Front. Optoelectron. 2020, 13(4): 418–424
https://doi.org/10.1007/s12200-019-0854-0

RESEARCH ARTICLE

A CCD based machine vision system for real-time text detection

Shihua ZHAO[1], Lipeng SUN[1], Gang LI[2], Yun LIU[1], Binbing LIU (✉)[3]

1 State Grid Hunan Electric Power Corporation Limited Research Institute, Changsha 410007, China
2 State Grid Hunan Electric Power Corporation Limited, Changsha 410007, China
3 School of Optical and Electronics Information, Huazhong University of Science and Technology, Wuhan 430074, China

Abstract Text detection and recognition is a hot topic in computer vision, which is considered to be the further development of the traditional optical character recognition (OCR) technology. With the rapid development of machine vision system and the wide application of deep learning algorithms, text recognition has achieved excellent performance. In contrast, detecting text block from complex natural scenes is still a challenging task. At present, many advanced natural scene text detection algorithms have been proposed, but most of them run slow due to the complexity of the detection pipeline and cannot be applied to industrial scenes. In this paper, we proposed a CCD based machine vision system for real-time text detection in invoice images. In this system, we applied optimizations from several aspects including the optical system, the hardware architecture, and the deep learning algorithm to improve the speed performance of the machine vision system. The experimental data confirms that the optimization methods can significantly improve the running speed of the machine vision system and make it meeting the real-time text detection requirements in industrial scenarios.

Keywords machine vision, text detection, optical character recognition (OCR), deep learning

1 Introduction

Optical character recognition (OCR) is a classic computer vision problem, it aims to convert document images with various types, such as handwriting, electronics, and printing, into machine-encoded text content. OCR is widely used in various industries and fields including life, industry and military. A typical CCD based machine vision system for real-time text detection and recognition is shown in Fig. 1.

Early OCR methods could only handle relatively simple application scenarios, such as printed electronic documents. In these application scenarios, the images containing text typically have higher image quality, lower noise, simpler image background, and less geometric distortion, so it is easier to extract and recognize individual characters from the images. With the rapid development of computer vision technology, especially the progress of deep learning technology, text recognition algorithms, especially single character recognition algorithms, have achieved quite good performance and meet the application requirements in different scenarios [1]. On the contrary, with the increasing popularity of practical vision systems and smart phones, text detection in natural scenes becomes a new research hotspot in recent years, though it is much more difficult because of the underlying uncertainty and variations in

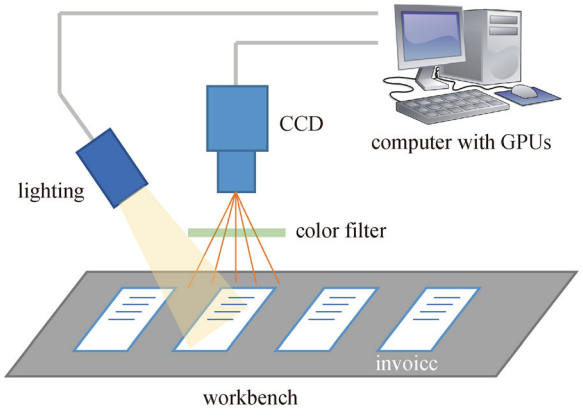

Fig. 1 A typical CCD based machine vision system for real-time text detection and recognition

Received August 28, 2018; accepted October 8, 2018

E-mail: liubinxp@163.com

natural image [2]. An example of natural scene text detection is shown in Fig. 2.

Natural scene text detection algorithms, which have been widely studied by researchers in recent years, are mainly divided into two categories, classical methods and modern methods. The classical methods implement text detection and segmentation by use of low-level image features such as color, texture, stroke, connectivity, and spectrum of the image region. Those features are usually extracted by handcrafted operators that are not robust enough to general scenarios. The key operators involved in the classical methods include MSER [3,4], SWT [5] and so on. In contrast, the modern methods directly identify the candidate image regions and determine whether they are text regions by classification confidence. Benefit from deep learning techniques, the modern methods surpass the classical methods with regard to robustness and generalization capability [6].

The modern methods typically include three steps: candidate text region generation, text region identification, and non-maximum suppression. Candidate text region generation aims to find image regions that may contain text; text region identification intends to determine whether the image region is a text block (typically a confidence score is provided); and non-maximum suppression eliminates those image regions with higher overlay but lower confidence score.

Since text detection seems to be a special case of object detection, researchers tried to apply the general object detection algorithms such as faster-RCNN [7], SSD [8], YOLO [9] directly to the text detection task. However, these object detection algorithms perform poorly on text detection. This is because text detection tasks are quite different from general object detection tasks:

1) Compared to general object, text bock usually has great variation on region width and aspect ratio;

2) Text block is directional, and the bounding box used for object annotation in general object detection is not sufficient to describe the location of text block;

3) Some patterns in natural scene images are quite similar to character appearance and they need to be distinguished by use of image global context;

4) Characters with special art fonts are difficult to recognize because they may appear as hollow, outlined or textured pattern.

Fig. 2 Natural scene text detection: find and locate the text blocks

In response to the above problems, deep learning-based text detection algorithms have emerged in recent years. They improve and optimize the general object detection algorithms from feature extraction, region proposal network (RPN), multi-task joint training, loss function improvement, non-maximum value suppression (NMS), semi-supervised learning and other aspects. As a result, the text detection accuracy in natural scene images are improved significantly. For example, connectionist text proposal network (CTPN) [10] improves the text detection accuracy by exploring and exploiting the contextual characteristics of text characters through the bidirectional long short-term memory (BiLSTM). RRPN [11] uses the bounding box along with the rotation angle as the text block annotation and training data, thereby obtaining the ability to detect the rotational text block. DMPNet [12] uses quads (non-rectangular) as text block annotation because it is more closely surround the text content. SegLink [13] cuts words into smaller blocks that are easier to detect, and then concatenates adjacent blocks into text line. TextBoxes [14,15] has the ability to detect thinner text lines by introducing rectangular convolution kernels. FTSN [16] uses mask-NMS instead of the traditional NMS algorithm to filter candidate bounding boxes. WordSup [17] uses a semi-supervised learning strategy to train character-level text detection models with word-level annotated data.

Among these text detection algorithms, CTPN [10] has been proved to be a very effective text detection algorithm in complex natural scenes. CTPN is based on the faster R-CNN object detection algorithm and incorporates LSTM network. CTPN achieves quite good generalization capability and overall performance through pre-training on the ImageNet [18] data set and retraining on the ICDAR 2013 [19] data set.

Although CTPN has achieved good detection performance, due to its complex network structure and high computational cost, it is difficult to implement real-time text detection in many resource-constrained industrial applications, such as industrial computers without GPUs. In this paper, through the principle analysis of CTPN algorithm, we improved the algorithm by hyper-parameter optimization, network model optimization and hardware acceleration, and realize a real-time text detection model. Finally, we verified the effect of the optimization methods proposed in this paper through experiments.

2 Theory and method

2.1 Principle of CTPN

CTPN is a natural scene text detection algorithm. It is one of the best text detection algorithms at present. The CTPN algorithm mainly includes four stages. The first stage uses the popular VGG16 [20] as the backbone network to

extract the features of the candidate image regions, and the second stage uses the BiLSTM network to extract the context features. The third stage performs text recognition and bounding box regression through a fully connected network, and the fourth stage performs non-maximum suppression and merges adjacent text blocks into text lines.

Since the goal of CTPN is text detection in complex natural scenes, in order to improve detection robustness, a relatively complex network model and detection flow are employed, this results in low detection speed under CPU hardware conditions. Therefore, when applying the algorithm to invoice text detection, it is necessary to balance between the detection accuracy and the detection speed of the algorithm.

2.2 Hyper-parameter optimization

CTPN is a text detection algorithm based on faster R-CNN, which contains dozens of hyperparameters. Some of them only affect the training process of the model, while others affect the speed and accuracy of the inference stage. To obtain the best detection performance, these hyperparameters need to be carefully tuned according to the application scenario. In the application scenario of invoice recognition, the configuration of the hyper-parameter is quite different due to the distinction between invoice images and general images.

Compared with general images, invoice images have some advantageous aspects:

1) The invoice images are usually obtained in a constrained environment (relatively stable illumination and scale), and the image quality is relatively high;

2) Due to the regular shape of the invoice objects, the geometric distortions can be easily corrected by image preprocessing. Therefore, the image orientation and image size are not changed much;

3) The font and color of the text in the invoice images are very limited, and the font type can be enumerated.

These advantageous aspects allow us to increase the detection speed without decreasing the detection performance through proper hyperparameter adjustment. On the other hand, the disadvantage of the invoice recognition application scenario is that the text content is usually overlapped by background patterns or the stamp patterns. Fortunately, we can cope with the trouble by preprocessing based on prior knowledge.

From the above analysis, we elaborately tuned the hyperparameters including “detection mode (horizontal or vertical),” “whether to use multi-resolution feature map,” “image resolution range,” “anchor generation parameter,” “NMS threshold parameter,” and “effective bounding box size.” For example, by the detecting mode parameter, the CTPN can be set to detect only horizontal text lines or also detect vertical text columns. One also can choose whether to allow CTPN to use multiresolution feature maps to improve the detection accuracy for text blocks with different scales. In addition, there are some trivial parameters provided for tuning the performance of CTPN. These parameters affect both running speed and detection accuracy. We have found that the image resolution and the number of anchors have a great impact on the performance of the algorithm. During model training, we maximize the reasoning speed of the model through cross-validate, while ensuring the required accuracy. As a result, the running speed of the text detection algorithm was significantly improved without reducing the detection accuracy. Related test results are shown in Section 3.2.

2.3 Network model optimization

Deep neural networks have achieved great success in many areas of computer vision. As the performance of the network model continues to increase, the complexity of the model is also increasing. To make these neural network models run on resource-constrained systems, such as industrial computers, mobile devices, etc., researchers have proposed many methods to accelerate the model. There are some advanced model acceleration technologies, such as MobileNet [21], ShuffleNet [22], SqueezeNet [23] and so on. Note that some of these models only compress the model size and do not help for model acceleration.

MobileNet is based on a streamlined architecture that uses depth-wise separable convolutions to achieve speed improvement. ShuffleNet uses pointwise group convolution and channel shuffle to greatly reduce computation cost while maintaining accuracy. SqueezeNet is mainly used for model compression, which reduces the size of the model weight parameters. Other model acceleration techniques, such as quantification and pruning, rely on the underlying implementation of the neural network framework.

We replace the original backbone network in the CTPN algorithm with an accelerated network model. For example, we replace the VGG16 network with MobileNet, and then re-training and re-evaluating the detection algorithm. In this way, we have achieved a certain speed performance improvement. See Section 3.2 for the relevant test results.

2.4 Hardware acceleration

In resource-constrained industrial scenarios, the computer running a text detection algorithm may not have a GPU graphics card configured, that means all the computations have to run on the CPU. To improve the computing capability of the CPU, manufacturers developed single instruction, multiple data (SIMD) technology. SIMD is a class of parallel computers in Flynn’s taxonomy. It describes computers with multiple processing elements that perform the same operation on multiple data points

simultaneously. SIMD is particularly applicable to common tasks such as image processing and neural network. Most modern CPU designs include SIMD instructions to improve the performance of multimedia use. By default, deep learning development frameworks, including Tensorflow, do not support SIMD acceleration for CPUs in order to maintain good compatibility. We implemented SIMD acceleration by using CPU hardware that supports SIMD and recompiled the source code of Tensorflow to achieve faster text detection algorithms. Of course, if the target computer contains a GPU device, the powerful parallel computing power of the GPU can be utilized to greatly increase the speed of the text detection algorithm. See Section 3.2 for the relevant test results.

2.5 Optical system improvement

Typically, CCD image sensors are used as image acquisition devices for machine vision systems. To get richer visual information, the color CCD camera is preferred by most applications.

However, in our system, by combining a monochrome CCD and a color filter in the optical path (as shown in Fig. 1), the system cost can be reduced and the speed performance of the system can be greatly improved. In case of invoice recognition, the images often contain random background patterns and stamp patterns that impact the performance of text detection and recognition. Since these patterns usually have a specific color, they can generally be removed by color component analysis in image preprocessing stage. However, the preprocessing module requires extra running time, and the three-channel image also requires more time for neural network inference. By simply inserting a custom color filter in the optical path, the system cost is reduced and simultaneously the neural network inference speed is almost 3 times increased because of the use of the single-channel image.

3 Experiments

3.1 Evaluation of the CTPN algorithm

We evaluated the detection performance of the CTPN algorithm under various natural scene images. From the test results, we find that CTPN not only has good detection performance for various general scene images, but also works very well on detecting text lines in the invoice images. Better performance can be achieved by combining prior knowledge of the application scenario with appropriate preprocessing algorithms. Figure 3 shows the detection results on two invoice images.

It can be seen from the test results that the CTPN algorithm can correctly detect all text lines, including some ambiguous text areas. Furthermore, the localization accuracy of the text blocks is also satisfactory. The detection results stored in form of text bounding box can be directly fed to the text recognition module for character segmentation and recognition.

To improve the robustness under natural scene images, the CTPN algorithm uses a complex network model and a multistage detection pipeline, this leads to a huge computation cost. We tested the time consumption of the CTPN algorithm in various hardware environments. The test results are shown in Fig. 4.

It can be seen that in the mainstream CPU environment, the time consumption of the CTPN algorithm is about 8 s per image. With SIMD acceleration technology, the time consumption can be shortened to 4 s. In the GPU environment, the time consumption further decreases to about 0.1 s to 0.4 s per image.

The detection pipeline of the CTPN algorithm consists

Fig. 3 Detection results of the CTPN algorithm on two invoice images

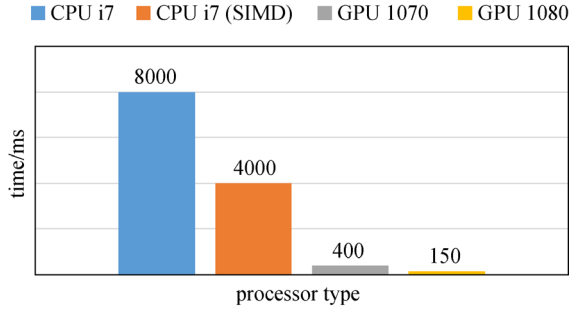

Fig. 4 Time consumption of the CTPN algorithm in various hardware environments

of 4 stages, namely CNN, LSTM, proposal, and detection. Each stage consumes quite different computing resources. To analyze and optimize the time consumption of each part of the CTPN algorithm, we tested the running time of each stage, and the test results are shown in Table 1.

Note that only the speed performance of the first stage CNN can be improved through model compression and optimization. The proposal stage consists of some trivial operations such as proposal anchor generation and text block merging. These operations are hard to be optimized by parallel computing such as SIMD and GPU, so we did not optimize it directly. In fact, the proposal stage is optimized indirectly by hyperparameter optimization since it is impacted significantly by some of the hyperparameters such as number of anchors and thresholds for block merging.

3.2 Evaluation of system acceleration

The data in Fig. 4 has proved that SIMD and GPU are of great benefit to algorithm acceleration. To evaluate the impact of the network model on the speed performance of the algorithm, we tested four network architectures, VGG16, InceptionV3, ResNet50 and MobileNet. The test results on CPU and GPU are shown in Figs. 5(a) and 5(b), respectively.

It can be found that MobileNet has a great effect on decreasing the time consumption during the training phase, while the speed improvement is negligible for the inference phase. The reason is that a small batch size of input images is not beneficial to take advantage of MobileNet. Moreover, the evidence of performance improvement presented in the original paper of MobileNet is based on the metric of floating point operations per second (FLOPS), and the actual running speed of the model depends on many factors. So the actual speed improvement in practice is very limited.

In our experiment, we used some optimization search schemes, such as grid search, gradient descent, genetic search, and so on, to find an acceptable hyperparameter set.

Based on the testing of the proposed optimization methods: hyperparameter optimization, network model optimization, hardware acceleration and optical system improvement, the results are summarized in Table 2.

Table 1 Time consumption of each stage of the CTPN algorithm

stage	time
CNN	3.4 s
LSTM	+0.2 s
proposal	+1.4 s
detection	+0.3 s
total	5.3 s

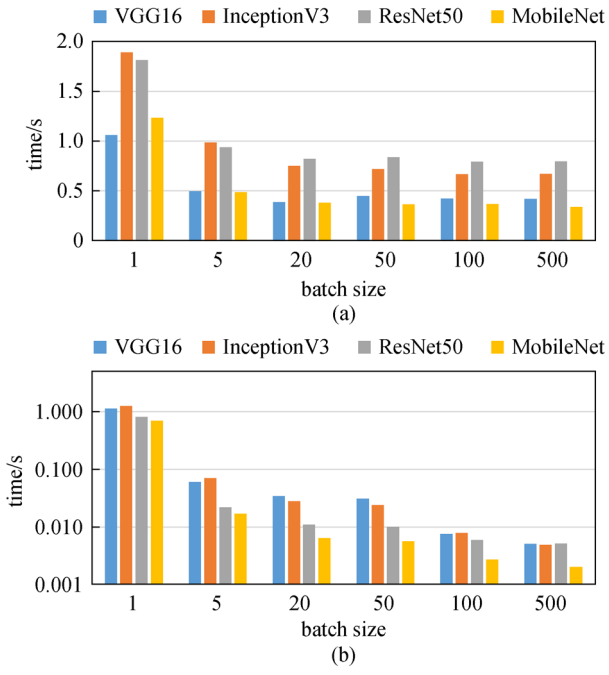

Fig. 5 Time consumption of the CTPN algorithm with various network architectures. (a) Test results on CPU (Intel i7); (b) test results on GPU (NVidia GTX1080)

Table 2 Contribution of various optimization methods to algorithm-acceleration

method	speed improvement	accuracy degradation
hyperparameter optimization	10%	yes
SIMD acceleration	50%	no
GPU acceleration	2000%	no
network model optimization	80%	yes
optical system improvement	250%	no

4 Conclusion

Text detection and recognition is a classic computer vision problem. With the development of deep learning technology, text recognition has made great progress. On the contrary, text detection is more challenging especially for complex natural scenes. In this paper, we analyzed the performance bottleneck of the CTPN algorithm, and proposed a series of optimization and acceleration methods. The experimental data shows that the proposed methods significantly improves the speed of the text detection algorithm, making the text detection algorithm suitable for real-time industrial scenarios with limited computation resources. Although the analysis and experiments are performed on CPTN, some of these optimization

strategies, such as the use of a speed-optimized network model as the backbone network architecture, are generic and suitable for other text detection models including EAST, Seglink, TextBoxes, and so on. Future works include achieving cloud deployment of the model to enable real-time text detection in industrial scenarios where computation resources are limited while a high network bandwidth is available.

References

1. Contes A, Carpenter B, Case C, Satheesh S, Suresh B, Wang T, Wu J D, Ng A Y. Text detection and character recognition in scene images with unsupervised feature learning. In: Proceedings of International Conference on Document Analysis and Recognition. Beijing: IEEE, 2011, 440–445
2. Ye Q, Doermann D. Text detection and recognition in imagery: a survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(7): 1480–1500
3. Zhang X, Gao X, Tian C. Text detection in natural scene images based on color prior guided MSER. Neurocomputing, 2018, 307: 61–71
4. Smith R. An overview of the tesseract OCR engine. In: Proceedings of International Conference on Document Analysis and Recognition. Parana: IEEE, 2007, 629–633
5. Epshtein B, Ofek E, Wexler Y. Detecting text in natural scenes with stroke width transform. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Francisco: IEEE, 2010, 2963–2970
6. Jaderberg M, Simonyan K, Vedaldi A, Zisserman A. Reading text in the wild with convolutional neural networks. International Journal of Computer Vision, 2016, 116(1): 1–20
7. Ren S, He K, Girshick R, Sun J. Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137–1149
8. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S E, Fu C, Berg A C. SSD: single shot MultiBox detector. In: Proceedings of European Conference on Computer Vision. Berlin: Springer, 2016, 21–37
9. Redmon J, Divvala S K, Girshick R B, Farhadi A. You only look once: unified, real-time object detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016, 779–788
10. Tian Z, Huang W, He T, He P, Qiao Y. Detecting text in natural image with connectionist text proposal network. In: Proceedings of European Conference on Computer Vision. Berlin: Springer, 2016, 56–72
11. Ma J, Shao W, Ye H, Wang L, Wang H, Zheng Y, Xue X. Arbitrary-oriented scene text detection via rotation proposals. IEEE Transactions on Multimedia, 2018, 20(11): 3111–3122
12. Liu Y, Jin L. Deep matching prior network: toward tighter multi-oriented text detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017, 3454–3461
13. Shi B, Bai X, Belongie S. Detecting oriented text in natural images by linking segments. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017: 3482–3490
14. Liao M, Shi B, Bai X, Wang X, Liu W. TextBoxes: A fast text detector with a single deep neural network. 2016, arXiv:1611.06779
15. Liao M, Shi B, Bai X. TextBoxes ++: a single-shot oriented scene text detector. IEEE Transactions on Image Processing, 2018, 27(8): 3676–3690
16. Dai Y, Huang Z, Gao Y, Xu Y, Chen K, Guo J, Qiu W. Fused text segmentation networks for multi-oriented scene text detection. 2018, arXiv:1709.03272
17. Hu H, Zhang C, Luo Y, Wang Y, Han J, Ding E. WordSup: exploiting word annotations for character based text detection. In: Proceedings of IEEE International Conference on Computer Vision. Venice: IEEE, 2017, 4950–4959
18. Deng J, Dong W, Socher R, Li L J, Li K, Li F F. ImageNet: a large-scale hierarchical image database. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Miami: IEEE, 2009, 248–255
19. Karatzas D, Shafait F, Uchida S, Iwamura M, Bigorda L G I, Mestre S R, Mas J, Mota D F, Almazàn J A, Heras L P D L. ICDAR 2013 robust reading competition. In: Proceedings of International Conference on Document Analysis and Recognition. Washington, DC: IEEE, 2013, 1484–1493
20. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. 2014, arXiv:1409.1556
21. Howard A G, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, Andreetto M, Adam H. MobileNets: efficient convolutional neural networks for mobile vision applications. 2017, arXiv:1704.04861
22. Zhang X, Zhou X, Lin M, Sun J. ShuffleNet: an extremely efficient convolutional neural network for mobile devices. 2017, arXiv:1707.01083v2
23. Iandola F N, Han S, Moskewicz M W, Ashraf K, Dally W J, Keutzer K. SqueezeNet: AlexNet-level accuracy with 50 × fewer parameters and < 0.5 MB model size. 2016, arXiv:1602.07360v4

Shihua Zhao graduated from Chongqing University and obtained the Ph.D. degree in 2013. He is now working in State Grid Hunan Electric Power Corporation Limited Research Institute. His main research interests include high voltage technology, power transformer fault detection and diagnosis. He is the author or the co-author of several technical papers.

Lipeng Sun completed the B.S. degree in 2005 from Chongqing University, Chongqing, China. He also received his master degree in 2008 from Chongqing University, Chongqing, China. His current research interest is high voltage and insulation technology.

Gang Li completed the B.S. degree in 1996 from Xi'an Jiaotong University, Xi'an, China. He has been engaged in operation and maintenance technology of transformer over 20 years. His current research interest is high voltage and insulation technology.

Yun Liu received his master degree in 2009 from Guangxi University, Nanning, China. He received deputy senior engineer in 2013. His research interest is high voltage and insulation technology.

Binbing Liu received the bachelor degree in Optoelectronics from Huazhong University of Science and Technology, Wuhan, in 2000, and got the master degree and Ph.D. degree in Physical Electronics from the same university, in 2003 and 2013, respectively. Now, he is working at School of Optical and Electronics Information as a lecturer. His research interests include computer vision and machine learning.