Multi-type web image text detection based on the improved EAST algorithm

Yuanji Xiang¹, Fei Luo²
¹ School of Software Engineering, Chengdu University of Information Engineering, Chengdu, Sichuan, 610225, China
² School of Software Engineering, Chengdu University of Information Engineering, Chengdu, Sichuan, 610225, China
xiangyuanji@foxmail.com

Abstract. Web image character recognition has important application value in the commercial field and text detection is the basis of extracting this text information. Among many text detection algorithms, EAST (efficient and accurate scene text detector) algorithm has a good performance in natural scenes, but it is ordinary in web pictures. Based on this, aiming at the problem that the number of channels in the QUAD output layer of EAST algorithm is redundant, which leads to the low accuracy of web image text detection, it is modified as the distance from the point marked as a positive sample to the four vertices of the text box where it is located instead of the coordinate offset to the vertices. The improvement reduces from the original 8 channels to 4 channels, which restricts the optimization direction of the model, and then by setting a new loss function in order to improve the loss value of the edge area of the text box and the pixels in the image that are difficult to detect, the model is more suitable for the text detection of the web image. The experimental results show that the improved EAST algorithm significantly improves the accuracy of text detection in web images.

1. Introduction
In the Internet world, picture is an important medium to transmit information. Especially in e-commerce, social networking, search and other fields, hundreds of millions of images are spreading every day. OCR has an important application value in the business field. It is the basis of data informatization and online and offline communication, and the research hotspot in the academic field. However, there is little research on the algorithm of OCR data set based on web pictures and mainly in Chinese. Based on this, this paper will study the Chinese-English mixed data set composed of web pictures named MTWI (Multi-type web image). As shown in the figure 1, the data set is full of data, covering dozens of fonts, several to hundreds of pixel font sizes, multiple formats, and more interference background.
Text detection is the most important premise of text recognition[1]. The core of text detection is the design of features to differentiate text and background. Traditionally, the feature is obtained by artificial design, while in the method based on deep learning, the effective features are directly learned from the training data. In the text detection based on deep learning, EAST[2] algorithm is one of many good performance text detection algorithms[3,4,5]. EAST contains two stages, the first stage is a fully convolutional network (FCN) model that directly produces word or text-line level predictions, and the second stage is a Non-Maximum Suppression that can yield results from previous stage's prediction. in this paper, we mainly improve the QUAD output layer of the first stage in EAST algorithm and its loss function. Experimental results show that the improved EAST algorithm significantly improves the detection accuracy on MTWI data sets.

2. Related Work
Scene text detection and recognition have always been a research hotspot in the field of computer vision, during which there are many enlightening ideas and effective methods[6,7,8]. This section will concentrate on the work related to these algorithms.

The traditional method depends on the characteristics of manual design. Epshtein et al.[9] proposed stroke width transformation (SWT), the text in the background image is extracted by transforming the original image into a stroke width map and combining with geometric reasoning to restore the form of the text. Neumann et al.[10] proposed Maximally Stable Extremal Regions (MSER), after extracting the MSER region from the image, the algorithm uses morphometric operations and geometric rules to detect the region where the text is located. Zhao yu[11] proposed an MSER based on morphological filtering, and the experiment proved that the improved stroke width transform algorithm could maintain the integrity of character region well.

With the development of neural network[12], the task of text detection has entered a new era based on deep neural network algorithm. Yao et al.[13] proposed to use the image for text detection, applied FCN to text detection, and obtained better detection results. Tian et al.[14] proposed CTPN and constructed CNN-RNN (Recurrent Neural Network) joint model through vertical candidate box mechanism to detect text lines. It has good effect for horizontal text detection, but poor effect for multi-directional text detection. Zhou et al.[15] proposed an efficient and accurate scene text detector (East), which is a full convolution neural network based on U-net mechanism, NMS is the post-processing part. By directly detecting the text or text lines in the image, they give up the unnecessary intermediate process, and can detect the text lines in different scales and directions, which is significantly better than the previous methods in performance and speed.

In this paper, we mainly improve the output part of the neural network and the corresponding loss function based on the East model, so that the improved algorithm can improve the accuracy of text detection in the web picture.

3. Methodology
The core of EAST algorithm is a neural network model which directly predicts the existence and
geometric characteristics of text instances through training. As shown in figure 2, the structure a full convolutional neural network suitable for text detection, which outputs dense per-pixel predictions of words or text lines.

**Figure 2. Neural network structure of EAST.**

In the feature extraction phase, the convolution network pre-trained on the ImageNet[16] dataset is used, which has the intersecting convolution layer and pooling layer. The image first passes through a convolution layer with a convolution kernel size of 7×7, and then passes through four convolution stages in turn. The feature map generated in each stage is named as $f_4$, $f_3$, $f_2$ and $f_1$ respectively, and the size is $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$ and $\frac{1}{32}$ of the original image.

$$
g(i) = \begin{cases} 
\text{unpool}(f_1), & i = 1 \\
\text{unpool}(h_{i-1}), & i = 2, 3, 4 
\end{cases}$$

$$
h_i = \text{conv}_3\times3(\text{conv}_1\times1[g_i; f_{i+1}]) \quad i = 1, 2, 3$$  

(1)

(2)

In the feature merging stage, the idea of U-Net[17] is adopted. The last four feature maps produced in the previous stage are merged according to the formula step by step. The formula is as follows: Where $g_i$ is the feature map before merging, and $h_i$ is the feature map after merging, and $f_i$ corresponds to the extracted feature map. The operator $[; ; ]$ represents connecting the feature map along the channel axis, and conv3×3 represents sending the feature map into the convolution layer with the convolution kernel size of 3×3 for convolution operation, and conv1×1 represents sending the feature map into the convolution layer with the convolution kernel size of 1×1 for convolution operation.

In the output layer, it includes the single channel text score and the geometric score of QUAD of 8 channels. The text score is the confidence probability that each pixel belongs to the text. For QUAD, East uses eight numbers to represent the coordinate offset $(\Delta x_i, \Delta y_i)$ from each point marked positive to the four vertices $\{p_i | i \in \{1, 2, 3, 4\}\}$ of its text box.

### 3.1. Improved output

As we discussed above, the accuracy of East algorithm in network image detection is not high, most of which is due to the particularity of web image and the output layer of network. Compared with the natural scene, the web image has more text, more complex background and more different text size. At
the same time, the network output layer QUAD contains too many channels, which increases the computation and limits the model's ability of web image detection. Here, we change the offset of the point to vertex position of the geometric score output to the point-to-point distance calculation and reduce the number of channels by two times. In terms of calculation amount, if the input image size is $256 \times 256$, one image will be calculated $256 \times 256 \times 4$ times less, which will save a lot of time in both model training and actual detection. At the same time, the experimental results show that the detection accuracy of the model is significantly improved by using the improved output layer.

Here, we change the offset of the point to vertex position of the geometric score output to the point-to-point distance calculation and reduce the number of channels by two times. In terms of calculation amount, if the input image size is $256 \times 256$, one image will be calculated $256 \times 256 \times 4$ times less, which will save a lot of time in both model training and actual detection. At the same time, the experimental results show that the detection accuracy of the model is significantly improved by using the improved output layer.

Here, we change the offset of the point to vertex position of the geometric score output to the point-to-point distance calculation and reduce the number of channels by two times. In terms of calculation amount, if the input image size is $256 \times 256$, one image will be calculated $256 \times 256 \times 4$ times less, which will save a lot of time in both model training and actual detection. At the same time, the experimental results show that the detection accuracy of the model is significantly improved by using the improved output layer.

### Table 1. Output comparison before and after modification.

| Output channels | Total |
|-----------------|-------|
| Before $(\Delta x_1, \Delta y_1), (\Delta x_2, \Delta y_2), (\Delta x_3, \Delta y_3), (\Delta x_4, \Delta y_4)$ | 8     |
| After $d_1, d_2, d_3, d_4$ | 4     |

### 3.2. Improved loss function

The output of East includes score map and geometry score, so its corresponding loss function can be described as

$$L = L_s + \lambda L_g$$

where $L_s$ and $L_g$ represent score map and geometry score respectively, and $\lambda$ measures the importance between two losses, and its default value is 1. In $L_s$, $\hat{y}$ and $y$ mean forecast sore and ground score. The specific formula of $L_g$ is as follows, which we will modify. In the following formula,

$$L_g = \min_{\tilde{Q} \in Q^*} \sum_{c_i \in C_Q, \tilde{c}_i \in C_{\tilde{Q}}} \frac{smoothed_{L1}(c_i - \tilde{c}_i)}{8 \times N_{Q^*}}$$

EAST uses smoothed-L1 loss[18]. $C_Q$ is an ordered set of all coordinate values, $C_{\tilde{Q}}$ is an ordered set of four vertex coordinate values, and $N_{Q^*}$ is the shortest edge of each text box.

We modified $L_g$ to adapt to the new output, the modified $L_g$ can be described as

$$L_g = \text{smoothed}_{L1} \sqrt{\frac{\sum_i^4 (d_i - \tilde{d}_i)^2}{4}}$$

where $d_i$ and $\tilde{d}_i$ represent the real distance and predicted distance from the point to the ith vertex respectively.

### 4. Experiment

#### 4.1. Dataset

The MTWI dataset contains 10000 web pictures with different resolutions. We will use 6000 pictures for training, 2000 for validation and 2000 for testing. Before the training. The way we deal with pictures including resizing to uniform size and generating 5-channel output labels. In addition, we also rank the pictures randomly.

#### 4.2. Training

Experimental hardware environment: 64-bit Intel i5-9400f @ 2.90ghz × 6 CPU, 15.6g ram, NVIDIA GeForce GTX 1660 6GB video card, and Ubuntu 16.4 operating system.

The model is trained end-to-end using AdamW optimizer. For better training, input image will be resized to $512 \times 512 \times 3$, the learning rate is initialized to $1e^{-3}$, decays to 0.94 every 5000 minibatches,
and stops at 1e-5. The model is trained until validation performance stops improving.

4.3. Evaluating indicator

Generally, in the field of text detection, there are three evaluation indicators, recall, precision and Fscore. The formula can be written as

\[
Recall = \frac{match(G,D)}{|G|}
\]

\[
Precision = \frac{match(G,D)}{|D|}
\]

\[
Fscore = 2 \times \frac{Recall \times Precision}{Recall + Precision}
\]

where \( G \) is the real text box set, \( D \) is the predicted text box set, \( match \) is used to calculate the number of matched text boxes in the two sets, that is, to detect the exact number of text boxes. Fscore is a comprehensive evaluation index of recall and precision, which is used to comprehensively reflect the overall results.

4.4. Results

On the MTWI dataset, we experiment the improved EAST and EAST algorithm, and compare the results. As shown in the figure 3, we selected some experimental results for comparison. It can be seen from the figure that there are some problems in the detection of East algorithm in network image, such as missing detection and inaccurate detection and positioning. The improved algorithm, in the number of detection, and accuracy have been significantly improved.

The evaluation results of East, improved East and CTPN on the MTWI dataset are listed in the table 2. It can be seen from the table that the recall, precision and Fscore obtained by improved East are superior to the others.
Table 2. Evaluation results of several algorithms.

| Algorithm   | Recall | Precision | Fscore |
|-------------|--------|-----------|--------|
| CTPN        | 0.5352 | 0.5664    | 0.5504 |
| EAST        | 0.6088 | 0.6561    | 0.6316 |
| Improved EAST | 0.7562 | 0.8052    | 0.7793 |

5. Conclusion and Future Work
This paper proposes an improved East algorithm, which is more suitable for network image text detection by rewriting the output layer and optimizing the loss function. Experimental results show that the improved East algorithm improves the detection accuracy and speed. In the future work, we can improve the East network more and improve its accuracy of various types of text detection.

References
[1] Tian, Z., Huang, W., He, T., He, P., & Qiao, Y. (2016) Detecting text in natural image with connectionist text proposal network. In: European Conference on Computer Vision. Paris. pp. 56-72.
[2] Zhou, X., Yao, C., Wen, H., et al. (2017) EAST: An Efficient and Accurate Scene Text Detector. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). New York. pp.2642-2651.
[3] Buta, M., Neumann, L., Matas, J. (2015) FASText: Efficient Unconstrained Scene Text Detector. In: 2015 IEEE International Conference on Computer Vision (ICCV). London. pp.66-71.
[4] Tian, S., Pan, Y., Huang, C., Lu, S., Yu, K., Lim, C. (2015) Text Flow: A Unified Text Detection System in Natural Scene Images. ICCV, 528: 34-39.
[5] Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A. (2016) Reading Text in the Wild with Convolutional Neural Networks. International Journal of Computer Vision,116: 1-20.
[6] Zhu, Y., Yao, C., Bai, X. (2016) Scene text detection and recognition: recent advances and future trends. Frontiers of Computer Science, 10: 19-36.
[7] Uchida, S. (2014) Text Localization and Recognition in Images and Video. Handbook of Document Image Processing and Recognition. 5: 843-883.
[8] Ye, Q., Doerrmann, D.(2015) Text Detection and Recognition in Imagery: A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 37: 1480-1500.
[9] Epshtein, B., Ofek, E., Wexler, Y. (2010) Detecting Text in Natural Scenes with Stroke Width Transform. In: IEEE Conference on Computer Vision & Pattern Recognition. New York. pp. 2963-2970
[10] Lukas N., Jiri M.(2011) A Method for Text Localization and Recognition in Real-World Images. Lecture Notes in Computer Science, 6494(1): 770-783.
[11] Zhao, Y. (2016) Research on text detection technology in natural scene. Journal of University of Electronic Science and technology, 3: 12-20.
[12] Xie, L.J., Ji, G.S., Peng, Q. (2018) Application of improved convolution neural network in pedestrian detection. Computer science and exploration. 12(5): 708-718.
[13] Yao, C., Bai, X., Sang, N. (2016) Scene Text Detection via Holistic, Multi-Channel Prediction. https://arxiv.org/abs/1606.09002.
[14] Tian, Z., Huang, W., He, T. (2016) Detecting Text in Natural Image with Connectionist Text Proposal Network. In: Proceedings of the 2016 European Conference on Computer Vision. Paris. pp. 56-72.
[15] Long, J., Shelhamer, E., Darrell, T. (2015) Fully convolutional networks for semantic segmentation. In: Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition. London. pp. 3431-3440.
[16] Jia, D., Wei, D, Richard, S., Li, J. L., Li, F.F. (2009). ImageNet: a Large-Scale Hierarchical Image Database. In: 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Miami. pp. 20-25.
[17] Karatzas, D., Shafait, F., Uchida, S. (2013) ICDAR 2013 Robust Reading Competition. In: Proceedings of the 12th International Conference on Document Analysis and Recognition. Washington, pp. 1484-1493.

[18] Girshick, R. (2015) Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision. 312: 1440–1448.