Electronic Engineering and Informatics
G. Izat Rashed (Ed.)

doi:10.3233/ATDE240123

The Text Detection and Recognition Method for Electrical Nameplates Based on Deep Learning

Yapeng WANG[1], Yanan DU, Zhengning PANG, Jingxian QI and Hongsong GE
Nanjing Nari Information & Communication Technology Co., Ltd., Nanjing, Jiangsu, 210003, China

Abstract. To achieve automatic extraction of text information from electrical equipment nameplates and improve equipment management efficiency, a text content extraction method for nameplate scenes is proposed. OCR, as a part of artificial intelligence, combined with deep learning, can achieve higher accuracy in nameplate recognition and a wider range of applications. There are two key issues worth exploring in the OCR recognition process: text region detection and text region recognition. The OCR recognition in this paper adopts the text detection and text recognition algorithm model based on Meaningful learning. Specifically, using the CTPN algorithm for text region detection and the CRNN algorithm for text recognition.

Keywords. OCR, image processing, deep learning, text region detection, text recognition

1 Introduction

With the increasing scale of China's power grid, the types and quantities of power equipment are becoming increasingly large. The equipment nameplate records the basic parameters and identity information of the equipment. When conducting statistics, management, and inspection of power equipment, it is often necessary to obtain the equipment nameplate information in order to collect the equipment nameplate image. So, using image processing technology to automatically recognize[1] the text information in the nameplate can improve the management of the entire power system equipment level is of great significance.

The nameplate information of power equipment contains a large number of characters and a variety of types, mainly Chinese characters, letters, numbers, and some special symbols. Moreover, the captured nameplate images often have complex backgrounds and are difficult to recognize. Given that deep learning methods have achieved very good results in the field of image recognition, this article proposes a deep learning[2] based electrical nameplate text detection and recognition method. Compared with the recognition technology based on traditional image processing, this method uses the CTPN algorithm for text detection in the scene area, and uses the convolutional Recurrent neural network[3] model CRNN for text recognition[4]. This article verifies the

[1] Corresponding author: Yapeng WANG, Nanjing Nari Information & Communication Technology Co., Ltd., e-mail: wangyapeng@sgepri.sgcc.com.cn

advantages of this method through experiments on real data on power equipment nameplates.

2 The OCR Technology and Related Research

2.1 OCR change process

The traditional concept of OCR[5] is more limited to document recognition, and later the concept of OCR has been gradually expanded to general text image recognition, mainly natural scene text image recognition. The information on natural scene images is more abundant. In view of its great diversity and complexity, the difficulty of text detection and recognition in natural scene images is greater than that in scanned images. In some cases where the background is slightly complex or there are variations in text, traditional methods will generally fail, and the generalization of the model is weak. The layout analysis (connected domain method) and projection transformation (line segmentation) methods can only limit the processing of relatively simple scenes. Once the scene becomes a natural scene, it will basically fail.

2.2 Traditional OCR method

Although traditional OCR methods have evolved many methods with the development of image processing and pattern recognition. For example, based on traditional manual design features (Handcraft Features), including connected domain based methods, projection analysis, and HOG based detection box descriptions. But usually, it can be divided into the following steps: the first step is to preprocess, denoise, minimize, enhance, and so on the image; the second step is layout analysis, where text can be selected from the image, followed by the position of the text line; the third step is character segmentation, which divides the candidate regions of the text into individual characters; the fourth step is character recognition. The classifier accepts the extracted feature input, then classifies it, and finally outputs the text information corresponding to the feature. The fifth step is post-processing to simplify the classification results.

2.3 OCR method based on deep learning

The text recognition scenario based on deep learning[6] divides some complex processes into two main steps, one is text detection (mainly used to locate the position of the text), and the other is text recognition (mainly used to identify the specific content of the text). The former is a problem of target detection. For specific scene text recognition tasks, such as equipment nameplates, this paper will use the scene text detection model to divide text segmentation into a series of fixed size windows, regression each small window, and detect text sequences of variable length, so as to improve the accuracy and robustness of detection. The latter is the problem of text recognition. This paper adopts the recognition method of the whole text sequence based on the convolutional recurrent neural network model, and combines the convolutional layer, the circular layer and the transcription layer to extract the characters or words as a whole.

3 Text Region Detection is Implemented Based on the CTPN Algorithm

3.1 Overview of the CTPN algorithm

Although text detection is actually a branch of target detection, there are certain differences between the two before. Their differences are as follows: first, the text is generally from left to right, and the width between words is roughly the same; second, the fixed width is one word at a time, so it is enough to detect the height of the text, and the text content can be variable length, forming a variable length sequence; third, the essence of text detection is still the RPN method. The CTPN algorithm is not based on whole text detection, because the length of the text is not fixed, but it can detect each small block, detect the content of each small block, and then stitch them together to form a text. Among them, the width of each small block detected is fixed, and the height is different.

3.2 The CTPN network architecture

VGG performs feature extraction, generates a candidate frame, judges the candidate frame, and judges whether the current candidate frame is text, which can be seen as a binary classification task, whether it is text or not. First, as shown in the ‘figure 2’, VGG16 is used as the basic network for feature extraction, and the features of conv5 are obtained as a feature map with a size of N*W*H*C; then a sliding window of three times three size is selected to perform intensive sliding on the feature map. Each window can get a feature vector with a size of three times three times c. At this time, a feature map of W*H*9C*N is output. K region proposals can be produced by k anchor points of each sliding window at the same time, and then Reshape the obtained feature map, and the size obtained after matrix variable dimension is W*9C*(NH). Then according to the data flow of Batch=NH&TIME_MAX=W, it is input to the bidirectional LSTM (Long-Short Term Memory), and then the BLSTM is output, and the shape is restored after a reshape, W*(NH)*256 becomes W*256*H*N. The features obtained at this time include not only spatial features but also sequence features learned by the long short-term memory network. Then after a 512 fully connected (FC) convolutional layer, it becomes W*512*H*N, and finally through the region proposal network (RPN), the text proposals are obtained, as shown in the ‘figure 1’. These text proposals will be merged into the final text box through a text line construction algorithm:

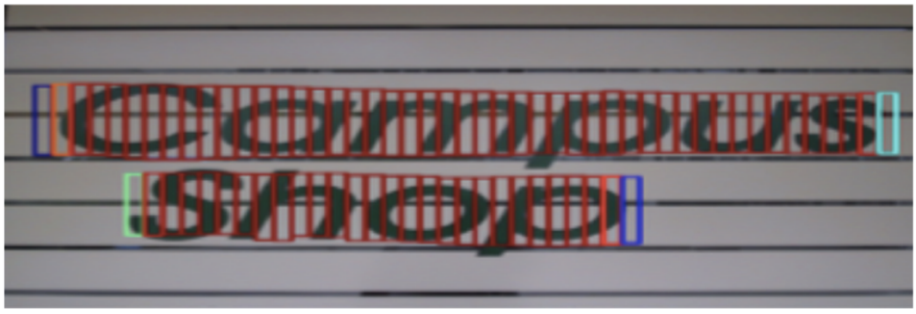

Figure 1. Text proposals

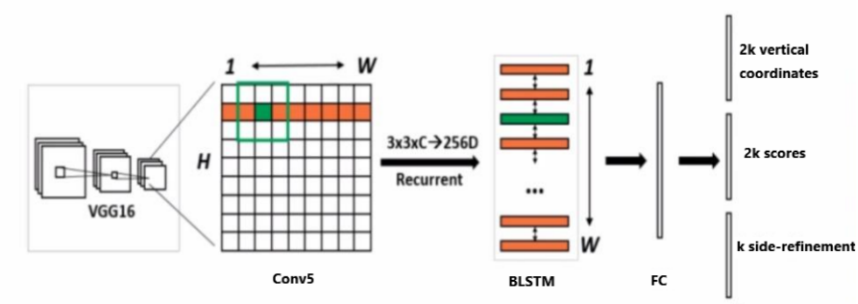

Figure 2. The CTPN network architecture

3.3 BLSTM (Bi-direction Long-Short Term Memory) Integrate into the contextual information

Due to the continuity of text, text detection requires not only the abstract spatial features of convolutional neural networks, but also sequential features. Among them, the convolutional neural network learns the spatial information in the receptive field, while the long-short-term memory network learns sequence features, emphasizing timing. Compared with LSTM[6], the advantage of BLSTM[7] is: BLSTM connects two LSTMs in opposite directions, that is, splicing the vectors from front to back and from back to front, which can make more features, as shown in the 'figure 3'.

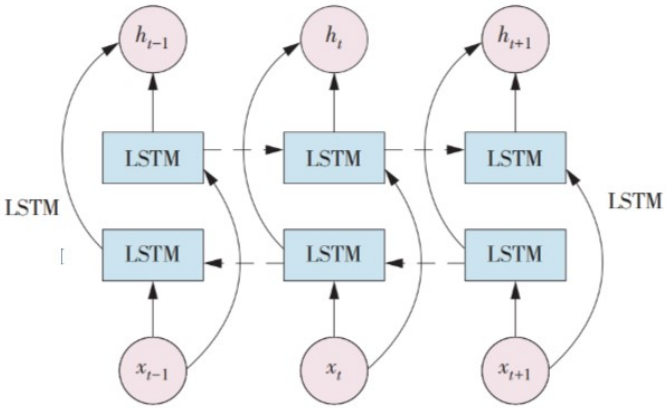

Figure 3. BLSTM

The advantage of adding BLSTM to CTPN is that if the situation is shown in the figure below, there is only a part of parentheses in Box 8. If it is difficult to judge, the content of Box 8 can be inferred according to the temporal context.

4 Text Content Recognition is Realized Based on the CRNN Algorithm

4.1 CRNN frame

CRNN (Convolutional Recurrent Neural Network) is the combination of CNN and RNN, that is, the convolutional cyclic neural network architecture, which is mostly employed

for image-based sequence recognition problems. It recognizes end-to-end text sequences with variable lengths. It is a text recognition The task becomes a timing-dependent sequence learning problem. Generally, the problem of character recognition is regarded as the problem of predicting the sequence, so the recurrent neural network is utilized to predict the sequence. To put it simply, the OCR text recognition process is to extract the image features through the convolutional neural network (CNN), select the recurrent neural network (RNN) to predict the sequence, and then obtain the final result through the CTC decoding mechanism.

The CRNN network consists of 3 parts. As shown in the 'figure 4', It is assumed that the input image is a three-channel color map with a width and height of 32 and 100, respectively. The first part is the convolutional layer (Convolutional Layers), which is utilized to extract the convolutional feature map of the image. The size of the above image becomes (1, 25, 512) convolutional feature matrix. The second part is the recurrent network layer (Recurrent Layers), here it is actually a deep BLSTM network, and the text sequence features continue to be extracted on the basis of the Convolutional feature. The third part is the transcription layer (Transcription Layers), which performs a surtax on the RNN output and output characters.

Assuming that there is an image now, the following processing needs to be done before inputting features into Recurrent Layers. First, the image is reduced to a height of 32, the width W can be arbitrary, the number of channels is 1, and then the height is changed through a convolutional neural network. It is 1, the width is W/4, the number of channels is 512, and then the time sequence number T=W/4 of the long and short time memory is set, and the dimension D=512 of a time sequence number input, that is, the length of each vector is 512. After the above series of process operations are expected to be completed, the features can be input into the LSTM. The 256 hidden nodes are the property of the long-term short-term memory. After the long-term short-term memory, they become T vectors with a length of class, and then let Softmax process them. The prediction probability of corresponding characters is represented by each element of the column vector (that is, the class length), and finally the redundant prediction results of the current time series are removed and combined into a complete recognition result.

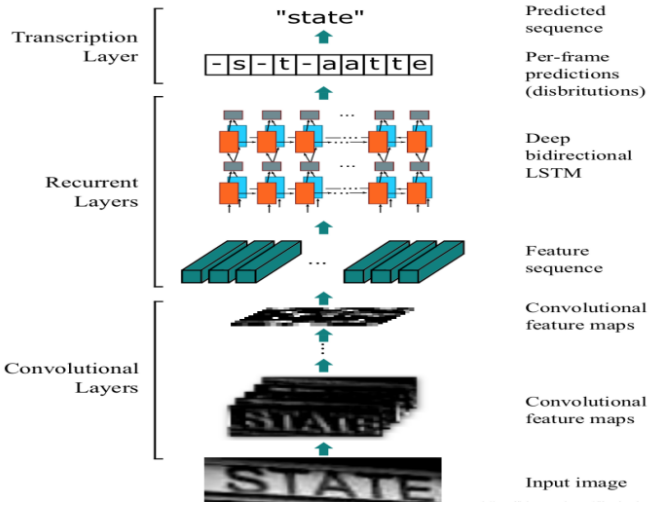

Figure 4. Flow chart of CRNN

4.2 CTC (Connectionist Temporal Classifier)

There are currently two widely used decoding mechanisms. One is the Seq2Seq mechanism, and the other is the CTC[8] decoding mechanism.

The Seq2Seq model represented by encoding and decoding uses RNN or CNN to decode after the input image is encoded by convolution. Encoding is to convert an input sequence into a vector of constant length, and decoding is to decode an input vector of constant length into an output sequence[9]. The Seq2Seq model is commonly used in regression prediction. But usually, in order to improve the correctness of text recognition, the Attention mechanism will be added. Since this study did not use this mechanism, it will not be set out in detail.

Connection time classifier, referred to as CTC, CTC module is utilized to align the input and output results. Due to problems such as different intervals between characters, the same character has unique forms of expression, but in fact, it is the same character. Just like words, some words have a long interval and the speaking speed is slow, while some speak fast, and the words are closely connected. During recognition, instead of recognizing a whole picture, the input image is divided into blocks before recognition, and the probability that each block belongs to a specific character is obtained (unrecognizable marks are generally represented by"-"). Due to the character interval or deformation and other factors, when the input image is recognized in blocks, the blocks next to each other may be recognized as the same result, resulting in repeated characters. Therefore, the CTC mechanism is utilized to solve the input and output alignment problem. After the model is trained, the space characters and repeated characters in the result are removed, as indicated in the 'figure 5' below.

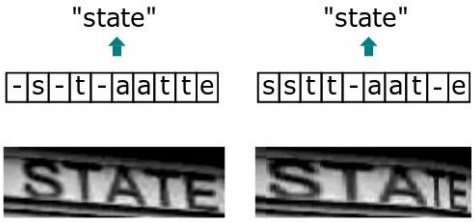

Figure 5. The CTC decoding mechanism

In the mathematical model, the CTC layer can search for the label sequence output with the highest probability distribution based on the predicted information of each input frame. The Loss function of CTC is defined as:

$$L_S = -\ln \prod_{(x,z)\in S} P(z \mid x) = -\sum_{(x,z)\in S} \ln P(z \mid x) \tag{1}$$

5 System Experiments and Conclusions

5.1 Analysis of target detection results

5.1.1 Experimental environment

The dataset used in this experiment is a real electronic nameplate dataset containing 800

images. Some samples are shown in the following 'figure 6'. The experiment is based on the object detection model to first segment the text area in the image. The dataset is divided into training, validation, and testing sets in a ratio of 8:1:1. The operating system is Windows 10, the training framework is Python, and the software programming environment is Python 3.7.

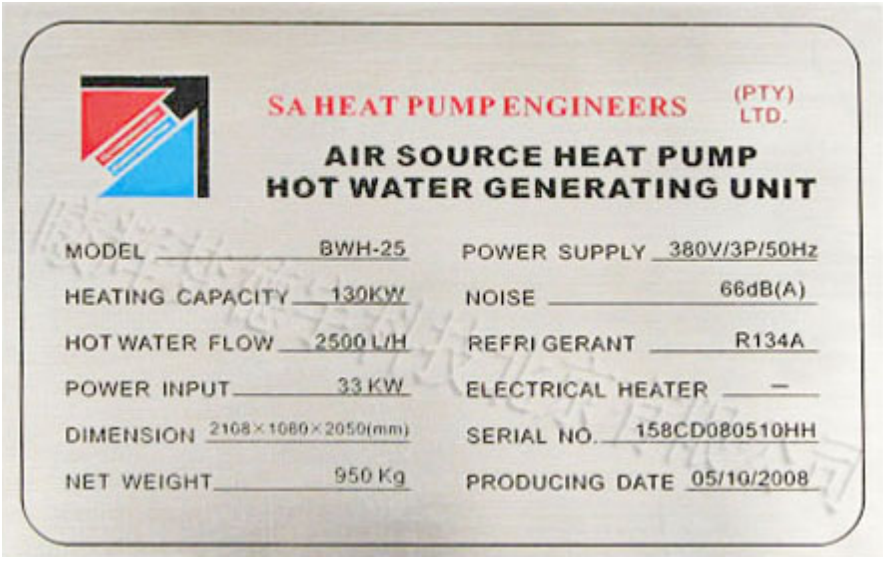

Figure 6. Sample images in experimental data

5.1.2 Evaluation method

We set TP as a positive sample with correct classification, FP as a negative sample with classification errors, FN as a positive sample with classification errors, and P (r) as the expression of the PR curve function. The method for calculating the precision and recall of test samples is as follows.

$$P = \frac{TP}{TP + FP} \tag{2}$$

$$R = \frac{TP}{TP + FN} \tag{3}$$

$$F_{measure} = \frac{1}{\frac{\alpha}{P} + \frac{1-\alpha}{R}} \tag{4}$$

where accuracy P represents the ratio of correctly recognized text to the number of all recognized text, recall rate R represents the identified quality, $F_{measure}$ represents the overall evaluation indicator, and α is the relative weight of the Harmonic mean, which is generally set to 0.5

5.1.3 Result analysis

As shown in Table 1, this model compares the CTPN algorithm with classic EAST and RRPN algorithms on the dataset and finds that the accuracy of this algorithm is improved by 5% to 10% compared to the other two classic algorithms; In terms of recall rate, this algorithm performs the best, with a 2% to 6% improvement compared to other algorithms; In terms of overall evaluation indicators, the algorithm in this article achieved an improvement of 2% to 4%.

Table 1. Comparison of experimental indicators in the dataset

Test model	Precision	Recall	F-measure
EAST	0.80	0.80	0.64
RRPN	0.76	0.75	0.65
CTPN	0.85	0.82	0.67

5.2 Analysis of text recognition results

5.2.1 Experimental environment

The ICDAR2017 dataset was selected for experimental testing, which is a classic dataset in text recognition. The text in the image is usually horizontal and has a simple background. The operating system is Windows 10, the training framework is Python, and the CRNN model training runs on the Pytorch framework.

5.2.2 Evaluation method

We set F as the text recognition rate, P as the total number of correctly identified samples, and T as the overall sample size. The calculation method for text recognition accuracy of test samples is as follows:

$$F = \frac{P}{T} \times 100\% . \tag{5}$$

5.2.3 Result analysis

The experimental results of replacing the CRNN network with different backbone networks are shown in Table 2.

Table 2. CRNN performance comparison

model	F/%	T/s
VGG16	91.3	0.054
Resnet	93.2	0.058
DenseNet	98.2	0.060

Replacing different backbones in CRNN has different effects on improving network performance. The text recognition rate of CRNN-Rsenet has increased by 1.9% compared to the original CRNN, and the text recognition rate of CRNN-Densinet has increased by 7.4%.

The partial experimental results output after fusing the recognition results and inputs are shown in the following ‘figure 7’:

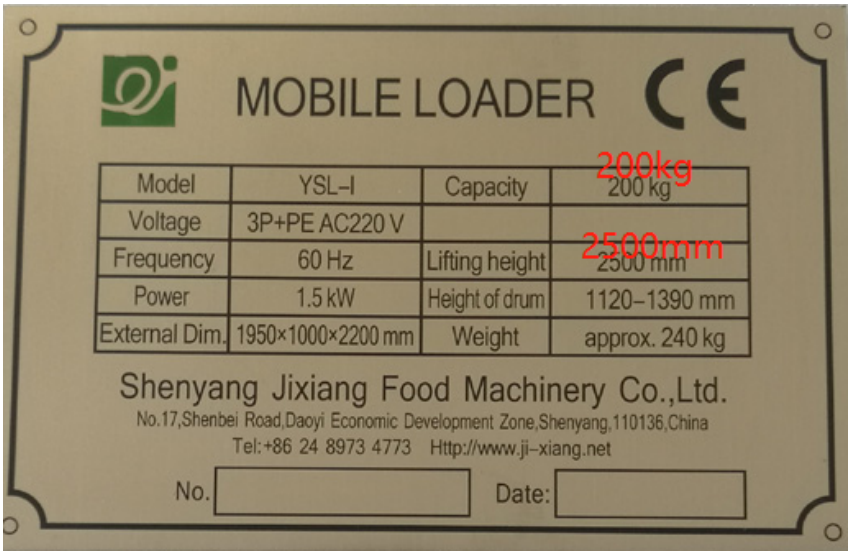

Figure 7. Examples of identification results in experimental data

6 Tag

In recent years, artificial intelligence has become very popular. With the development of deep learning, OCR recognition technology has also rapidly developed as a small branch. The application of deep learning in text recognition and detection has achieved good results. This article analyzes and compares traditional and meaningful learning based object detection algorithms, and proposes a text detection and recognition method in the field of electrical nameplate recognition.

The CTPN algorithm is used for text region detection, which combines small window sliding detection and BLSTM to accurately label text boxes on images. Then, sequences with higher text box scores are connected as text regions and output to the next recognition step. This detection model performs well in detecting horizontal and horizontal images, with good accuracy and high robustness. The CRNN algorithm is used for text recognition, which does not require image segmentation and recognition. It can recognize any length of text without the need for a unified size input, and has been proven to achieve good performance through experimental data comparison. In future research work, we will attempt to improve the detection network architecture to solve the problem of multi angle text detection and further improve the effectiveness of text detection, and consider the changes brought about by the introduction of attention mechanisms[10] in recognition work.

References

[1] Zhaoye Zhou, Gengyan Zhao, Richard Kijowski, etc. Deep Convo-lutional Neural Network for Segmentation of Knee Joint Anatomy[J].Magnetic Resonance in Medicine,2018(6):2759-2770.

[2] Li Z, Li Y, Xiong W, et al. Research on Voiceprint Recognition Technology Based on Deep Neural Network [C]// BIC 2021: 2021 International Conference on Bioinformatics and Intelligent Computing. 2021.

[3] MOHAMED Y, KHALED F H, USAMA S M. Accurate, data-efficient, unconstrained text recognition with convolutional neural networks [J]. Pattern recognition, 2020, 108: 1-13.
[4] Janarthanan A, Pandiyarajan C, Sabarinathan M, et al. Research on Deep Learning Techniques in Breaking Text-Based Captchas and Designing Image-Based Captcha [J]. 2021.
[5] Kim G, Hong T, Yim M, et al. Donut: Document Understanding Transformer without OCR [J]. 2021.
[6] ZHANG J, JIANG F. Multi-level supervised network for person re-identification [C]. Proceedings of the 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Piscataway: IEEE, 2019: 2072-2076.
[7] WANG Lihui, QIN Chengshuai, YANG Xianbiac, et al. Image detection on welding area of cooling water pipe in power station based on deep learning [J]. Electric Power Engineering Technology, 2020, 39(5): 191-196.
[8] Jaderberg M, Simonyan K, Vedaldi A, et al. Reading Text in the Wild with Convolutional Neural Networks [J]. International Journal of Computer Vision, 2016, 116(01): 1-20.
[9] BELHARBI S, HÉRAULT R, CHATELAIN C, et al. Deep neural networks regularization for structured output prediction [J]. Neurocomputing, 2018, 281: 169-177.
[10] Cheng Z, Bai F, Xu Y, et al. Focusing attention: Towards accurate text recognition in natural images [C]. Proceedings of the IEEE international conference on computer vision. 2017: 5076-508.