Scene Text Detection

Sara Elbesomy 201601849

***Abstract*— Text is the first universal source of information in the world and it is the first way to exchange it through generations. Therefore, it has a big influence on the human civilization. Because of all of this, text detection and recognition in natural scenes is a hot research topic in computer vision. The community does much effort to face many challenges in this field such noise, distortion, blur and many other problems. One of the most robust solutions that does a huge progress in computer vision is deep learning. By using deep learning and neural networks, one can get a range of accuracies in solving hard problems that he did not get using the classical methods of computer vision. The purpose of this paper is to use a very well trained model called "CTPN" for scene text detection in a challenging dataset and try to get acceptable results identifying the failed cases and making recommendation for improving the model.**

Keywords—text detection, CTPN, computer vision, deep learning)

I. Literature Review

In the past decades, scientists have proposed many methods for text detection in natural images. There are mainly three types of methods: texture based methods, component based methods and hybrid methods.

Texture based methods

It focuses on the texture properties of the image such as local intensities, filter responses and wavelet coefficients, to differentiate between text and non text regions in the images. This kind of localization is very expensive from computations perspective because all the regions of the image are scanned. Moreover, this method has a major disadvantage which is that it provide good results on horizontal texts mainly and it does badly in case of rotation and scale change.

In an early work, Zhong et al.presented a basic method for text localization in images. The localization is done using horizontal spatial variance. Then classification of the regions if they are text or not is determined using color segmentation. Kim et al. produced a SVM that do the classification on the pixel using the raw pixel intensity to be as the local feature. This method provide a good results on both images and videos that has a simple background but it is failed with complex backgrounds.

Thinking of unconventional method, Zhong et al. provide an algorithm that can do text detection after transforming the image to discrete cosine transform domain. This alogorithm is useful because there is no need to decode the image to be processed but its accuracy is very limited. To have more quicker algorithm, Chen et al. have used many Adaboost classifier in series by training each weak one with a set of features. This method yield a high accuracy compared to other algorithms but it is not such high dealing with real images.

Component based methods

Components that may be our candidates could be determined using many options such as clustering based on colors or extreme area extraction. The next step is filtering our candidates to determine which is text and which is not based on manual automatically or manual classifiers. The advantage of this kind is that it is less expensive computationally wise compared to the previous kind because the amount of components needs processing is small. In addition, this method is not sensitive to rotation, font and scale change. Thus, this type became the most common in recent days.

Epshtein et al. introduce a new operator: stroke width transform (SWT). This operator could easily

extract text with different scales and angles from complex images. The disadvantage of this method is that is useful with horizontal text only.

Neumann et al. introduced a popular algorithm based on maximally stable extremal regions (MSER). This algorithm extracts specific areas that maximally stable and remove the invalid regions via a classifier. Then, those regions are linked with each other through a group of rules but these rules are useful with horizontal text only so, this method cannot be used with any rotations.

Huang et al. proposed a unique method for scene text detection that combined MSER with convolutional neural networks (CNN). The MSER extracts the candidates and the CNN checks the true candidates. This algorithm provide excellent results compared with conventional methods.

Hybrid methods

This method is an integration of texture based methods and component based methods, which combine the advantages of these two methods. Liu et al., introduced a method that extracts the edge pixels of possible text areas via edge detection strategy. Generating candidate text areas could be done using verifying gradient properties and then texture based method could be used classify the text regions and non text regions.

This algorithm share a common disadvantage with other algorithms that is applied only on horizontal text.

Until now we discussed the three types of methods and gave a brief about some algorithms. The next step is discuss our work by explaning the model we choose to apply and the dataset that we find it challenging enough to be an addition to the scientific community if we could achieve good results.

II. METHODOLOGY

A. Dataset

We tried CPTN model on a bit challenging dataset called “Total Text” provided by Center of Image and Signal Processing, Faculty of Computer Science and Information Technology, University of Malaya. The original edition contain 1555 images divided as following 1255 images as training set and 300 as test set. However, in my implementation, I used only 100 images as test set and the remaining as training set.

The ground truth offered in two types; rectangle and polygon. I have used the rectangle format because the trained model is compatible with it but the polygon format will provide better results because it surround the text in a better way. I mean here by rectangle and polygon the shape of detection box. For any image, its ground truth is represented as following, a matrix that contains all the detection and recognition information. Every row represent a new detected word. There are eight columns; the first four columns represent the coordinates of the two points that draw the detection box. Then the next two columns are the height and the width of detection box. The last two columns are the text itself and a character refers to text orientation; if it is horizontal, curved, multi orientations (combine both orientations).

Figure 1

Example from Total Text dataset

In my implementation, I only used the first four columns which represent the coordinates of the two points that draw the detection box.

The challenging properties of this dataset is that it contains many texts that is not horizontal. It contains large spectrum of different scales. Moreover, it have some images with low resolution. This makes it very difficult because it includes all the challenges that faced most of the models we mentioned in the literature review.

B. *CPTN Model*

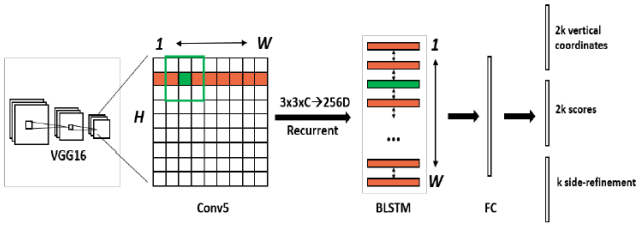

Figure 2
Architecture of CPTN

Architecture

Convolution of a 3x3 spatial window and the last convolutional maps (conv5) of the VGG16 model. The sequential windows in each row are recurrently
connected by a Bi-directional LSTM (BLSTM), where the convolutional feature (3x3xC) of each window is used as input of the 256D BLSTM (including two 128D LSTMs). The RNN layer is connected to a 512D fully-connected layer, followed by the output layer, which predicts text/non-text scores, y-axis coordinates and side refinement offsets of k anchors.

Figure 3
The CPTN output

Contributions

Connectionist Text Proposal Network (CTPN) mainly localizes text sequences in convolutional layers. This solves a variety of main problems that faced the previous approaches in the literature review that were building on character detection. It has a set of advantages that comes from the deep convolutional features. CTPN architecture which is shown in Fig. 2. It yields the following results:
Firstly, it converts the problem of text detection into localizing a sequence of fine- scale text. Then it makes an anchor regression mechanism that could estimate vertical location and text/non-text score of each text candidate with a satisfying localization accuracy. This is better than using RPN prediction which not provide a good accuracy.
Secondly, it connects the text candidates serially using network recurrence mechanism in the convolutional features. The advantage of this technique is that can broaden the options for the detector and enable it from exploring meaningful information of text which make it reliable.
Thirdly, the two previous methods are combined together to be suitable for the nature of the sequence of text.
The advantages of this model are that it is not very sensitive to scale change and verity of languages. It tackle these cases without any post filtering or preprocessing.

III. Evaluation Protocol

To evaluate the CPTN model, I used the concept of intersection over union (IOU) that is popular object detection problems. The complete evaluation pipeline is as following:

Firstly, I developed a function that calculates the IOU when it gets the ground truth box and the model output box, then, using a for loop, I did the same with all the boxes in the image.

Secondly, it calculates the average accuracy of one image by adding all the IOUs for the boxes

together and dividing by the number of boxes in the image.

Thirdly, it does the same for all the images and calculates the final accuracy by getting the average IOU over all the images.

This method is not optimal at all because it lacks the accurate evaluation. It shows the model worse than it is actually because in some cases when there are two words are close to each other, the model detect them as one text and draw one box while the labels of ground truth considered them as two different texts with two boxes. This makes the IOU score is very low because of the big difference in the dimensions between the two boxes that are compared, while actually our purpose is the detection, so it does not matter it is one word or more. The only important thing in this stage is that it is a text.

You can refer to recommendations section to know about the suggested solution to this problem.

IV. RESULTS AND DISCUSSION

The IOU score I get over 100 test images through transfer learning is 0.51 which is a moderate score that could be improved as you will see in the recommendations section but it is better than the IOU score after testing on 55 images from ICDAR 2015 dataset (the used data for training the CTPN model) which is 0.42. this indicates that our evaluation criteria needs much improvement because in any ways the score for the new test data cannot be better than the score for data looks like the data used for training this model

I could say that most of the failed cases is due to the curvature of the text. The model get its best scores in tackling horizontal texts. Moreover, the small scale yields some failed cases. Low resolution contributes also in the amount of failed cases but I cannot judge to what extent because the number of images with low resolutions in the test set is very small, so I cannot generalize the result at this point.

V. RECOMMENDATIONS

There are two points that need improvement in this project.

The first point is the method of evaluation and this is very important to successfully judge the model. I recommend add a new parameter to the evaluation process which is the distance between the different texts on the image. As we need to define a minimum distance that can say this two boxes could be one box without any problems or add them together may cause other problem on the model.

The second point is how to improve the IOU score and this is can be done by using another style of transfer learning. Instead of using finetuning the convnet style, we can use convnet as fixed feature extractor style where the last layer of the model is trained on the current data. This style may give better results because the model has a sense of what is the current data look like.

VI. REFERENCES

Zhu, Y., Yao, C. and Bai, X., 2015. Scene text detection and recognition: recent advances and future trends. Frontiers of Computer Science, 10(1), pp.19-36.

Zhi T, Weilin H, Tong.H, Pan H, and Yu .Q, 2016, Detecting Text in Natural Image with Connectionist Text Proposal Network.