images, while the speed of our system will be increased when lightweight structures for text detection and recognition are designed based on our annotated UPBMR images. (II) Due to the influence of text detection and recognition being greater on the system, the system using YOLOv3-MobileNet has little difference in accuracy compared with the system using the other two models. In the future, we will optimize text detection and text recognition models to make the difference between YOLOv3-MobileNet and other table detection models more obvious. (III) DeepSSR performs poorly for images with multiple lines of text in one cell. Although these images are less numerous in real life, their medical value is equally important. In the future, we will train a table structure recognition model to recognize cells in a table, and then merge multiple lines of text in the cells. Despite the limitation, the study may provide technical details for realizing text structurization from UPBMR images.

Conclusions

This paper presents a deep learning system for structurizing text from UPBMR images. The system has the advantages of automation and being lightweight. First, it can realize automatic input. People can directly obtain the structured form information by entering the medical report image. Second, when it is deployed to the actual application scenario, the time to identify and construct medical reports is less than 2 s, which realizes lightweight on the premise of ensuring accuracy. Therefore, our deep learning system can automatically extract the form information in the medical report, solve the problem of cumbersome and time-consuming manual input operation in the actual scene, and realize the machine-aided recognition function of paper report content.

Table 3 Text recognition results on the ICDAR 2015 dataset. These methods with "+" are collected from [Lu *et al.* 2021, (49)]. These methods with "–" are collected from [Liao *et al.* 2019, (23)]. These methods with "#" are collected from [Zhang *et al.* 2021, (50)]. Poly-FRCNN-3 is similar to that of [Xue *et al.* 2019, (48)]

Methods	Precision	Recall	F1 score	Note
Seglink + VGG16	73.10	76.80	75.00	+
WordSup	77.03	79.33	78.16	+
EAST + VGG16	80.05	72.80	76.40	+
EAST + ResNet50	77.32	81.66	79.43	+
EAST + PAVNET2x	83.60	73.50	78.20	+
EAST + PAVNET2x MS	84.64	77.23	80.77	+
STN-OCR (Saif *et al.* 2020)	78.53	65.20	71.86	+
Poly-FRCNN-3 (Ch'ng *et al.* 2020)	80.00	66.00	73.00	+
RFRN-4s (Deng *et al.* 2021)	85.10	76.80	80.80	+
EAST (Lu *et al.* 2021)	85.59	76.94	81.03	+
CTPN (Tian *et al.* 2016)	74.20	51.60	60.90	–
EAST (Zhou *et al.* 2017)	83.60	73.50	78.20	–
SSTD (He *et al.* 2017)	80.20	73.90	76.90	–
WordSup (Hu *et al.* 2017)	79.30	77.00	78.20	–
Corner (Lyu *et al.* 2018)	94.10	70.70	80.70	–
TB (Liao, Shi, and Bai 2018)	87.20	76.70	81.70	–
RRD (Liao *et al.* 2018)	85.60	79.00	82.20	–
MCN (Liu *et al.* 2018)	72.00	80.00	76.00	–
TextSnake (Long *et al.* 2018)	84.90	80.40	82.60	–
PSENet (Wang *et al.* 2019)	86.90	84.50	85.70	–
SPCNet (Xie *et al.* 2019)	88.70	85.80	87.20	–
LOMO (Zhang *et al.* 2019)	91.30	83.50	87.20	–
ATRR (Wang *et al.* 2019)	89.20	86.00	87.60	#
CRAFT (Baek *et al.* 2019)	89.80	84.30	86.90	–
PAN (Wang *et al.* 2019)	84.00	81.90	82.90	#
ContourNet (Wang *et al.* 2019)	87.60	86.10	86.90	#
SAE (720) (Tian *et al.* 2019)	85.10	84.50	84.80	–
GCN (Zhang *et al.* 2020)	88.50	84.70	86.60	#
Texts as Lines (Wu *et al.* 2020)	81.70	77.10	79.40	#
WSSTD (Zhang *et al.* 2021)	83.10	85.70	84.40	#
SAE (990) (Tian *et al.* 2019)	88.30	85.00	86.60	–
Ours (DB)	91.80	83.20	87.30	–

ICDAR, International Conference on Document Analysis and Recognition; Seglink, Segment Linking; VGG16, Visual Geometry Group Network; WordSup, Exploiting Word Annotations for Character based Text Detection; EAST, Efficient and Accurate Scene Text Detector; ResNet50, Residual Neural Network; PAVNET, Deep but Lightweight Neural Network; STN-OCR, Spatial Transformer Network; Poly-FRCNN, Polygon-Faster-Region-based Convolutional Neural Network; RFRN, Recurrent Feature Refinement Network; CTPN, Connectionist Text Proposal Network; SSTD, Single Shot Text Detector; Corner, scene text detector that localizes text by corner point detection and position-sensitive segmentation; TB, TextBoxes++; RRD, Rotation-sensitive Regression Detector; MCN, Markov Clustering Network; TextSnake, A Flexible Representation for Detecting Text of Arbitrary Shapes; PSENet, Progressive Scale Expansion Network; SPCNet, Scale Position Correlation Network; LOMO, Look More Than Once; ATRR, Arbitrary Shape Scene Text Detection with Adaptive Text Region Representation; CRAFT, Character Region Awareness for Text Detection; PAN, Pixel Aggregation Network; SAE, Shape-Aware Embedding; GCN, Graph Convolutional Network; WSSTD, Weakly Supervised Scene Text Detection; DB, differentiable binarization network.

Table 4 Text recognition results in a dataset that contains 357 images of medical laboratory reports. The methods are collected from [Xue *et al.* 2019, (48)]

Methods	Accuracy (%)	mED	Size (MB)
Attention OCR (Brzeski *et al.* 2019)	83.8	2.51	221.5
Xue *et al.*	95.8	3.29	42.9
Ours (CRNN)	90.6	3.79	34.0

mED, mean Edit Distance; CRNN, Convolutional Recurrent Neural Network; MB, mega byte; OCR, optical character recognition.

Acknowledgments

Funding: This research was funded by the National Key Research and Development Project [grant numbers 2019YFC0120102 (HW, JB, and YL) and 2019YFC0121907 (JB and YL)], the National Natural Science Foundation of China [grant number 61901192 (JB)], and Guangdong Provincial Key Laboratory of Traditional Chinese Medicine Informatization [2021B1212040007 (JB and YL)].

Footnote

Reporting Checklist: The authors have completed the TRIPOD reporting checklist. Available at https://atm.amegroups.com/article/view/10.21037/atm-21-6672/rc

Data Sharing Statement: Available at https://atm.amegroups.com/article/view/10.21037/atm-21-6672/dss

Peer Review File: Available at https://atm.amegroups.com/article/view/10.21037/atm-21-6672/prf

Conflicts of Interest: All authors have completed the ICMJE uniform disclosure form (available at https://atm.amegroups.com/article/view/10.21037/atm-21-6672/coif). The authors have no conflicts of interest to declare.

Ethical Statement: The authors are accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

References

1. Lopresti D, Nagy G. Automated table processing: An (opinionated) survey. Proceedings of the Third IAPR Workshop on Graphics Recognition 1999;109-134.
2. Shu T, Liu H, Goss FR, et al. EHR adoption across China's tertiary hospitals: a cross-sectional observational study. Int J Med Inform 2014;83:113-21.
3. Kieninger T, Dengel A. The T-Recs Table Recognition and Analysis System. In: Lee SW, Nakano Y. editors. Document Analysis Systems: Theory and Practice. DAS 1998. Lecture Notes in Computer Science, vol 1655. Berlin, Heidelberg: Springer, 1999:255-270.
4. Yildiz B, Kaiser K, Miksch S. pdf2table: A method to extract table information from pdf files. Proceedings of the 2nd Indian International Conference on Artificial Intelligence, 2005:1773-85.
5. Koci E, Thiele M, Romero O, et al. A genetic-based search for adaptive table recognition in spreadsheets. 2019 International Conference on Document Analysis and Recognition (ICDAR), 2019:1274-9.
6. Gilani A, Qasim SR, Malik I, et al. Table detection using deep learning. 2017 14th IAPR international conference on document analysis and recognition (ICDAR), 2017;1:771-6.
7. Siddiqui SA, Malik MI, Agne S, et al. DeCNT: Deep Deformable CNN for Table Detection. IEEE Access 2018;6:74151-61.
8. Ren S, He K, Girshick R, et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Trans Pattern Anal Mach Intell 2017;39:1137-49.
9. Lin TY, Dollár P, Girshick R, et al. Feature pyramid networks for object detection. 2017 IEEE Conference

on Computer Vision and Pattern Recognition (CVPR); Honolulu, HI, USA. 2017:936-944.
10. Sun N, Zhu Y, Hu X. Faster R-CNN based table detection combining corner locating. 2019 International Conference on Document Analysis and Recognition (ICDAR); Sydney, NSW, Australia. IEEE, 2019:1314-9.
11. Huang Y, Yan Q, Li Y, et al. A YOLO-based table detection method. 2019 International Conference on Document Analysis and Recognition (ICDAR) 2019:813-8.
12. Redmon J, Farhadi A. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767, 2018.
13. Zanibbi R, Blostein D, Cordy JR, et al. A survey of table recognition. Document Analysis and Recognition 2004;7:1-16.
14. Mayan JA, Deep KA, Kumar M, et al. Number plate recognition using template comparison for various fonts in MATLAB. 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), 2016:1-6.
15. Epshtein B, Ofek E, Wexler Y. Detecting text in natural scenes with stroke width transform. 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2010:2963-70.
16. Yin XC, Yin X, Huang K, et al. Robust Text Detection in Natural Scene Images. IEEE Trans Pattern Anal Mach Intell 2014;36:970-83.
17. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9:1735-80.
18. Tian Z, Huang W, He T, et al. Detecting text in natural image with connectionist text proposal network. European Conference on Computer Vision, 2016:56-72.
19. Zhou X, Yao C, Wen H, et al. East: an efficient and accurate scene text detector. Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2017:5551-60.
20. Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015:3431-40.
21. Kong S, Fowlkes CC. Recurrent pixel embedding for instance grouping. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018:9018-28.
22. Wang W, Xie E, Li X, et al. Shape robust text detection with progressive scale expansion network. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019:9336-45.
23. Liao M, Wan Z, Yao C, et al. Real-time scene text detection with differentiable binarization. Proceedings of the AAAI Conference on Artificial Intelligence, 2020:11474-81.
24. Sheshadri K, Divvala SK. Exemplar Driven Character Recognition in the Wild. Proceedings of the British Machine Vision Conference. BMVA Press, 2012:13.1-13.10.
25. Coates A, Carpenter B, Case C, et al. Text detection and character recognition in scene images with unsupervised feature learning. 2011 International Conference on Document Analysis and Recognition. IEEE, 2011:440-5.
26. Mishra A, Alahari K, Jawahar C. Top-down and bottom-up cues for scene text recognition. 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012:2687-94.
27. Wang T, Wu DJ, Coates A, et al. End-to-end text recognition with convolutional neural networks. Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), 2012:3304-8.
28. Shi B, Bai X, Yao C. 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 2017;39:2298-304.
29. Graves A, Fernández S, Gomez F, et al. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. Proceedings of the 23rd International Conference on Machine Learning, 2006:369-76.
30. Cheng Z, Bai F, Xu Y, et al. Focusing attention: Towards accurate text recognition in natural images. Proceedings of the IEEE International Conference on Computer Vision, 2017:5076-84.
31. Liu X, Liang D, Yan S, et al. Fots: Fast oriented text spotting with a unified network. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018:5676-85.
32. Saravanan C. Color image to grayscale image conversion. 2010 Second International Conference on Computer Engineering and Applications, 2010:196-9.
33. Kumar T, Verma K. A Theory Based on Conversion of RGB image to Gray image. International Journal of Computer Applications 2010;7:7-10.
34. Trier OD, Taxt T. Evaluation of binarization methods for document images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1995;17:312-5.
35. Otsu N. A threshold selection method from gray-scale histograms. IEEE Transactions on Systems, Man, and Cybernetics 1979;9:62-6.

36. Ching YT. Detecting line segments in an image–a new implementation for Hough transform. Pattern Recognition Letters 2001;22:421-9.
37. Ishitani Y. Document skew detection based on local region complexity. Proceedings of 2nd International Conference on Document Analysis and Recognition (ICDAR'93), 1993:49-52.
38. Cao Y, Wang S, Li H. Skew detection and correction in document images based on straight-line fitting. Pattern Recognition Letters 2003;24:1871-9.
39. Singh R, Kaur R. Improved skew detection and correction approach using Discrete Fourier algorithm. International Journal of soft computing and Engineering 2013;3:5-7.
40. Howard AG, Zhu M, Chen B, et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. arXiv preprint arXiv:1704.04861, 2017.
41. He K, Zhang X, Ren S, et al. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. IEEE Trans Pattern Anal Mach Intell 2015;37:1904-16.
42. Feng Y, Li Y. An overview of deep learning optimization methods and learning rate attenuation methods. Hans Journal of Data Mining 2018;8:186-200.
43. Du Y, Li C, Guo R, et al. PP-OCR: A practical ultra lightweight OCR system. arXiv preprint arXiv:2009.09941, 2020.
44. Padilla R, Netto SL, da Silva EA. A survey on performance metrics for object-detection algorithms. 2020 International Conference on Systems, Signals and Image Processing (IWSSIP). IEEE, 2020:237-42.
45. Zhong X, ShafieiBavani E, Jimeno Yepes A. Image-based table recognition: data, model, and evaluation. European Conference on Computer Vision, 2020:564-80.
46. Baviskar D, Ahirrao S, Kotecha KJIA. Multi-layout Unstructured Invoice Documents Dataset: A dataset for Template-free Invoice Processing and its Evaluation using AI Approaches. IEEE Access 2021;9:101494-512.
47. Wang J, Wu R, Zhang S, et al. Robust Recognition of Chinese Text from Cellphone-acquired Low-quality Identity Card Images Using Convolutional Recurrent Neural Network. Sensors and Materials 2021;33:1187-98.
48. Xue W, Li Q, Xue Q. Text detection and recognition for images of medical laboratory reports with a deep learning approach. IEEE Access 2019;8:407-16.
49. Lu M, Mou Y, Chen CL, et al. An Efficient Text Detection Model for Street Signs. Appl Sci 2021;11:5962.
50. Zhang W, Qiu Y, Liao M, et al. Scene Text Detection with Scribble Line. International Conference on Document Analysis and Recognition, 2021:79-94.
51. White-Dzuro CG, Schultz JD, Ye C, et al. Extracting Medical Information from Paper COVID-19 Assessment Forms. Appl Clin Inform 2021;12:170-8.

Cite this article as: Liu H, Wang H, Bai J, Lu Y, Long S. DeepSSR: a deep learning system for structured recognition of text images from unstructured paper-based medical reports. Ann Transl Med 2022;10(13):740. doi: 10.21037/atm-21-6672

