TextBoxes Preliminary Summary

Shishir Jakati
University of Massachusetts, Amherst
sjakati@umass.edu

Abstract

Scene text recognition, as a text detection/localization problem, exhibits various characteristics which make it more difficult to solve with off-the-shelf text detection methods. One major problem is the enlarged aspect ratio of bounding boxes on words. TextBoxes[1] attempts to solve this problem through the use of rectangular receptive fields, fully convolutional networks, and predefined aspect ratios better suited for word instance detections.

1. Network Architecture

TextBoxes utilizes the VGG-16[3] network as a backbone. Specifically, it utilizes layers from blocks 1 through 4, and then modifies the VGG network by replacing the fully connected layers with convolutional layers. Additionally, these final layers are followed by more convolutional and pooling layers.

1.1. Text-Box Layers

One major facet of the TextBox network framework is the use of text-box layers. These layers are inserted as output layers between various intermediate convolutional layers, as well as the final output of the network. These layers are trained to produce and regress text boxes for suspected word instances.

These text-box layers produce offsets from each map locations, which include both the confidence level that a text instance is present, as well as the offset values $(\Delta x, \Delta y, \Delta w, \Delta h, c)$. These offset values are used to indicate a box $b = (x, y, w, h)$, from the original map location (i, j) associated with the default box $b_0 = (x_0, y_0, w_0, h_0)$.

To deal with the varying aspect ratios and relative sizes of text instances, each map location has various default boxes of different sizes. These different default boxes fit the following aspect ratios: 1,2,3,5,7,and 10. In addition, to reduce the horizontal density of the horizontally-dominant boxes, each default box comes with vertical offsets. These horizontal boxes are learned through the use of 1*5 convolutional filters; these effectively make the receptive field of the filters horizontally rectangular.

1.2. Learning

The loss function is adopted from the SSD: Single Shot Multibox Detector [2]. We let x be the match indication matric, c be the confidence, l be the predicted location, and g be the ground-truth location. The loss objective is defined as follows:

$$L(x, c, l, g) = \frac{1}{N}(L_{conf}(x, c) + \alpha L_{loc}(x, l, g))$$

N is the number of default boxes that match ground-truth boxes. If there are no boxes which match the ground truth boxes we set the loss to be 0. L_{loc} is a smooth L1 loss, and L_{conf} is a 2-class softmax loss.

The location loss can be further explained. The indication matrix x is defined as follows:

$$x_{ij} = 1, 0$$

It is an indicator for matching the i-th default box to the j-th ground-truth box.

1.3. Multi-Scale Inputs

The training procedure includes rescaling input images to the following scales: 300 * 300, 700 * 700, 300 * 700, 500 * 700, 1600 * 1600.

1.4. Non-Maximum Suppression

Due to the network architecture, there are many outputs from all text-box layers. These are aggregated and then NMS is applied to these. For multi-scale inputs, an additional NMS is applied.

2. Word Spotting and End-to-End Recognition

As a framework used for text localization and recognition, TextBoxes adopts the CRNN model as a text recognizer. The CTC output layer estimates sequence probability conditioned on the input image. Plainly, the CTC output layer provides a probabilistic estimate as to whether a certain text instance is probable given the input image. The

probability score, s, is defined as follows:

$$s = max_{w \in W} p(w|I)$$

W is the predefined lexicon, in this case a generic lexicon which consists of 90,000 English words. w is the sequence found in the image, I.

When considering Word Spotting, TextBoxes produces a redundant set of word candidates by detecting with a low score threshold and a high NMS overlap threshold. Another re-evaluation, using the probabilistic score above, is performed along with a second thresholding and NMS.

References

[1] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu. Textboxes: A fast text detector with a single deep neural network, 2016. 1

[2] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. 2015. 1

[3] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition, 2014. 1