Feature Embedding Based Text Instance Grouping for Largely Spaced and Occluded Text Detection

Pan Gao Qi Wan RenWu Gao LinLin Shen*
Computer Vision Institute
Shenzhen University, Shenzhen, China

Introduction

In this paper, we focus on the accurate detection of largely spaced, slender or curved, or partially obscured text instances. We propose Text Instance Embedding Module (TIEM) to embed each pixel into feature space and the average pixel feature in the text area is regarded as the feature of the text area. Subsequently, text candidate regions with similar features are grouped by the proposed text instance grouping algorithm. By doing this, text instance segmented into several regions, due to the large space between texts, slender or curved shape, or partial occlusion, can be grouped into one.

Figure 1:Text with large space between texts/characters

Figure 2: Text with partial occlusion

Proposed Method

There are two branches in our network: a) text segmentation branch and b) text instance embedding branch. The embedding branch is composed of TIEM that outputs four-dimensional embedding vectors for each pixel.

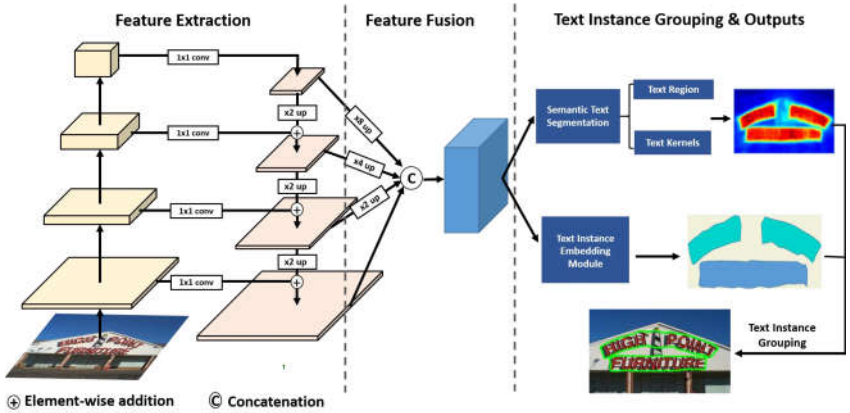

Figure 3: Illustration of the pipeline of our framework. The left part is feature extraction based on FPN. The middle part is feature fusion from low to high levels. The right part is the output of network and the proposed text instance grouping.

Loss Function of TIEM

Since we don't know the label of each pixel's embedding vector, we borrow the idea of clustering and supervise the module training by reducing the intra-instance distance and increasing the inter-instance distance.

a) Reducing the intra-instance distance

$$L_{intra} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{|O(T_i)|}\sum_{p\in O(T_i)}\ln(Dis(p,T_i)+1)$$

$$Dis(p,T_i) = max(\|F_p - F_{T_i}\| - \theta_{intra}, 0)^2$$

b) Increasing the inter-instance distance

$$L_{inter} = \frac{1}{N(N-1)}\sum_{i=1}^{N}\sum_{j=1, j\neq i}^{N}\ln(Dis(K_i,K_j)+1)$$

$$Dis(K_i,K_j) = max(\theta_{inter} - \|F_{K_i} - F_{K_j}\|, 0)^2$$

Where N is the number of text instances in the picture, T_i represents the i-th text instance, Dis(p; T_i) represents the distance between the pixel p and the text instance T_i, O(·) is the OHEM function that return the top 30% pixels with the largest feature distance and θ_{intra} is a constant set to 0.4. K_i represents the kernel area of the i-th text instance. θ_{inter} is a constant set to 4.

Figure 4: Example results of our method on CTW-1500(a), Total-Text(b) and IC15(c).

Table 1: Single-scale Result On CTW-1500. "P", "R" And "F" Represent The Precision, Recall And F-measure Respectively.

Method		CTW-1500		
		P	R	F
Training from scratch	CTPN*[23]	60.4*	53.8*	56.9*
	Seglink* [24]	42.3*	40.0*	40.8*
	EAST*[7]	78.7*	49.1*	60.4*
	CTD+TLOC [19]	77.4	69.8	73.4
	PSENet-1s [15]	80.6	75.6	78.0
	PAN-640 [1]	84.6	77.7	81.0
	Ours	**85.1**	**77.9**	**81.3**
Pre-trained on external dataset	TextSnake[25]	67.9	**85.3**	75.6
	CSE[26]	78.7	76.1	77.4
	LOMO[27]	**89.2**	69.6	78.4
	SAE[2]	82.7	77.8	80.1
	TextField[13]	83.0	79.8	81.4
	MSR[14]	84.1	79.0	81.5
	PSENet-1s[15]	84.8	79.7	82.2
	DB[28]	86.9	80.2	83.4
	CRATF[29]	86.0	81.1	83.5
	PAN-640[1]	86.4	81.2	**83.7**
	Ours	87.9	79.9	**83.7**

Summary

In this paper, we have presented a novel framework, which includes the proposed text instance embedding module (TIEM) and text instance grouping algorithm, to detect text with occlusion, arbitrary-shape and large spacing. Extensive experiments demonstrate the superior advantages of our approach when compared to previous state-of-the-art text detectors. In particular, our method achieves the state-the-the-art performance on text-line level annotated benchmark (i.e. CTW-1500) and competitive performance on word level annotated benchmark (i.e. Total-Text, IC15).