Hindawi
Scientific Programming
Volume 2021, Article ID 6406856, 11 pages
https://doi.org/10.1155/2021/6406856

Research Article

Research on Methods of English Text Detection and Recognition Based on Neural Network Detection Model

Chunlan Li

Geely University of China, Chengdu, Sichuan 610095, China

Correspondence should be addressed to Chunlan Li; lichunlan@bgu.edu.cn

Received 25 October 2021; Revised 16 November 2021; Accepted 19 November 2021; Published 13 December 2021

Academic Editor: Bai Yuan Ding

With the rapid development of computer science, a large number of images and an explosive amount of information make it difficult to filter and effectively extract information. This article focuses on the inability of effective detection and recognition of English text content to conduct research, which is useful for improving the application of intelligent analysis significance. This paper studies how to improve the neural network model to improve the efficiency of image text detection and recognition under complex background. The main research work is as follows: (1) An improved CTPN multidirectional text detection algorithm is proposed, and the algorithm is applied to the multidirectional text detection and recognition system. It uses the multiangle rotation of the image to be detected, then fuses the candidate text boxes detected by the CTPN network, and uses the fusion strategy to find the best area of the text. This algorithm solves the problem that the CTPN network can only detect the text in the approximate horizontal direction. (2) An improved CRNN text recognition algorithm is proposed. The algorithm is based on CRNN and combines traditional text features and depth features at the same time, making it possible to recognize occluded text. The algorithm was tested on the IC13 and SVT data sets. Compared with the CRNN algorithm, the recognition accuracy has been improved, and the detection and recognition accuracy has increased by 0.065. This paper verifies the effectiveness of the improved algorithm model on multiple data sets, which can effectively detect various English texts, and greatly improves the detection and recognition performance of the original algorithm.

1. Introduction

With the rise and popularity of the Internet of Things, there will be a huge amount of data every day, these data with the development and change of society. However, with the rapid growth of data volume, there will be a large number of data that is difficult to understand and difficult to manage, including pictures and video image data accounting for a considerable proportion. The image not only contains the shape, color, and other underlying information, but also contains the text and other high-level semantic information, which plays an indispensable role in the analysis and utilization of the image. In order to solve the problem of multidirectional text detection, the SegLink [1] algorithm cuts the text into smaller text blocks that are easier to detect and then connects the small text blocks into complete text areas. TextBoxes [2] algorithm, with SSD as the basic framework, adjusts the text area candidate box's length and width ratio and convolution core into rectangles, proposing an end-to-end text detector, so that it is more suitable for detecting slender lines of text. Mask TextSpotter [3–5] algorithm, in order to solve the problem of text that can detect any shape, combines FPN network, Fast RCNN network, and RPN network and introduces the idea of segmentation to propose an end-to-end text detection and recognition algorithm. The neural network model under deep learning can automatically extract image features, refine the feature matrix through special convolution and pooling operations, and automatically optimize network parameters. At the same time, with the help of high-performance computing platforms and large-scale data sets, methods based on deep learning have made great breakthroughs in the field of computer vision in recent years and are currently the main technical direction for studying text recognition problems from all walks of life. Aiming at the unique structure of Chinese characters, this paper uses convolutional neural

network to achieve effective overcoming of interference with natural scenes and to detect and recognize scene text.

2. Improved CTPN English Text Detection Algorithm

English text detection refers to extracting the text area in the image. There are many ways to extract, including rectangular box extraction, polygon extraction, and pixel-level extraction. This paper draws on the multiscale network structure of inception and designs an improved English text detection algorithm based on CTPN. The algorithm uses a multiscale convolution structure to extract English text features and uses adaptive text lines to improve the CTPN algorithm. Use the improved convolutional neural network based on inception to extract the features of the input image, use the RPN to obtain the feature sequence, and input the feature sequence to the bidirectional long and short period memory network to achieve feature fusion, and then input each feature after fusion into two to predict the position and confidence of the text in the parallel fully connected network, and finally stitch the obtained text area to get the final text area. It can better adapt to the detection of English text and improve the detection efficiency of English text [4] as shown in Figure 1.

2.1. CTPN Algorithm. The CTPN algorithm is a text proposal network that combines a convolutional neural network and a recurrent neural network. This algorithm introduces the recurrent neural network to the task of English text detection for the first time. It can directly locate the English text sequence in the convolutional layer to a certain extent. The above solves the limitations of traditional character detection methods [5–8]. The main structure of CTPN is shown in Figure 2:

CTPN converts the problem of English text detection into a series of fine-grained texts and proposes a new anchor box regression mechanism based on the RPN network. First, the English text area is subdivided into each small area, and then the area of each area is predicted. The vertical position of the English text and the confidence of the text finally obtain a priori information of the position of the text area with high precision. The algorithm uses a recurrent neural network to connect the convolutional feature maps. This seamless connection allows the network to obtain the context information of the English text line, so that it can detect more challenging text lines. The algorithm can process multiscale English text in a single process, avoiding subsequent filtering and refinement operations [6, 9, 10].

The advantage of the CTPN algorithm: The English text box that needs to be detected is divided into a series of small text boxes with a fixed width, making the detection horizontal.

2.2. English Text Detection Model

2.2.1. Input Image Preprocessing. The input picture size and the number of channels are not the same. In order to make it conform to the structure of the network, the input picture needs to be converted into a single-channel image and its size is scaled to 32×100.

2.2.2. Sequence Feature Extraction Based on CNN. Sequence recognition models can be divided into explicit segmentation models and implicit segmentation models, as shown in Figure 3. The main challenge of traditional text recognition algorithms based on explicit segmentation models is the correct segmentation at the pixel or character level. The quality of the segmentation effect directly affects the subsequent recognition effect. However, implicit segmentation only needs to perform simple segmentation on the sample, and there is no requirement for the quality and accuracy of character-level segmentation. A word is a continuous string. In many natural scenes, there may be interfering factors such as adhesion or uneven lighting between characters, which makes it impossible for us to correctly segment each character. Therefore, we adopt the method of implicit segmentation and use CNN to slide the window on the image to extract the feature sequence [11].

Since the length of the words existing in nature generally does not exceed 26, a total of 26 subwindows are extracted as input. The window size of CNN is 32×25, and the sliding step size is 3. The sequence feature extracted by CNN is expressed as $x = \{x_1, x_2 \ldots x_T\}$, where $T = 26$.

The CNN model of the sequence feature extraction part is based on the structure of the VGG-Very Deep network, but the final full link layer is removed, as shown in Table 1. CNN extracts sequence features from the left-to-right sliding window. The feature at each moment is the union of all the feature maps corresponding to the window position, and it is pulled into a column vector as the final input feature with a dimension of 512.

2.2.3. Processing of Context Information Based on Two-Way LSTM. CNN is based on the interdependence of the sequence features extracted by hidden segmentation and contains rich context information, which will greatly improve the recognition effect of fuzzy, nonuniform illumination and occluded words [11]. At the same time, RNN has a powerful ability for sequence learning. For the problems of RNN, we use LSTM here to replace it.

LSTM is directional; it only uses past contextual information, but for word pictures, the contextual information before and after it is meaningful for recognition. Therefore, similar to [46], we merge two LSTMs, one of which is used for forward propagation and the other is used for backward propagation to realize a two-way LSTM. The abstraction level of the deep network structure is higher than that of the shallow network, and it has achieved remarkable results in the task of speech recognition. Therefore, this method also uses two-way LSTM to process the sequence. Each layer of LSTM has 512 memory cell modules, which correspond to the dimensions of the features extracted by CNN. The input layer has a total of 512 neurons, which are fully connected to their hidden layer. After the hidden layer, they are fully connected to the output layer. Then softmax is used to classify them. There

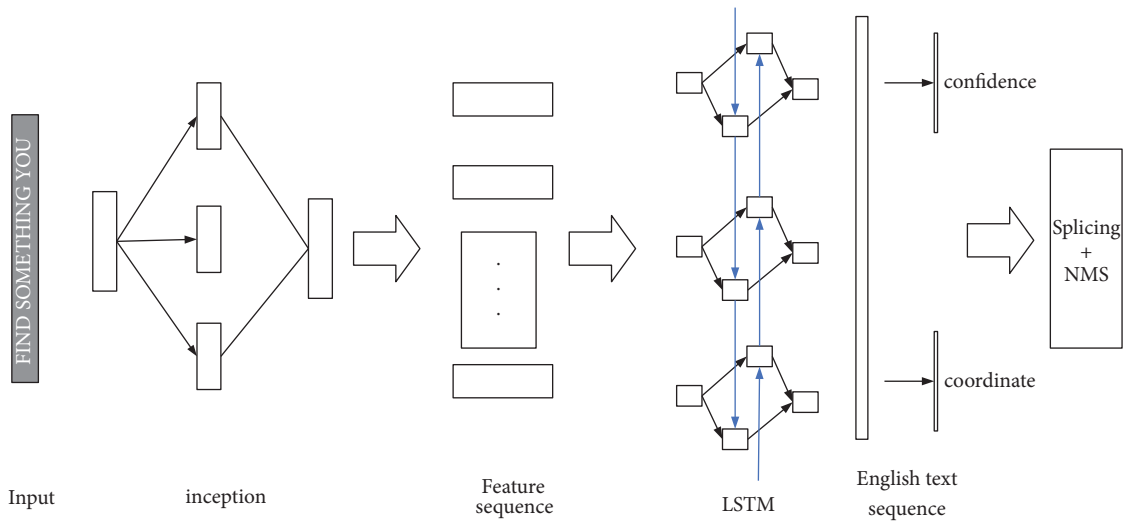


Figure 1: Improved CTPN flowchart.

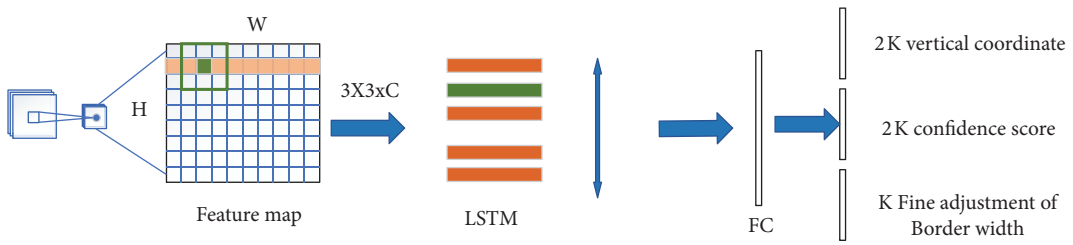


Figure 2: CTPN algorithm structure.

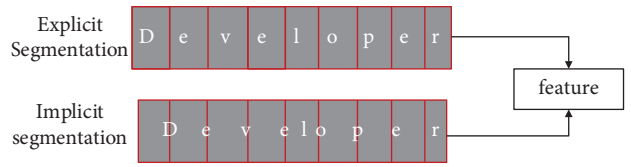


Figure 3: The difference between explicit segmentation and implicit segmentation.

Table 1: The structure and parameters of convolutional neural network.

Type	Number of channels	Nuclear size	Step size	Padding size
Convolutional layer 1	64	3×3	1	1
Maximum pooling layer 1		2×2	2	
Convolutional layer 2	128	3×3	1	1
Maximum pooling layer 2		2×2	2	
Convolutional layer 3	256	3×3	1	1
Convolutional layer 4	256	3×3	1	1
Maximum pooling layer 3		1×2	2	
Convolutional layer 5	512	3×3	1	1
Convolutional layer 6	512	3×3	1	1
Maximum pooling layer 4		1×2	2	
Convolutional layer 7	512	2×2	1	

are 36 categories (including ten numbers from 0 to 9 and twenty-six English letters from a to z that are not case sensitive). The result of the bidirectional LSTM prediction output is $y = \{y_1, y_2 \cdots y_T\}$, and its length is the same as the length of the input sequence feature.

2.2.4. CTC-Based Transcription. CTC is specifically designed for sequence labeling tasks, especially those input sequences that are difficult to segment into specific targets. In our recognition network, the CTC layer is directly connected to the output of LSTM, and its effect is similar to

the output layer of LSTM. It not only eliminates the need for presegmentation of the input image [12], but also allows us to achieve end-to-end training by minimizing the loss function.

2.3. Overall Framework Description and Algorithm Evaluation

2.3.1. Overall Framework Description. Assuming that N pictures are entered, then we have the following.

(1) First, scale N pictures to the same size, and normalize the pixels to the range of [−0.5,0.5] to ensure that gradient descent can be used to accelerate training.

(2) Input the normalized picture into the improved inception network, extract the feature map with multiple receptive fields, and the size of the obtained feature map is $N \times C \times H \times W$.

(3) Then use a 3×1 convolution window to slide on the feature map. Each sliding window generates k anchor boxes on the original image and obtains a feature vector of size $N \times C$ to represent the features of the original image area.

(4) The feature vector of each row is used as a set of feature sequences and input into the two-way long- and short-term memory network for encoding. The encoded features incorporate contextual information.

(5) Input the encoded features into two parallel fully connected networks, and predict 2k confidence scores and 2k vertical regression box positions.

(6) In the training phase, in order to facilitate the calculation of IOU, the original English text line splicing method of CTPN is still used. During the test, in order to obtain a more accurate proposal area for the slanted English text, the improved text line splicing method proposed in this paper is used.

(7) Use the improved NMS algorithm to suppress the regression box to obtain the optimal English text prediction box, and use the perspective transformation to correct the English text box as the input of the English text recognition network. The entire network is composed of multiple parts cascaded, but through ingenious construction, the various modules are combined into a whole, so that the network can be mapped from end to end. In the process of training, the network extracts features from the original picture without complicated preprocessing.

2.3.2. Algorithm Evaluation. The performance evaluation of the proposed algorithm is carried out by using the competition evaluation criteria corresponding to the IC15 database, which is measured by dividing the area of the overlapping part of the inspection result rectangle and the ground-truth rectangle by the area of the union part. The evaluation indicators used are recall rate, training speed, and prediction speed.

3. Improved CRNN English Text Recognition Algorithm

For scanned English text, traditional optical character recognition and other algorithms have become mature, but natural scene text is still very challenging due to the interference of complex backgrounds, fonts, lighting, textures, and viewing angles. After text detection, a series of long texts are obtained. Recognize these long texts. This chapter uses the CRNN algorithm to unify the feature extraction, sequence modeling, and transcription of the long image into a framework to realize the end-to-end recognition algorithm of the image. The CRNN algorithm has the characteristics of end-to-end training, recognition of any length, undefined vocabulary recognition, lightweight structure, and strong generalization ability [13].

The network architecture of CRNN consists of three parts in total, including a convolutional network used to extract features, a recurrent network used to identify sequence information, and a CTC layer used to transcribe and map the final label.

The processing process is as follows: the input English text image is passed through the convolutional layer to obtain a series of feature maps, these feature maps are compressed into features acceptable to the recurrent network, these features are predicted by frame, and then they are decoded and transcribed by CTC.

A sequence of labels is in the same format. The whole can be jointly trained through a loss function.The network architecture of the CRNN algorithm is shown in Figure 4. The algorithm is as follows.

In this chapter, on the CRNN network infrastructure, the role of increasing the antineural network layer is to artificially increase the learning difficulty of the algorithm, so as to achieve the function of the algorithm to recognize the occluded text, improve the recognition accuracy of the algorithm, and increase the robustness of the algorithm [14]. At the same time, the transcription layer adopts the combination of CTC and attention mechanism to improve the accuracy of transcoding. The structure diagram of the improved CRNN text recognition algorithm is shown in Figure 5.

3.1. Feature Extraction Layer. Based on the VGG16 improved convolutional neural network to construct the convolutional layer, use the convolutional neural network to extract. Enter the text characteristics of the English text image. When inputting the network, the height of the English text image needs to be scaled to a fixed value, such as 32 pixels, and the width is proportional. Then, the feature map extracted by the convolutional network is expanded from left to right in columns, converted into a feature sequence, and input to the antineural network layer [15].

3.2. Against the Neural Network Layer. Use the VGG16 network pair to counter the network to share features, but not to share parameters. The feature map generated by the VGG16 network is used as input, and the mask is generated through the mask network. The mask is used to

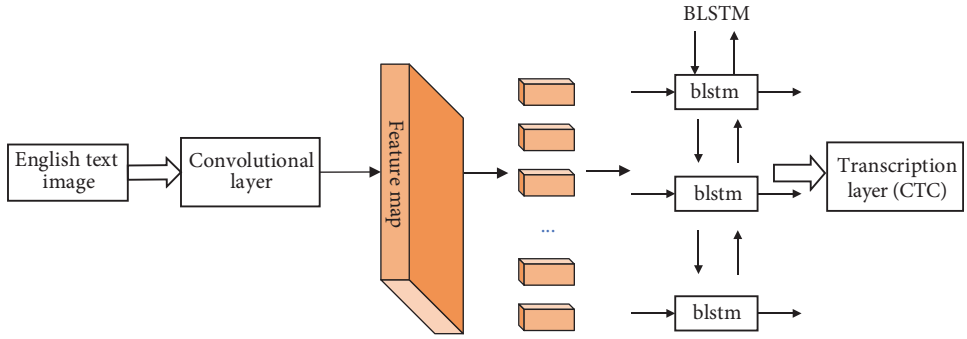


Figure 4: The network architecture of the CRNN algorithm.

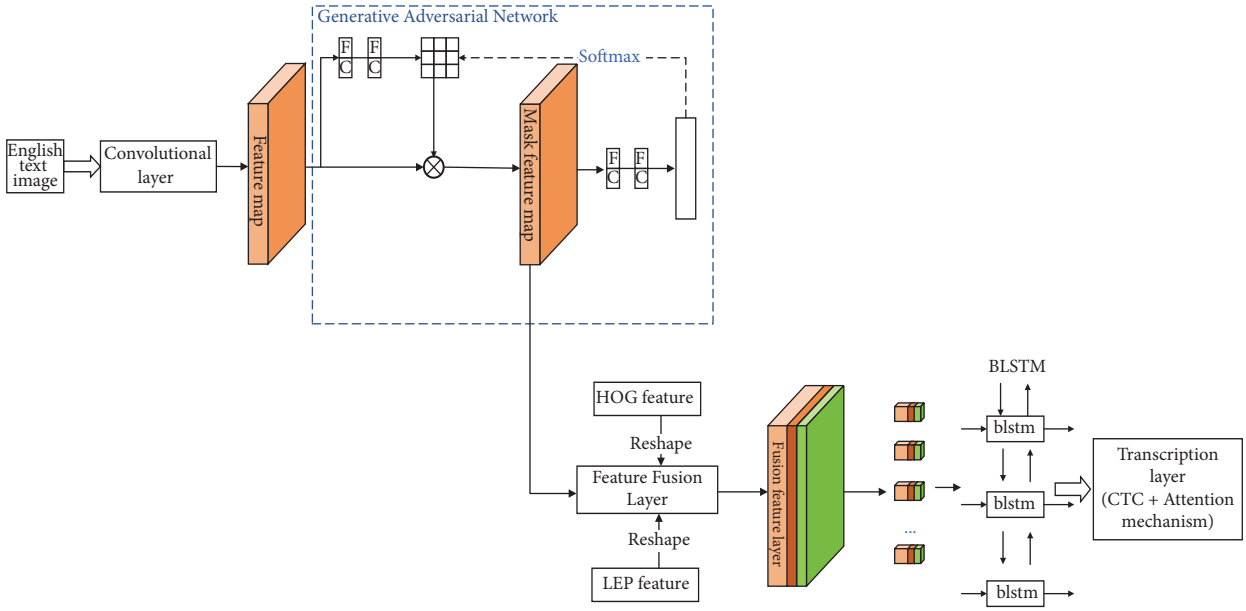


Figure 5: Improved CRNN text recognition algorithm.

determine which parts of the feature map should be subtracted to generate a masked feature map, and then send it to the classifier for judgment. The mask will automatically make corresponding adjustments according to the loss function. First, train the iterative CRNN network about 10,000 times. First, get a model that can basically be recognized; second, train the adversarial neural network separately to predict specific occluded parts. It first divides the feature map into 9 grids. In order to generate training information against the neural network, these 9 grids are occluded in sequence, and the grid with the largest classification loss after occlusion is the grid that is most worthy of occlusion. The training loss function of the antineural network is to classify and judge these 9 grids, and whether each grid is the most worthy of occlusion, so the output is a graph composed of classification probabilities. When using the output results, take the 1/2 pixels with the highest probability of being classified as "most worthy of occlusion" and randomly select 1/3 of these pixels for occlusion, and the remaining 2/3 without occlusion, to increase a certain random factor. Finally, the adversarial neural network and CRNN are jointly trained, and inspired by reinforcement learning, they focus on training the binarization mask that significantly reduces the classification effect.

3.3. Feature Fusion Layer. Extract the HOG feature and LBP feature of the text line image, and fuse them with the mask feature extracted in the previous step to generate a fusion feature map.

3.4. Cyclic Network Layer. Input the feature matrix into the two-way LSTM network to extract the sequence features of the text.

3.5. Transcription Layer. The essence of text recognition is a translation process, which converts an image sequence into a text sequence. The conversion system consists of two parts. The feature extraction of the text image is completed before and the features are encoded. Then the last step is to generate a feature decoder to convert features into text. Therefore, the conditional probability can be defined:

$$\log\ p(y|x) = \sum_{j=1}^{m} \log\ p\left(y_j|y_{<j}, s\right). \tag{1}$$

This article uses the structure of encoding and decoding to realize the recognition of the text. Specifically, each character decoded can be converted into a probability:

$$p\left(y_j|y_{<j}, s\right) = \text{soft}\max\left(g\left(h_j\right)\right). \tag{2}$$

$g(h_j)$ is the conversion function, h_j is the hidden unit of the RNN, and the calculation method is

$$h_j = f\left(h_{j-1}, s\right). \tag{3}$$

The function $f(h_{j-1}, s)$ uses the output of the hidden layer at the previous moment to calculate the current hidden state.

The model based on attention mechanism decoding directly predicts the output of the entire sequence without conditional independence, sexual hypothesis. Using the attention mechanism to decode without building a character-level language model at the output end, compared with CTC, the calculation speed and the recognition rate have been significantly improved. The model can be defined by the following recursive equation, using the previously calculated label sequence. Predict the probability distribution of the label at the current moment: the specific process is shown in equations (4)–(6):

$$P(y|x) = \prod_{u} P(y_u|x\ ,\ y_{1:\,u-1}), \tag{4}$$

$$h = \text{Encoder}(x), \tag{5}$$

$$y_u = \text{Attention Decoder}\left(h, y_{1:\,u-1}\right). \tag{6}$$

In practice, CTC decoding only considers the feature vector at the current moment, which is very restrictive, but there is no contextual information obtained. Supplement: therefore, the sequence obtained is not necessarily globally optimal. The attention mechanism scans all encoding vectors and calculates the value of each feature as shown in Figure 6. Weights, using global information for decoding, with lack of constraints, lead to excessive freedom in the decoding stage and slow network convergence. This paper uses the multitask loss function method to jointly train CTC and attention; using CTC as the attention a priori constraint [16], the loss function is as follows:

$$L_{\text{MTL}} = \lambda L_{\text{CTC}} + (1-\lambda)L_{\text{Attention}}. \tag{7}$$

Practice has proved that the grid convergence speed of the joint training of CTC and attention mechanism is faster, and the accuracy of decoding is greatly improved.

4. Algorithm Simulation Test and Result Analysis

In order to evaluate the improved CRNN English text detection and recognition algorithm, experiments were conducted on the standard data set of English text detection and recognition.

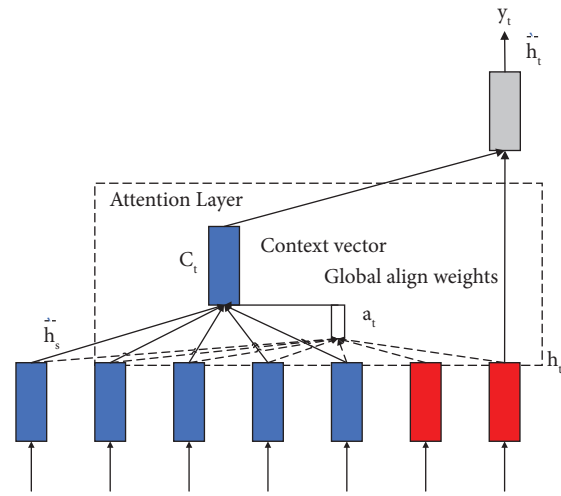


Figure 6: Attention mechanism.

Table 2: Mjsynth data set.

Category	Quantity
Total image	8919273
Number of images in the training set	7224612
Number of images in validation set	802734
Number of images in the test set	891927
LABEL type	88172

4.1. Experimental Design

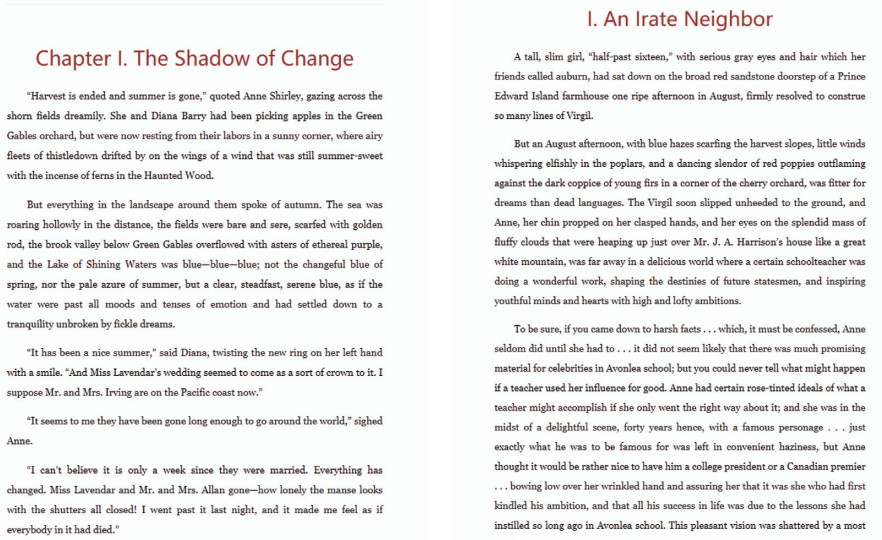
4.1.1. Training Set. The training set used by the algorithm in this chapter is the Mjsynth data set. This data set is a synthetic data set published by Jaderberg et al. At present, the number of English text images used in this data set has reached more than 9 million, which provides sufficient training samples for users to train. The data set contains nearly 90,000 labels. The algorithm in this chapter was trained on the Mjsynth data set and tested on the standard English text detection and recognition benchmark data set. The specific conditions of the Mjsynth training set are shown in Table 2.

It should be pointed out that when training the model, first iterate the CRNN network on the Mjsynth data set 10,000 times, then train the improved CRNN text detection and recognition network, and finally train the entire model.

4.1.2. Test Set. The algorithm in this chapter selects the benchmark data set ICDAR 2013 (IC13) and Street View Text (SVT) as the test set. ICDAR2013 (IC13) was proposed for scene text detection in the ICDAR 2013 Robust Reading Competition. It contains high-resolution images, 229 training images, and 233 test images, including English text. The comment is a rectangular box of words. It contains 1015 actual cropped word images. The SVT test data set consists of 249 Street View images collected from Google Street View. 647 word images were cut out from them. Each word image has a 50-word dictionary. This test set mainly tests English text [17–21].

Table 3: Time table consumed by different models.

Model structure	Recall rate (%)	Training speed (hours)	Forecast speed (seconds)
CTPN_convID	90.86	2.9	1–1.6
CTPN_LSTM	92.22	4.6	1.3–2.5
EAST	89.73	4	1.2–2



(a) (b)

Figure 7: Original image of text detection. (a) Test sample 1. (b) Test sample 2.

4.2. Result Analysis

4.2.1. Analysis of English Text Detection and Recognition Results. The optimization function uses the Adam optimizer, which can directly optimize the algorithm for end-to-end training, and each labeled anchor point needs to be calculated in advance before entering the model. Using momentum of 0.9 and weight decay of 0.0005, batch data is 1024, period is 50000, and each period is 50 steps. In the first 16000 iterations, the learning rate is set to 0.0001. On the core of Conv1D, after several experiments, 5 was finally adopted as the parameter. In the ICDAR2013 data set, the Conv1D kernel uses 7 as a parameter.

Because the feature extractor uses the VGG16 model, the transfer learning method is introduced here. Load the training weights of ImageNet directly into the VGG16 model. At the same time, some public text data sets are used to pretrain the model. In this way, when training self-labeled data, the overall loss has a starting point of about 1.3. The front-end network adopts VGG, and two cases where the loop module adopts LSTM and Conv1D are, respectively, tested. Also use 10,000 pictures as an experiment and train for 50,000 cycles. Training loss is recorded every 10 cycles.

In order to verify the performance of the model on the public data set, several algorithms were verified on the ICDAR2019 data set. The results are shown in Table 3.

It can be seen from 4.2 that, in the process of comparing the three models, no matter which data set, the accuracy rate and recall rate in the evaluation index are relatively close, which is acceptable in engineering. The training speed is a qualitative leap in the Conv1D model. This speed-up can speed up model iteration and accelerate the prediction process. The final effect is shown in Figures 7 and 8.

4.2.2. Analysis of English Text Recognition Results. The English text recognition experiment data uses 1 million English pictures, and the optimization function uses the SGD optimizer. Each batch of data uses 2048 pictures. Each cycle has 500 steps. The learning rate is divided into three stages: 0.004, 0.0004, and 0.00004. Both GRU and Conv1D use 40 cycles, and LSTM uses 50 cycles. First, experiments were conducted on the feature extraction network, using a shallow structure similar to VGG pairs and experimenting with different architectures of recurrent networks, using LSTM and GRU, respectively. The experimental process loss is shown in Figures 9(a) and 9(b). The coordinate is the function loss, and the abscissa is the period. Conv1D loss and acc are shown in Figures 10(a) and 10(b). In Figure 10(b), the ordinate is the accuracy of the verification set, and the abscissa is the period.

In the experiment, it was found that the accuracy rate is basically the same, but the training speed is very different. Here, we use 1 million pieces of data with a width of 286 pixels and a height of 32 pixels, and the training time is 95% accurate. The training time of each cycle is shown in Table 4.

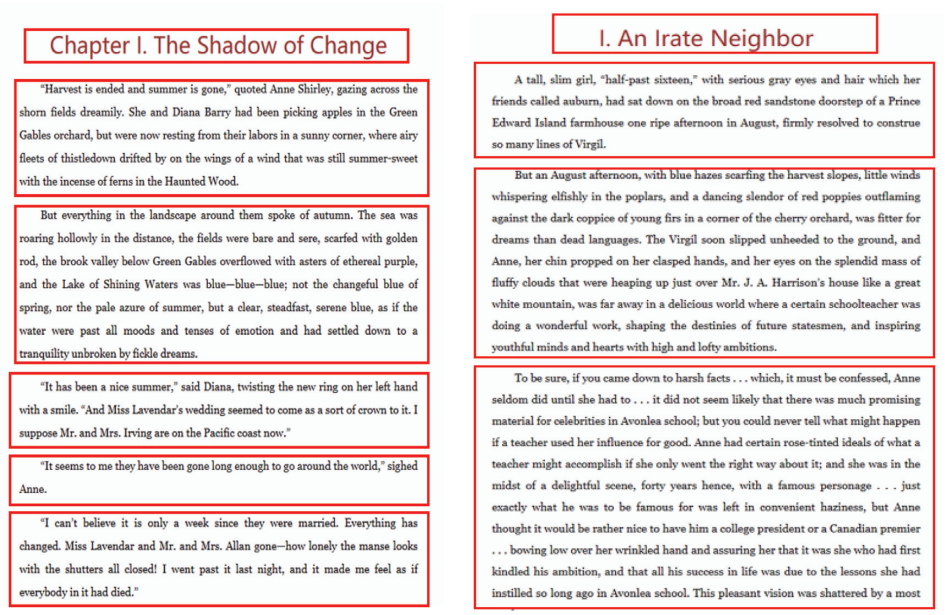


Figure 8: Text detection effect diagram. (a) Identification sample 1. (b) Identification sample 2.

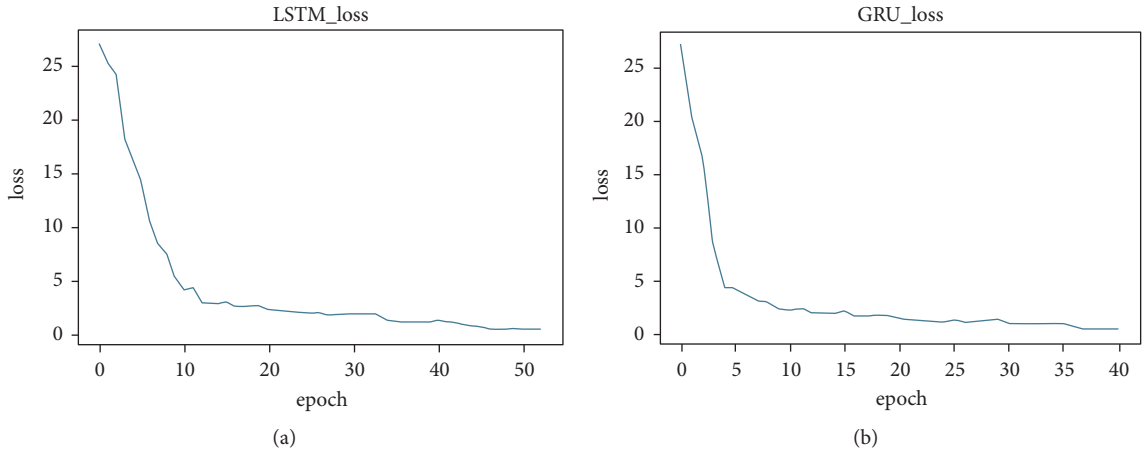


Figure 9: LSTM and GRU training curve. (a) LSTM loss curve. (b) GRU loss curve.

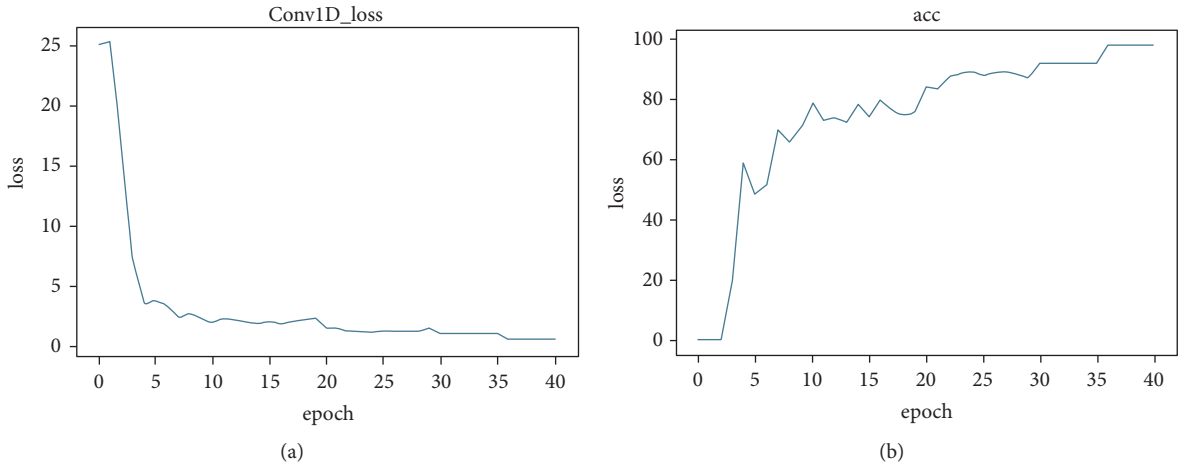


Figure 10: Conv1D training curve. (a) Conv1D loss curve. (b) Conv1D acc curve.

Table 4: Cycle structure diagram.

Model structure	Accuracy (%)	Life cycle	Training time per cycle (minutes)
LSTM	95	48	32
GRU	95	32	17
convID	95	36	5

Table 5: Accuracy test table.

Model structure	Accuracy (%)	Life cycle	Training time per cycle (minutes)
LSTM	98.38	59	32
GRU	98.38	41	17
convID	98.36	37	5

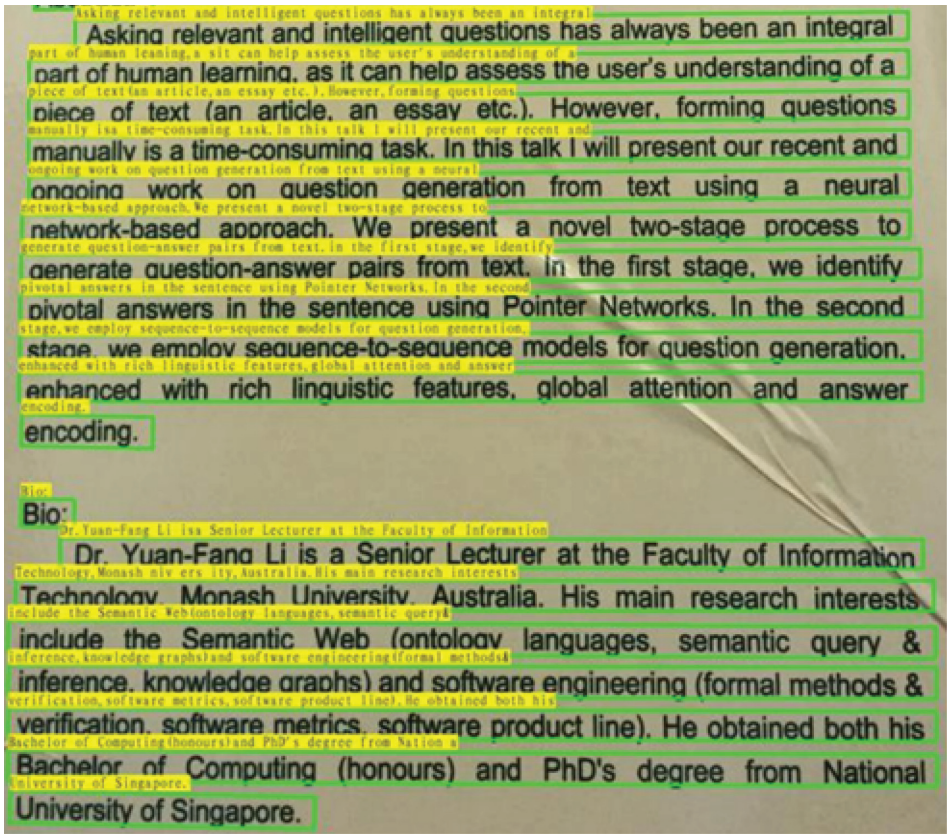


Figure 11: English text recognition graphics.

From Table 4, it can be found that the training duration of a single cycle of Conv1D is the smallest, which is 6.4 times higher than that of LSTM, while the cycle is reduced by one-third. Compared to the fastest GRU structure in the RNN structure, it is also 3.2 times faster. And from the loss curves in Figures 9(a) and 9(b) and 10(a), it can be seen that when the same loss function is used, the loss images are similar. It can be shown that the model can accomplish the task well under the current data set. The highest accuracy rate of the final experiment is shown in Table 5.

The final result is shown in Figure 11.

5. Conclusion

This paper studies how to improve the neural network model to improve the efficiency of text detection and recognition of English text images. The main research work is as follows.

In the aspect of English text detection, an improved CTPN multidirectional English text detection algorithm is proposed to solve the problem that the CTPN network can only detect the text in the approximate horizontal direction. The algorithm consists of a preprocessing model, a CTPN network English text positioning model, and a text box fusion model. Using the proposed text box fusion algorithm,

multiple candidate boxes obtained by CTPN are fused to obtain the best text box. Experiments show that, compared with the CTPN algorithm, the accuracy of this algorithm increases by 0.07, the recall rate increases by 0.217, and the comprehensive index increases by 0.16, which proves that the algorithm can detect multidirectional English text and has a higher detection accuracy.

In terms of natural scene text recognition, an improved CRNN English text recognition algorithm is proposed. The algorithm is based on the CRNN model. On the basis of the model, the adversarial network and fusion features are added to enable the recognition of occluded English text. The algorithm was tested on the IC13 and SVT data sets. Compared with the original CRNN model, this algorithm has a higher recognition accuracy. Especially on the SVT data set, the accuracy is increased by 0.065 without the constraint dictionary. It is proved that the algorithm can effectively recognize the occluded English text.

This paper uses the transfer training CRNN model with better English recognition effect to realize the variable length recognition of Chinese and English texts. And the recognition accuracy rate of 97.85% was obtained on the verification set. Since the recognition result of the entire English text line is a string with no spaces, this paper uses Viterbi algorithm to segment the English text string in order to present a better readable effect. This paper compares and analyzes the final detection and recognition results with the results of CTPN and DenseNet models and has obtained significant advantages in accuracy and test speed.

For the text detection algorithm proposed in this article, it is found that there is a high time complexity and space complexity, which will occupy a large amount of computing resources during detection. When running on embedded hardware such as tree mold, it can not meet the requirements at all, so the model needs to be further compressed in the future.

Data Availability

The dataset can be accessed upon request to the author.

Conflicts of Interest

The author declares that there are no conflicts of interest.

References

[1] M. H. Liao, B. G. Shi, X. Bai, X. G. Wang, and W. Y. Liu, "Textboxes: a fast text detector with a single deep neural network," in *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, pp. 4161–4167, AAAI, San Francisco, CA, USA, March 2017.

[2] 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," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, 2019.

[3] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and H. Belongie, "Feature pyramid networks for object detection," 2016, https://arxiv.org/abs/1612.03144.

[4] X. Zhao, W. Li, Y. Zhang, T. A. Gulliver, S. Chang, and Z. Feng, "A faster RCNN-based pedestrian detection system," in *Proceedings of the 2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, IEEE, Montreal, Canada, 18-September 2016.

[5] J. Ma, W. Shao, H. Ye et al., "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Transactions on Multimedia*, vol. 20, pp. 3111–3122, 2017.

[6] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao, "Detecting text in natural image with connectionist text proposal network," in *Proceedings of the 14th European Conference on Computer Vision*, pp. 56–72, Springer, Cham, Switzerland, October 2016.

[7] Y. L. Liu and L. W. Jin, "Deep matching prior network: toward tighter multi-oriented text detection," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3454–3461, IEEE, Honolulu, HI, USA, July 2017.

[8] B. G. Shi, X. Bai, and S. Belongie, "Detecting oriented text in natural images by linking segments," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3482–3490, IEEE, Honolulu, HI, USA, July 2017.

[9] A. Bissacco, M. Cummins, Y. Netzer, and H. N. Photoocr, "Reading text in uncontrolled conditions," in *2013 Proceedings of the 2013 IEEE International Conference on Computer Vision*, pp. 785–792, Sydney, Australia, December 2013.

[10] M. Busta, L. Neumann, and J. Matas, "Deep textspotter: an end-toendtrainable scene text localization and recognition framework," in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2223–2231, Venice, Italy, October 2017.

[11] Z. Cheng, F. Bai, Y. Xu, G. Zheng, S. Pu, and S. Zhou, "Focusing attention: towards accurate text recognition in natural images," in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5086–5094, Venice, Italy, October 2017.

[12] Z. Cheng, Y. Xu, F. Bai, Y. Niu, S. Pu, and S. Zhou, "Aon: towards arbitrarily-oriented text recognition," in *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5571–5579, Salt Lake City, UT, USA, June 2018.

[13] C. K. Chng and C. S. Chan, "Total-text: a comprehensive dataset for scene text detection and recognition," in *Proceedings of the 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, pp. 935–942, Kyoto, Japan, November 2017.

[14] J. Dai, K. He, Y. Li, S. Ren, and J. Sun, "Instance-sensitive fully convolutional networks," in *Proceedings of the Computer Vision - ECCV 2016*, pp. 534–549, Amsterdam, The Netherlands, October 2016.

[15] Y. Dai, Z. Huang, Y. Gao et al., "Fused text segmentation networks for multi-oriented scene text detection," in *Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR)*, pp. 3604–3609, Beijing, China, August 2018.

[16] J. Donahue, L. Anne Hendricks, S. Guadarrama et al., "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2625–2634, Boston, MA, USA, June, 2015.

[17] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin, "Convolutional sequence to sequence learning," in *Proceedings of the ICML*, pp. 1243–1252, Sydney, Australia, August 2017.

[18] R. B. Girshick, "Fast R-CNN," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1440–1448, Santiago, Chile, December 2015.

[19] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, Columbus, OH, USA, June 2014.

[20] L. G´omez and D. Karatzas, "Textproposals: a text-specific selective search algorithm for word spotting in the wild," *Pattern Recognition*, vol. 70, pp. 60–74, 2017.

[21] R. Gomez, B. Shi, L. Gomez et al., "Icdar2017 robust reading challenge on coco-text,"vol. volume 1, pp. 1435–1443, in *Proceedings of the 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, vol. volume 1, IEEE, Kyoto, Japan, November 2017.