*Computers, Materials & Continua* CMC, vol.61, no.1, pp.289-300, 2019

# Text Detection and Recognition for Natural Scene Images Using Deep Convolutional Neural Networks

**Xianyu Wu[1], Chao Luo[1], Qian Zhang[2], Jiliu Zhou[1], Hao Yang[1, 3, *] and Yulian Li[1]**

**Abstract:** Words are the most indispensable information in human life. It is very important to analyze and understand the meaning of words. Compared with the general visual elements, the text conveys rich and high-level moral information, which enables the computer to better understand the semantic content of the text. With the rapid development of computer technology, great achievements have been made in text information detection and recognition. However, when dealing with text characters in natural scene images, there are still some limitations in the detection and recognition of natural scene images. Because natural scene image has more interference and complexity than text, these factors make the detection and recognition of natural scene image text face many challenges. To solve this problem, a new text detection and recognition method based on depth convolution neural network is proposed for natural scene image in this paper. In text detection, this method obtains high-level visual features from the bottom pixels by ResNet network, and extracts the context features from character sequences by BLSTM layer, then introduce to the idea of faster R-CNN vertical anchor point to find the bounding box of the detected text, which effectively improves the effect of text object detection. In addition, in text recognition task, DenseNet model is used to construct character recognition based on Kares. Finally, the output of Softmax is used to classify each character. Our method can replace the artificially defined features with automatic learning and context-based features. It improves the efficiency and accuracy of recognition, and realizes text detection and recognition of natural scene images. And on the PAC2018 competition platform, the experimental results have achieved good results.

**Keywords:** Detection, recognition, resnet, blstm, faster R-CNN, densenet.

## 1 Introduction

With the development of multimedia information technology, the technology of obtaining useful information from large data has broad prospects. The information contained in the text is direct and effective. The digitization of text information is of great significance to improve the ability of multimedia retrieval, industrial automation and scene understanding.

[1] School of Computer Science, Chengdu University of Information Technology, Chengdu, 610225, China.

[2] School of Computer Science, University of Nottingham Jubilee Campus, NG8 1BB, UK.

[3] School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, 610054, China.

* Corresponding Author: Hao Yang. Email: vhaoyang@gmail.com.

At present, the traditional document detection technology has made great achievements. However, the detection and recognition of complex natural scene images has greater exploration space and application prospects. Compared with the traditional document image, the biggest difference between the natural image and the traditional document is that the background of the natural scene image is complex and changeable, and the image has the phenomena of distortion, incompleteness, blurring and fracture. There will also be noise, illumination, low resolution and angle interference in the image. Text detection and recog- nition in natural scene images is an important part of the field of computer vision [Vondrick, Khosla, Malisiewicz et al. (2012); Giordano, Murabito, Palazzo et al. (2015); Yang (2002); Sebe, Gevers, Dijkstra et al. (2006); Dong, Loy, He et al. (2016)]. And it is also a challenging task and is still in the exploratory period.

The flow chart of text detection and recognition in traditional natural scenes is shown in the following Fig. 1. The first step is to get the image containing the characters to be recognized through image information acquisition and analyze its structure. In the second step, image processing methods such as threshold operation are used to denoise and correct the measured object. The third step, because of the particularity of text information, requires row and column segmentation to detect a single character or several characters. In the fourth step, the segmented character image is imported into the recognition model for processing, and the character information in the original image is obtained.


**Figure 1:** Text detection and recognition flow chart

The key and difficulty of character detection and recognition in natural scenes is character feature extraction. Although a lot of work has been done to define a good set of text features, most of the features used in practical applications are not universal. In extreme cases, many features are almost invalid or even impossible to extract, such as strokes, shape features and so on. On the other hand, defining and extracting artificial features is a time-consuming and energy-consuming task.

Therefore, text detection and recognition of complex natural images have great pressure and challenges. We try to combine the deep learning neural network technology with text detection technology, and propose an effective text detection method, which provides a new practical method to solve the text detection problems in natural scene images. In this paper, we use ResNet network to extract high-level visual features from the bottom pixels in text detection [He, Zhang, Ren et al. (2015)], we need these high-level features to express more useful content. The BLSTM layer is used to extract the context features of character sequences [Tian, Huang, He et al. (2016)]. However, the idea of faster R-CNN vertical anchor is introduced to find the boundaries of detected text, which effectively improves the effect of text detection [Ren, He, Girshick et al. (2017)]. In addition, in text recognition tasks, DenseNet model is used to construct text recognition based on Kares, and the output of Softmax is used to classify each character. Finally, combine with corpus to find corresponding characters. The experimental results show that the method achieves

good results in text detection and recognition of natural scene images. It implements an efficient framework for depth learning [Schmidhuber (2015); Sun, Wang and Tang (2014); Glorot, Bordes and Bengio (2011)], which can support a variety of neural network structures and provide a series of effective training strategies. The framework preliminarily validates the effectiveness of text detection and recognition method based on deep learning in natural scene images.

## 2 Related work

Text detection is the process of converting image information into a sequence of symbols that can be represented and processed by a computer. There are many related studies on text detection in natural scene images [Dai, Huang, Gao et al. (2017); Liu, Liang, Yan et al. (2018)]. Through the study of characters, the effective feature information of characters is extracted, and the text area in the image is detected accurately and effectively. The task of text recognition can be regarded as a special translation process: translating image signals into natural languages. This is similar to speech recognition and machine translation. From a mathematical point of view, they will contain a large number of noise input sequences, and form a set of given tag output sequences through automatic learning model.

FCN (Full convolutional network) is the basic network that removes the full connection (fc) layer [Shelhamer, Long and Darrell (2014)]. It was originally used to implement semantic segmentation tasks. The advantage of FC is that it uses upsampling operations such as deconvolution and unpooling to restore the feature matrix to the size close to the original image, and then makes category prediction for the pixels at each location, so as to recognize the clearer object boundary. In the detection network based on FCN, the object boundaries are predicted directly according to the high resolution feature map instead of returning to the object boundaries through candidate regions. FCN is more robust in predicting irregular object boundaries than Faster-RCNN because it does not need to define the ratio of candidate box length to width before training. Because of the high pixel resolution of the last layer feature map of FCN network, and the need to rely on clear text strokes to distinguish different characters (especially Chinese characters) in the task of text recognition, FCN network is very suitable for extracting text features. When FCN is used for text recognition tasks, each pixel in the last layer of feature graph will be divided into two categories: text line (foreground) and non-text line (background).

In EAST (Efficient and Accuracy Scene Text Detection Pipeline) model, the full convolution network (FCN) is used to generate multi-scale fused feature maps, and then pixel-level text block prediction is performed directly. In this model, there are two kinds of text area tagging, which are revolving rectangle box and arbitrary quadrilateral. The model has better effect in detecting English words and less effect in detecting Chinese long text lines. Perhaps, according to the characteristics of Chinese data targeted training, the detection effect still has room for improvement.

FTSN (Fused Text Segmentation Networks) model uses segmentation network to support skewed text detection [Dai, Huang, Gao et al. (2017)]. It uses Resnet-101 as the basic network and uses multi-scale fusion feature maps. The annotation data includes the pixel mask and the border of the text instance, and the joint training of pixel prediction and border detection is used.

FOTS (Fast Oriented Text Spotting) is an end-to-end learnable network model for synchronous training of image text detection and recognition [Liu, Liang, Yan et al. (2018)]. Detection and recognition tasks share the convolution feature layer, which not only saves computing time, but also learns more image features than two-stage training. RoI Rotate is introduced to generate oriented text regions from convolutional feature maps, which can support the recognition of skewed text.

Traditional methods of text detection and recognition and some methods of text detection and recognition based on in-depth learning are mostly multi-stage, which need to be optimized in many stages in the training process, which will inevitably affect the effect of the final model, and is very time-consuming.

## 3 Method and architecture framework

### *3.1 Mothod*

#### *3.1.1 Detection*

We first detect the text and digital regions in the training image, and then recognize the detected text regions. However, the complexity of training data with complex scenes, fonts in images and shooting angles may be difficult for human eyes to distinguish, except for computers. This is also the difficulty of our work. Our detection method is based on deep convolution neural network (CNN) in natural scene images, using ResNet network to obtain higher-level features from the underlying pixels. We choose ResNet network model to detect text in images, because its detection quality is higher than other network models, and it is suitable for detecting rough and complex image information. The BLSTM layer is used to extract the context features of character sequences, and then the idea of faster R-CNN vertical anchor is introduced to detect text boundaries, which effectively improves the effect of text detection.

**ResNet** He et al. [He, Zhang, Ren et al. (2015)] proposed a deep residual network called ResNet, the greatest contribution of what is to avoid the problem of gradient disappearance or gradient explosion as the number of network layers increases using a residual network. It accelerates the speed of convergence, confirms the accuracy of the deep neural network, and deepens it. A shortcut connection is added to every residual unit. From a functional point of view, the change in identity is increased. From the forward propagation, the introduction of identity transformation can render the adjustment of the network parameters more powerful. From the backpropagation, direct, forward-layer propagation is added to the error term to alleviate the problem of gradient reduction. Thus, the problem of gradient disappearance is solved. The residual unit formula can be expressed as follows:

$$x=R(x)=\sigma(F(x, W))+x, \tag{1}$$

where $x$ represents the output of the residual unit, $x$ represents its input, $F$ is also called the residual function, $W$ is a weight matrix, and $\sigma$ represents the ReLU activation function.

**Faster R-CNN** as a test network framework [Ren, He, Girshick et al. (2017); Ruohan Meng and Sun (2018)], its goal is to find the compact surrounded by detecting the Bounding Box around the object. It introduces RPN (Region Proposal Network) on the basis of Fast RCNN detection framework to quickly generate multiple candidate Region reference frames (anchor) that are close to the length and width of the target object. It

generates the regional features of normalized fixed sizes through the Region of Interest Pooling layer for multiple size reference boxes. It uses the shared CNN to simultaneously input Feature Maps to the above RPN network and ROI Pooling layer, thus reducing the number of convolutional layer parameters and the amount of calculation.

### *3.1.2 Recognition*

We use DenseNet network model to construct our character recognition model based on Kares [Huang, Liu, Laurens et al. (2017)]. The main advantage of DenseNet is to enhance the dissemination of features and encourage feature reuse. The core idea is to create a cross-layer connection to connect the front and back layers of the network, which is very suitable for scene character recognition. We use the Softmax classifier as the final output layer [Liu, Wen, Yu et al. (2016)]. Softmax function is based on Softmax regression. It is a supervised learning algorithm.

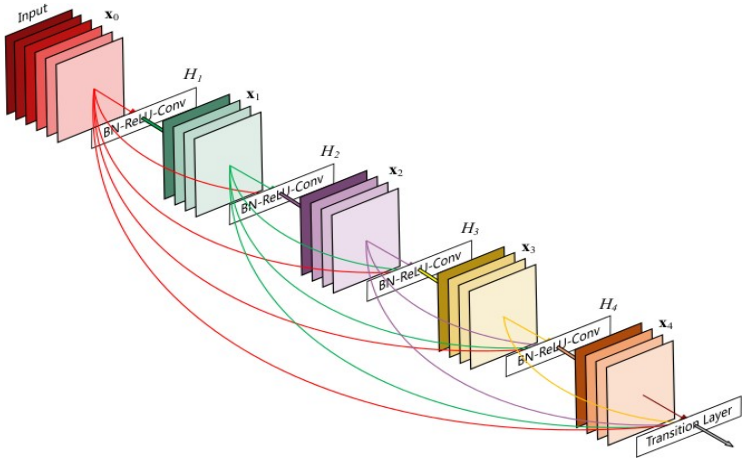


**Figure 2:** DenseNet model

**DenseNet** DenseNet is a convolutional neural network with dense connections. In this network, there is a direct connection between any two layers, the input of each layer of the network is the union of the output of all the previous layers, and the feature maps learned by this layer will be passed directly to all the layers behind it as input. The structure of a block is as follows: BN-ReLU-Conv (1*1)-BN-ReLU-Conv (3*3), while a DenseNet is composed of several such blocks. The layer between each DenseNet block is called transition layers, which is composed of BN-Conv (1*1)-average Pooling (2*2). It can be said that DenseNet has absorbed the most essential part of ResNet, and has done more innovative work on it, making the network performance further improved. Dense connection, alleviating the problem of gradient disappearance, enhancing feature propagation, encouraging feature reuse, greatly reducing the amount of parameters.

$$x_l = H_l([x_0, x_1, \ldots, x_{l-1}]), \tag{2}$$

The above formula represents the DenseNet module. $[x_0, x_1, \ldots, x_{l-1}]$ The output feature map from 0 to $l\_1$ is concatenation. Conatenation is the merger of channels, just like Inception. $H_l$ includes convolutions of BN, ReLU and kernel.

### *3.2 Architecture framework*

Our network structure is shown in the Fig. 3. We first map the data set to the feature space through the ResNet network, and then input the final convolution map to the 3*3 convolution map. These features will be used to predict the corresponding category information and location information of K anchors (anchor definition is similar to faster RCNN). BLSTM (bidirectional LSTM) is used to recursively add sequential windows to each row, where the convolution characteristics of each window (3*3*C) are used as input for 256-dimensional BLSTM (bidirectional, two 128D LSTM). The BLSTM layer is followed by the 512 FC layer, which jointly outputs text or non-text probabilities, Y-axis coordinates and the lateral refinement offset of the *K* anchor. We set the threshold of the detected text area to 0.8, and anchor values above 0.8 are identified as text, and vice versa. The recognition network structure is DenseNet. Finally, we will use the trained model to test the untrained data. Through the trained model, text information in natural scene images can be detected and recognized.

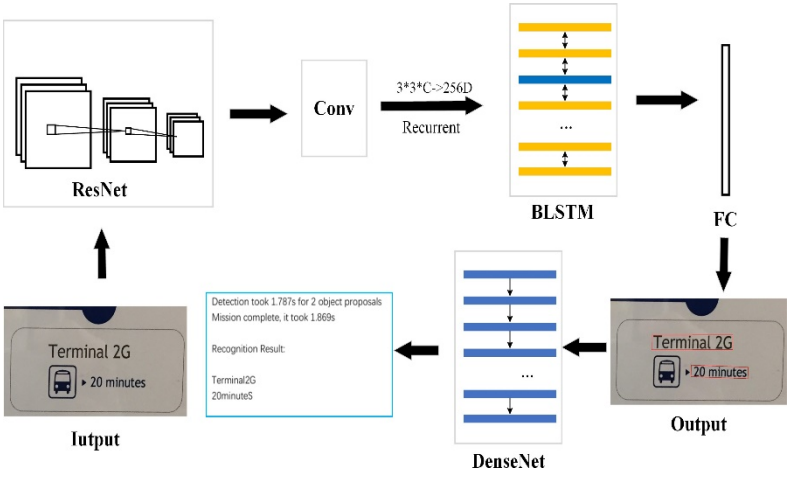


**Figure 3:** Model architecture for detection and recognition

## 4 Experiments and discussion

In this experiment, the data set is VOCdevkit. We used all data set to divide into five parts randomly. According to the ratio of 3:1:1, we made training set, validation set and test set respectively. The training process is iterated by 50 epoch, and the training iteration is tested by 1 epoch. Finally, the accuracy of the training model is tested by the test set. We will introduce this part from three aspects. Firstly, the results of text detection are presented and compared with the existing advanced methods; secondly, the results of text recognition are presented and compared with the existing advanced methods; finally, the factors affecting the experimental performance are analyzed.

To complete this experiment, we used P/R/F as a measure of text detection, and all of the experiments were implemented by Tensorflow and Keras frameworks in Python 2.7. We trained the network M40 GPU on NVIDIA Tesla to execute the best test data set preservation using the SGD solver and model for further analysis.

### *4.1 Experimental procedure*

#### *4.1.1 Image text detection*

We made all the images into a data set in VOC format, using folders Annotation, Image Sets, and JPEG Images. Where the folder Annotation mainly stores xml files, each xml corresponds to an image, and each xml stores the location and category information of the tagged targets, and the name is usually the same as the corresponding original image; while in ImageSets we only need to use the Main folder, which contains some text files, Typically train. txt, test. txt, etc., the contents of this file are the names of the images that need to be trained or tested (no suffixes, no paths); the JPEG Images folder contains the original images that we have named according to the uniform rules. We input the data set into the network for training, iteration 50 Epoch, each input 30 pictures (batchsize=30), each iteration is completed to verify together, and finally save the best model as a detection model.

We evaluated our test results on two benchmark data sets ICDAR 2011 [Minetto, Thome, Cord et al. (2010)] and ICDAR 2013 [Karatzas, Shafait, Uchida et al. (2013)]. These test data sets are challenging, including different angles, very small scales, and low resolution. Tab. 1 shows our assessment results on two public data sets. Comparing our work with other existing methods [Huang, Lin, Yang et al. (2013); Yao, Bai and Liu (2014); Yin, Yin and Huang et al. (2013); Buta, Neumann and Matas (2015); Neumann and Matas (2015); Zhang, Shen, Yao et al. (2015); Tian, Pan and Huang (2016)], it is easy to find that our work achieves optimal performance on both data sets. Especially on the ICDAR 2013 data set [Karatzas, Shafait, Uchida et al. (2013)], it has increased from P of 0.85 to 0.91 over the latest method TextFlow [Tian, Pan, Huang et al. (2016)], and has made tremendous progress in R/F. In addition, we also test our method in VOCdevkit data set in Fig. 4. The text regions in natural scene images are automatically detected by our model and surrounded by red boxes. It can be found that our approach works perfectly in these challenging situations, some of which are difficult for many previous approaches. It can be seen that our method is very suitable for challenging and complex natural scene image detection. In addition, we also validated the results on MSRA-TD500 dataset, and the experimental results also showed amazing results. It can be seen that our method is suitable for many data sets in natural image text detection. It has a good generalization ability.

**Table 1:** State-of-the-art results on the ICDAR 2011 and ICDAR 2013. P represents precision, R represents recall and F represents F-measure. The best performance is indicated in bold and red

| ICDAR 2011 | | | | ICDAR 2013 | | | |
|---|---|---|---|---|---|---|---|
| Method | P | R | F | Method | P | R | F |
| [Huang, Lin, Yang et al. (2013)] | 0.82 | 0.75 | 0.73 | [Yin, Yin and Huang (2013)] | 0.88 | 0.66 | 0.76 |
| [Yao, Bai and Liu (2014)] | 0.82 | 0.66 | 0.73 | [Neumann and Matas (2015)] | 0.82 | 0.72 | 0.77 |
| [Yin, Yin and Huang (2013)] | 0.86 | 0.68 | 0.76 | FASText [Buta, Neumann and Matas (2015)] | 0.84 | 0.69 | 0.77 |
| [Zhang, Shen, Yao et al. (2015)] | 0.84 | 0.76 | 0.80 | [Zhang, Shen, Yao et al. (2015)] | 0.88 | 0.74 | 0.80 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| TextFlow [Tian, Pan, Huang et al. (2016)] | 0.86 | 0.76 | 0.81 | TextFlow [Tian, Pan, Huang et al. (2016)] | 0.85 | 0.76 | 0.80 |
| Our | **0.87** | **0.78** | **0.82** | Our | **0.91** | **0.80** | **0.83** |

*4.1.2 Image text recognition*

We use DenseNet to train a text recognition model. Our training data set uses VOCdevkit to iterate over 50 epochs, input 30 pictures (batch size=30) at a time, verify each iteration, and finally save the best model as the prediction model. We put the image which has detected the text area into the trained model for recognition, and combine with corpus to find corresponding characters. The Tab. 2 shows our final recognition results. We can see that the recognition rate of LeNet method on VOCdevkit data set is 0.872 [Yu, Jiao and Zheng (2015)], while that of NinNet method is only 0.824 [Lin, Chen and Yan (2013)]. The accuracy is improved to 0.933 by using VGGNet method [Wang, Sheng, Huang et al. (2015)]. In our method, the introduction of DenseNet can get the accuracy of 0.939, which shows the effectiveness of our method.

The Fig. 5 is a recognition effect map on VOCdevkit and MSRA-TD500 data set. As can be seen from the figure, our method is very effective in text recognition of natural images. Not only can English characters be recognized, but also Chinese characters can be basically recognized. Although there are still some mistakes, there are also a lot of objective factors in it. Such as, font deformations, lighting and shooting angles. In addition, there is also a lack of the ability to correctly detect the impact of text area on the subsequent recognition of characters.

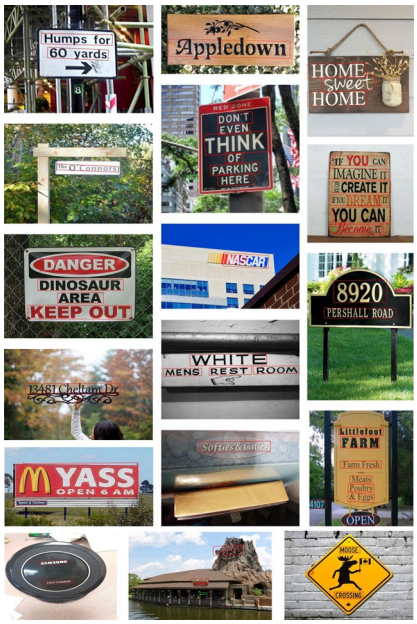


**Figure 4:** Visual effect for detection on VOCdevkit data set. Red boxes are the text regions and green boxes are the non-text regions

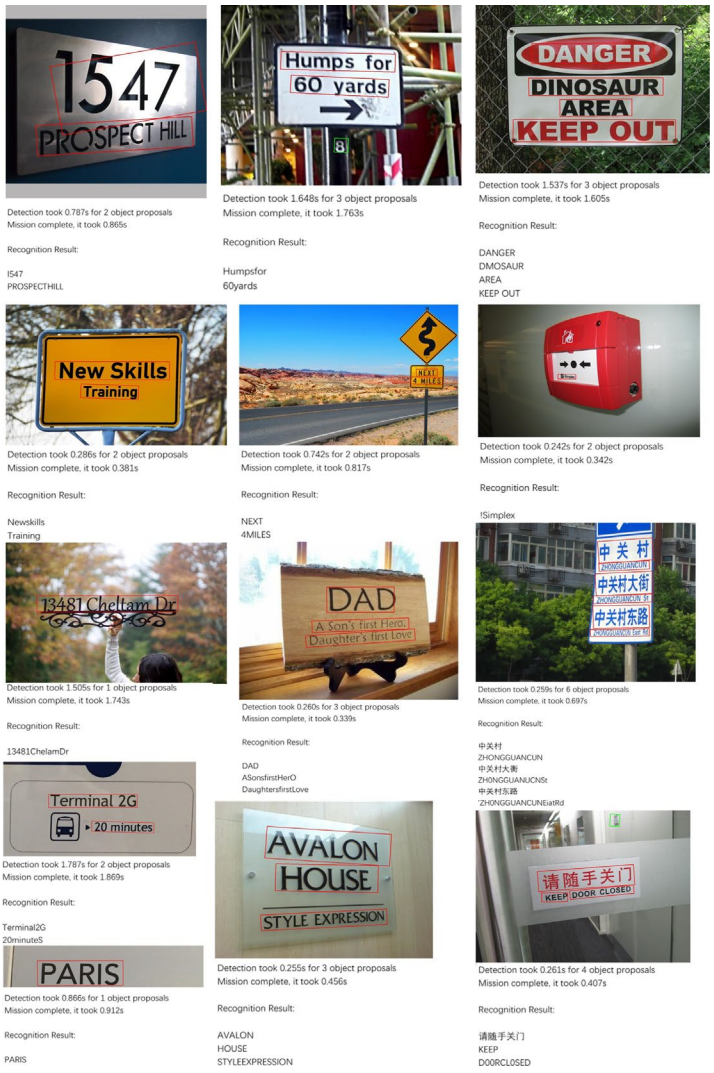


**Figure 5:** Visual effect for recognition on VOCdevkit and MSRA-TD500 data set. It can be shown that our method has good generalization on different data sets

**Table 2:** Comparisons of recognition results between existing advanced methods and our proposed methods on VOCdevkit datasets

| Model | Training sets | Testing sets | accuracy |
|---|---|---|---|
| LeNet [Yu, Jiao and Zheng (2015)] | 254800 | 10000 | 0.872 |
| NinNet [Lin, Chen and Yan (2013)] | 254800 | 10000 | 0.824 |
| VggNet [Wang, Sheng, Huang et al. (2015)] | 254800 | 10000 | 0.933 |
| DenseNet (our) | 254800 | 10000 | 0.939 |

### *4.2 Discussion*

We proposed a text detection and recognition for natural scene images using deep convolutional neural networks. It builds on CNN by adding ResNet network, and BLSTM layer to our model. Finally, using DenseNet Network Model to Construct Text Recognition. Our method can outperform the exiting methods. We analysis the following important factors which can enable our network to perform better.

**ResNet** Residual learning showed good performance in our proposed method. It solved the problem by training the network becomes challenging with increasing depth. In the process of training, forward propagation enabled the higher values in the results. Back propagation alleviated the problem of gradient reduction, thus resolving the problem of gradient disappearance. The final experimental results show that the proposed method is effective.

**BLSTM** CNN learns spatial information in the receptive field. Moreover, with the deepening of the network, the features that CNN learns become more and more abstract. For text sequence detection, it is obvious that the abstract spatial features learned by CNN are needed. In addition, the sequence feature of text is also helpful for text detection. For horizontal text lines, each text segment is connected, so a network structure of BLSTM is adopted to make the detection result more robust.

**DenseNet** We choose DenseNet, whose greatest advantage is to enhance the dissemination of features and encourage feature reuse. The core idea is to create a cross-layer connection to connect the front and back layers of the network. DenseNet connection is dense, which can alleviate the problem of gradient disappearance, enhance feature propagation, encourage feature reuse, greatly reduce the amount of parameters, and is very suitable for scene character recognition.

## 5 Conclusion

This paper presents a text detection and recognition method for natural scenery images based on depth convolution neural network. The experimental results show that this method is superior to the existing methods in accuracy. Although it detects the wrong area in a few cases, this is also due to objective reasons, pictures, many professionals are still trying to solve this problem. In the future, in order to accurately and effectively detect text areas from natural scene images and correctly identify them, we are committed to improving the model. Such as, the improvement of new network structure and training strategy.

**Acknowledgement:** This study was supported by the Scientific Research Foundation (KYTZ201718) of CUIT.

## References

**Buta, M.; Neumann, L.; Matas, J.** (2015): Fastext: efficient unconstrained scene text detector. *IEEE International Conference on Computer Vision*, pp. 1206-1214.

**Dai, Y.; Huang, Z.; Gao, Y.; Xu, Y.; Chen, K. et al.** (2017): Fused text segmentation networks for multi-oriented scene text detection. arXiv: 1709.03272.

**Dong, C.; Loy, C. C.; He, K.; Tang, X.** (2016): Image super-resolution using deep covolutional networks. *IEEE Transactions on Pattern Analysis & Machine Intelligence*,

vol. 38, no. 2, pp. 295-307.

**Giordano, D.; Murabito, F.; Palazzo, S.; Spampinato, C.** (2015): Superpixel-based video object segmentation using perceptual organization and location prior. *Computer Vision and Pattern Recognition*, pp. 4814-4822.

**Glorot, X.; Bordes, A.; Bengio, Y.** (2011): Domain adaptation for large-scale sentiment classification: a deep learning approach. *International Conference on International Conference on Machine Learning*, pp. 513-520.

**He, K.; Zhang, X.; Ren, S.; Sun, J.** (2015): Deep residual learning for image recognition. *IEEE Conference on Computer Vision & Pattern Recognition*, pp. 770-778.

**Huang, G.; Liu, S.; Laurens, V. D. M.; Weinberger, K. Q.** (2017): Condensenet: an efficient densenet using learned group convolutions. arXiv: 1711.09224.

**Huang, W.; Lin, Z.; Yang, J.; Wang, J.** (2013): Text localization in natural images using stroke feature transform and text covariance descriptors. *IEEE International Conference on Computer Vision*, pp. 1241-1248.

**Karatzas, D.; Shafait, F.; Uchida, S.; Iwamura, M.; Bigorda, L. G. I. et al.** (2013): Icdar 2013 robust reading competition. *International Conference on Document Analysis and Recognition*, pp. 1484-1493.

**Lin, M.; Chen, Q.; Yan, S. C.** (2013): Network in network. arXiv: 1312.4400.

**Liu, W.; Wen, Y.; Yu, Z.; Yang, M.** (2016): Large-margin softmax loss for convolutional neural networks. *International Conference on International Conference on Machine Learning*, vol. 48, no. 10, pp. 507-516.

**Liu, X.; Liang, D.; Yan, S.; Chen, D.; Qiao, Y. et al.** (2018): Fots: fast oriented text spotting with a unified network. *Computer Science-Computer Vision and Pattern Recognition*, arXiv: 1801.01671.

**Minetto, R.; Thome, N.; Cord, M.; Fabrizio, J.; Marcotegui, B.** (2010): Snoopertext: a multiresolution system for text detection in complex visual scenes. *IEEE International Conference on Image Processing*, pp. 3861-3864.

**Neumann, L.; Matas, J.** (2015): Efficient scene text localization and recognition with local character refinement. *International Conference on Document Analysis and Recognition*, pp. 746-750.

**Ren, S.; He, K.; Girshick, R.; Sun, J.** (2017): Faster R-CNN: towards real-time object de- tection with region proposal networks. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 39, no. 6, pp. 1137-1149.

**Ruohan, M.; Steven, G. R.; Jin, W.; Sun, X.** (2018): A fusion steganographic algorithm based on faster R-CNN. *Computers, Materials & Continua*, vol. 55, no. 1, pp. 1-16.

**Schmidhuber, J.** (2015): Deep learning in neural networks: an overview. *Neural Network*, vol. 61, pp. 85-117.

**Sebe, N.; Gevers, T.; Dijkstra, S.; Weije, J. V. D.** (2006): Evaluation of intensity and color corner detectors for affine invariant salient regions. *Conference on Computer Vision and Pattern Recognition Workshop*, pp. 18.

**Shelhamer, E.; Long, J.; Darrell, T.** (2014): Fully convolutional networks for semantic

segmentation. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 99, pp. 3431-3440.

**Sun, Y.; Wang, X.; Tang, X.** (2014): Deep learning face representation from predicting 10,000 classes. *Computer Vision and Pattern Recognition*, pp. 1891-1898.

**Tian, S.; Pan, Y.; Huang, C.; Lu, S.; Yu, K. et al.** (2016): Textflow: a unified text detection system in natural scene images. *IEEE International Conference on Computer Vision*, pp. 4651-4659.

**Tian, Z.; Huang, W.; He, T.; He, P.; Qiao, Y.** (2016): Detecting text in natural image with connectionist text proposal network. *European Conference on Computer Vision*, pp. 56-72.

**Vondrick, C.; Khosla, A.; Malisiewicz, T.; Torralba, A.** (2012): Inverting and visualizing features for object detection. arXiv: 1212.2278.

**Wang, L.; Sheng, G.; Huang, W.; Yu, Q.** (2015): Places205-vggnet models for scene recognition. arXiv: 1508.01667.

**Yang, M. H.** (2002): Kernel eigenfaces vs. kernel fisherfaces: face recognition using kernel methods. *IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 215-220.

**Yao, C.; Bai, X.; Liu, W.** (2014): A unified framework for multioriented text detection and recognition. *IEEE Transactions on Image Processing*, vol. 23, no. 11, pp. 4737-4749.

**Yin, X. C.; Yin, X. W.; Huang, K. Z.; Hao, H. W.** (2013): Robust text detection in natural scene images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 5, pp. 970-983.

**Yu, N.; Jiao, P.; Zheng, Y.** (2015): Handwritten digits recognition base on improved lenet5. *Control & Decision Conference*.

**Zhang, Z.; Shen, W.; Yao, C.; Bai, X.** (2015): Symmetry-based text line detection in natural scenes. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2558-2567.