Football Match Scene Text Detection Based on Convolutional Neural Network

Yang Li¹, Shaobin Li¹, Yunchao Xie², Lei Chen¹ and Shuang Feng¹

¹Communication University of China, Beijing, China
²China Art Science and Technology Institute, Beijing, China
E-mail: ly654979495@163.com

Abstract. With the development of deep learning, text detection based on neural network has gained more in-depth research and more extensive application. Considering the characteristics of the text in the football match scene, a novel neural network architecture is proposed based on the TextBoxes. The proposed algorithm performs well in the task of text detection in the football maes which fit the text boxes in football match scene. In order to solve the sample imbalance problem that interference model optimization, we propose Focal Loss as a loss function for classification. Finally, the Non-maximal suppression is applied to eliminate redundant bounding boxes and aggregate outputs. We make a dataset for training, and verify the effectiveness of the algorithm.

1. Introduction

Text detection is one of the most challenging research topics in the field of computer vision in recent years. Text as a special kind of visual information, in addition to the basic computer vision features such as color and texture, it also has clear and targeted semantic information, which plays a key role in image and scene understanding.

Football is one of the most popular sports in the world. Audiences, team coaches, fans, etc. all have the need for intelligent analysis of football matches. In recent years, the demand for automatic analysis tools for football matches has greatly increased. In the complex scenes of football matches, more accurate text detection of football matches is provided, which provides a basis for understanding the information in scenes and analyzing football matches.

Text detection in scene images is the first step in scene text analysis. There are many different methods of text detection. Previous method for text detection can be roughly categorized into two categories: one is a method based on connected component analysis, and the other is sliding window-based. Yao [1], Epshtein [2], Neumann [3] and others used connected component analysis to detect texts. This type of method [1,2,3,4,5] first analyzes connected component based on similarity characteristics such as color similarity or spatial adjacency. Then the connected area is classified as the text area or the non-text area, so that the text area is distinguished from the whole image. Kim [6], Gllavata [7], Lyu [8] et al. used the sliding window-based method for text detection, and the sliding window-based method [6,7,8,9,10] mainly used variable-size sliding windows to sampling on multiple spatial scales, and then use machine learning methods to determine whether there is text in the window.

Based on the TextBoxes [11] algorithm, this paper proposes a new convolutional neural network, which can effectively detect texts in the images of football matches scene. On the one hand, for the football matches scene, in where the shape of the text is various and the aspect ratio of the jersey
number and the billboard is different, the default boxes ratio adapted to the text detection in the football scene can be set. On the other hand, for the football matches, the background of the image is far more than the text which results the sample imbalance problem. We propose the use of Focal Loss as a loss function for classification to solve this problem. And the dataset is labeled for text detection in the football match scene. Validation of the algorithm is performed on the dataset and verifies the effectiveness of the algorithm.

2. Another section of your paper

2.1. Text detection network structure

The architecture of text detection is shown in figure 1. It is a 28-layers convolutional network. The former 13 layers inherit the VGG-16 architecture [12], and keeping the layers from conv1_1 through conv4_3. The last 15 layers comprise 13 convolutional layers and 2 pooling layers. The size of the convolutional layer is gradually reduced and objects with different scales can be detected. The feature maps of the 6 convolutional layers are directly connected to the text-box layers. The output of the text-box layer is the classification score and offsets to its associated default boxes. Non-maximum suppression (NMS) is applied to aggregated outputs of text-box layers. The configuration parameters of the network are shown in table 1.

| Type   | Configurations                  |
|--------|---------------------------------|
| Conv1_1| Maps: 64, kernel: 3×3, stride: 1, pad: 1 |
| Conv1_2| Maps: 64, kernel: 3×3, stride: 1, pad: 1 |
| Pooling1| Max, kernel: 2×2, stride: 2 |
| Conv2_1| Maps: 128, kernel: 3×3, stride: 1, pad: 1 |
| Conv2_2| Maps: 128, kernel: 3×3, stride: 1, pad: 1 |
| Pooling2| Max, kernel: 2×2, stride: 2 |
| Conv3_1| Maps: 256, kernel: 3×3, stride: 1, pad: 1 |
| Conv3_2| Maps: 256, kernel: 3×3, stride: 1, pad: 1 |
| Conv3_3| Maps: 256, kernel: 3×3, stride: 1, pad: 1 |
| Pooling3| Max, kernel: 2×2, stride: 2 |
| Conv4_1| Maps: 512, kernel: 3×3, stride: 1, pad: 1 |
| Conv4_2| Maps: 512, kernel: 3×3, stride: 1, pad: 1 |
| Conv4_3| Maps: 512, kernel: 3×3, stride: 1, pad: 1 |
| Pooling4| Max, kernel: 2×2, stride: 2 |
| Conv5_1| Maps: 512, kernel: 3×3, stride: 1, pad: 1 |
| Conv5_2| Maps: 512, kernel: 3×3, stride: 1, pad: 1 |
| Conv5_3| Maps: 512, kernel: 3×3, stride: 1, pad: 1 |
| Pooling5| Max, kernel: 3×3, stride: 1, pad: 1 |
| Conv6 | Maps: 1024, kernel: 3×3, stride: 1, pad: 6 |
| Conv7 | Maps: 1024, kernel: 1×1, stride: 1 |
| Conv8_1| Maps: 256, kernel: 1×1, stride: 1, pad: 0 |
| Conv8_2| Maps: 512, kernel: 3×3, stride: 2, pad: 1 |
Some researchers scale the images to different sizes and merge the results at final in order to detect texts of different sizes. However, by utilizing feature maps from different layers in the network for detection we can mimic the same effect. The smaller the receptive field corresponding to the lower layer, the lower layer can capture more fine details of the image; the larger the receptive field corresponding to the higher layer, the more global information can be captured. In this paper, we use the characteristics of different layers in the convolutional network to extract different features, which can achieve the purpose of detection at multiple scales. Therefore, the detection network uses six different feature maps to achieve multi-scale detection.

2.2. Default boxes

In the football match scene, the text mainly includes the jersey number, the billboard in the scene, the player's name, and the TV station logo. Different from the text in an ordinary scene, the text categories and geometric shapes are various in the football match, such as jersey number and English word. The number shape is very different from the word shape. As shown in figure 2, the length of jersey numbers are mostly larger than width, and the width of player names or words in the advertisement are mostly larger than length.

![Figure 2. Sketch of text in football match scene.](image)

According to the text characteristic, we select the default boxes which fit the football match scene for text detection. Select the default boxes with the aspect ratio of \{1:5, 1:3, 1:2\} for numbers and select the aspect ratio of \{1:1, 2:1, 3:1, 5:1\} for word texts, thus defines the default boxes for the 7 ratios of \{1:5,1:3,1:2,1:1,2:1,3:1,5:1\}. Moreover, in text-box layers we adopt 3×3 convolution filters instead of the 1×5 convolution filters in the original model, and the 3×3 convolution filter was suitable for default boxes with a relatively large aspect ratio and a small aspect ratio.

The channels of the convolution kernels for text-box layers is changed according to the change of the default boxes. Per feature maps location generates a set of default boxes. At every map location of the six feature maps (Conv4_3, Conv7, Conv8_2, Conv9_2, Conv10_2, and global 6), it outputs the classification scores and offsets to its associated default boxes., the corresponding classification scores and offsets to its associated default boxes can be predicted. Specifically, for each box out of \(k\) at a given location, we compute 2 class scores and 4 offsets, then (2+4)×\(k\) outputs will be generated. For the \(m\times n\) feature map, the channels should be set to (2+4)×\(k\) to yield (2+4)×\(k\times m\times n\) outputs.

2.3. Loss function

The training process consists of 2 tasks: classification and regression. The classification task is used to determine is there a text in the default box, and the regression task is used to predict the offsets. The loss function is a sum of the confidence loss and the localization loss, which is defined as:

\[
L(p, l, v', v) = L_{\text{conf}}(p, l) + \alpha L_{\text{loc}}(v', v)
\]  

(1)
Where \( l \) is the label to represent category, \( l=1 \) indicates text, \( l=0 \) indicates background; \( p \) is the probability of each category; \( v \) denotes the predicted box; \( v^* \) denotes the ground truth box. \( \alpha \) is the trade-off value of the two loss functions.

The localization loss \( L_{loc} \) is the Smooth L1 loss [13], which is defined as:
\[
L_{loc}(v^*, v) = \sum_{i \in (x, y, h, w)} smooth_{\lambda}(v_i^* - v_i)
\]
(2)
\[
smooth_{\lambda}(x) = \begin{cases} 
0.5x^2 & \text{if } |x| < 1 \\
|x| - 0.5 & \text{otherwise}
\end{cases}
\]
(3)

The parameters \( v \) and \( v^* \) are calculated as follows:
\[
v_x = \frac{x - x_a}{w_a}, \quad v_y = \frac{y - y_a}{h_a}
\]
\[
v_h = \log \frac{h}{h_a}, \quad v_w = \log \frac{w}{w_a}
\]
(4)
\[
v_x^* = \frac{x^* - x_a}{w_a}, \quad v_y^* = \frac{y^* - y_a}{h_a}
\]
\[
v_h^* = \log \frac{h^*}{h_a}, \quad v_w^* = \log \frac{w^*}{w_a}
\]
(5)

Where \( x, y, h, w \) are the horizontal coordinate, the vertical coordinate, the height and the width of the predicted box; \( x_a, y_a, h_a, w_a \) are for the default box, \( x^*, y^*, h^*, w^* \) are for the ground truth box.

The confidence loss \( L_{conf} \) is the Focal Loss [14], which is defined as:
\[
L_{conf}(p, l) = FL(p)
\]
(6)
\[
FL(p) = -\alpha(1 - p)^\gamma \log(p)
\]
(7)

Where \( \alpha \) is a weighting factor; \( \gamma \) is a focusing parameter.

Focal loss is used to make the model more focused on hard-to-train samples. It can reduce the loss contribution from easy examples and in turn increase the importance of hard examples. From equation (6), we can see that when an example is misclassified and the probability is small, the \((1-p)\) factor will be close to 1 and its loss is unaffected. When an example is well-classified, its probability is very close to 1, the factor \((1-p)\) goes to 0 and the loss is down-weighted. The \( \gamma \)-parameters adjust the rate of easy example to lower weights.

2.4. Non-maximal suppression

Non-maximum suppression is applied to eliminate the redundant predicted boxes. The best predicted box can be found after non-maximum suppression process.

3. Experiment

3.1. Datasets

(1) Dataset of the football match scene: the text in images of football match scene is labelled. The images are captured in 20 football match videos. Annotate dataset manually and generate corresponding label files. And the English word, number, symbol ":" and symbol "," are labelled. The label files contain the position information and the content of the text. 3,000 samples were obtained, including 34,512 text. The 2300 samples were used to fine tune the network and the remaining 700 samples were used to test.
(2) SynthText [15]: SynthText is a public dataset that is synthesized from real natural scene images and text instances. This dataset contains 858,750 synthesized text images, which consist of 11,698 background images and 7,268,866 words. SynthText is used to pre-train the network.

3.2. Experimental details
The input is 300×300, RGB images. In the training process, the VGG-16 model (the parameters of Conv1_1 to Conv4_3 in Table 1) are loaded to initialize the network. Networks are trained with stochastic gradient descent (SGD). The momentum is set to 0.9, and the weight decay is set to 0.0005. The learning rate is initially set to 0.001, and updates by exponential decay. All the experiments are carried out on a PC with one NVIDIA GTX1080 GPU. We implement the network within the TensorFlow framework with Python.

3.3. Results and Analysis
We train and test the basic network TextBoxes, the improved network OursDefault that resets the default boxes, and OursDefault+Focal which uses Focal Loss as a loss function to improve the network based on OursDefault.

Test data is 700 samples from dataset of the football match scene, which is completely different with the training data. There are three indicators for evaluation: precision P, recall R, and F-measure.

Table 2. Text detection on Dataset of the football match scene. P, R and F refer to precision, recall and F-measure respectively. Comparison among three methods.

| Method       | TextBoxes | OursDefault | OursDefault+Focal |
|--------------|-----------|-------------|-------------------|
| P (All text) | 98.83%    | 98.37%      | 97.53%            |
| F (All text) | 84.33%    | 89.01%      | 89.36%            |
| R (All text) |           |             |                   |
| Number text  | 73.54%    | 81.27%      | 82.45%            |
| Word text    | 84.76%    | 86.78%      | 88.21%            |

The results are summarized and compared in table 2. We observe that the proposed model in this paper has higher recall and F-measure. First of all, compared OursDefault and the basic model TextBoxes, the recall of OursDefault has been greatly improved. This shows that OursDefault proposed in this paper can detect more real texts correctly. In particular, for the recall of number text, OursDefault model compared with the TextBoxes model has improved by nearly 10%, indicating that the algorithm proposed in this paper can effectively detect number text. Secondly, compared with the first two models, the recall of all texts, numbers, and word texts in the OursDefault+Focal model is improved, and the F-measure is the highest.

Figure 3 shows the detection results of the TextBoxes and the OursDefault. The TextBoxes algorithm can successfully detect horizontal text. However, a text with a small aspect ratio is difficult to detect, such as the jersey number “7” and “16” in figure 3. After OursDefault resetting the default boxes, it can effectively detect these vertical texts with small aspect ratio. The reason is that the aspect ratio of the default boxes in the original TextBoxes is bigger than 1; OursDefault model contains default boxes with a width to height ratio of less than 1. These default boxes can better fit the jersey number which is mostly vertical.

Figure 4 shows the detection result of the OursDefault and the OursDefault+Focal. Compared with the Ours Default model, the OursDefault+Focal model can detect text more successfully in the case of occlusion, incompleteness, and blurring. The reason is that there are far more background areas in the football match scene than the text area. Most of the default boxes are negatives, which causes a significant imbalance between the positive and negative examples. We propose to use Focal Loss as a classification loss function for training. This loss function adjusts weights to different samples in the training process. The weights of well-classified examples are smaller, and the weights of difficult-to-detect examples are larger, which can effectively solve the problems caused by unbalanced samples.
For indistinguishable samples such as occlusion, incompleteness, and blurring texts, weights are given during training, therefore, these kind of texts can be effectively detected.

4. Conclusions
In this paper, we have researched text detection in the football match scenes used convolutional neural network. We use CNN to detect the texts in the images of football match scene. Based on the TextBoxes network, we proposes two improvements: First, we reset the default boxes to solve the problem that difficult to detect vertical texts such as jersey numbers. Second, we use Focal Loss as a classification loss function to solve the problem of imbalance between positive and negative samples. In addition, the dataset of football match scenes is produced for the training and testing. Experiments show that the improved method proposed in this paper is effective. In further study, we consider to use the angle information in position regression tasks to achieve arbitrary-oriented text detection.

References
[1] Yao C, Bai X, Liu W, Ma Y and Tu Z 2012 Detecting texts of arbitrary orientations in natural images IEEE Conf. on Computer Vision & Pattern Recognition 157 pp 1083-90
[2] Epshtein B, Ofek E and Wexler Y 2010 Detecting text in natural scenes with stroke width transform Computer Vision and Pattern Recognition 119 pp 2963-70
[3] Neumann L and Matas J 2010 A Method for Text Localization and Recognition in Real-World Images Proc. of the 10th Asian Conf. on Computer Vision 6494 pp 770–783
[4] Chen H., Tsai S S, Schroth G, Chen D M., Grzeszczuk R and Girod B 2011 Robust text detection in natural images with edge-enhanced Maximally Stable Extremal Regions IEEE Int. Conf. on Image Processing pp 2609-2612
[5] Jain A K and Yu B 2002 Automatic text location in images and video frames Int. Conf. on Pattern Recognition 2 p 1497
[6] Kim K I, Jung K and Jin H K 2003 Texture-based approach for text detection in images using
support vector machines and continuously adaptive mean shift algorithm *IEEE Trans. pattern Anal. mach. intell* **25** 1631-1639.

[7] Gllavata J, Ewerth R and Freisleben B 2004 Text detection in images based on unsupervised classification of high-frequency wavelet coefficients *Proc. of Int. Conf. on Pattern Recognition* **1** pp 425-428.

[8] Lyu M R, Song J and Cai M 2005 A comprehensive method for multilingual video text detection, localization, and extraction *IEEE Press* **15** pp 243-255

[9] Hanif S M. and Prevost L 2009 Text Detection and Localization in Complex Scene Images using Constrained AdaBoost Algorithm *Int. Conf. on Document Analysis and Recognition* pp 1-5

[10] Wang T, Wu D J, Coates A and Ng A Y 2013 End-to-end text recognition with convolutional neural networks *Int. Conf. on Pattern Recognition* pp 3304-3308

[11] Liao M, Shi B, Bai X, Wang X and Liu W 2016 Textboxes: a fast text detector with a single deep neural network

[12] Simonyan K and Zisserman A 2014 Very Deep Convolutional Networks for Large-Scale Image Recognition *Computer Science*

[13] Ren S, He K, Girshick R and Sun J 2015 Faster R-CNN: towards real-time object detection with region proposal networks. *Int. Conf. on Neural Information Processing Systems* **39** pp 91-99

[14] TsungYi Lin, Goyal P, Girshick R, He K and Dollar P 2017 Focal loss for dense object detection *International Conference on Computer Vision* pp 2999-3007

[15] Gupta A, Vedaldi A and Zisserman A 2016 Synthetic data for text localisation in natural images *Computer Vision and Pattern Recognition* pp 2315-2324.