Shape Robust Text Detection with Progressive Scale Expansion Network

Wenhai Wang [1,4*] , Enze Xie [2,5*] , Xiang Li [3,4** †] , Wenbo Hou [1] , Tong Lu [1‡] , Gang Yu [5] , Shuai Shao [5]

[1] National Key Lab for Novel Software Technology, Nanjing University

[2] Department of Comuter Science and Technology, Tongji University

[3] School of Computer and Engineering, Nanjing University of Science and Technology

[4] Momenta

[5] Megvii (Face++) Technology Inc.

Wang W, Xie E, Li X, et al. Shape Robust Text Detection with Progressive Scale Expansion Network[J]. arXiv preprint arXiv:1903.12473, 2019.

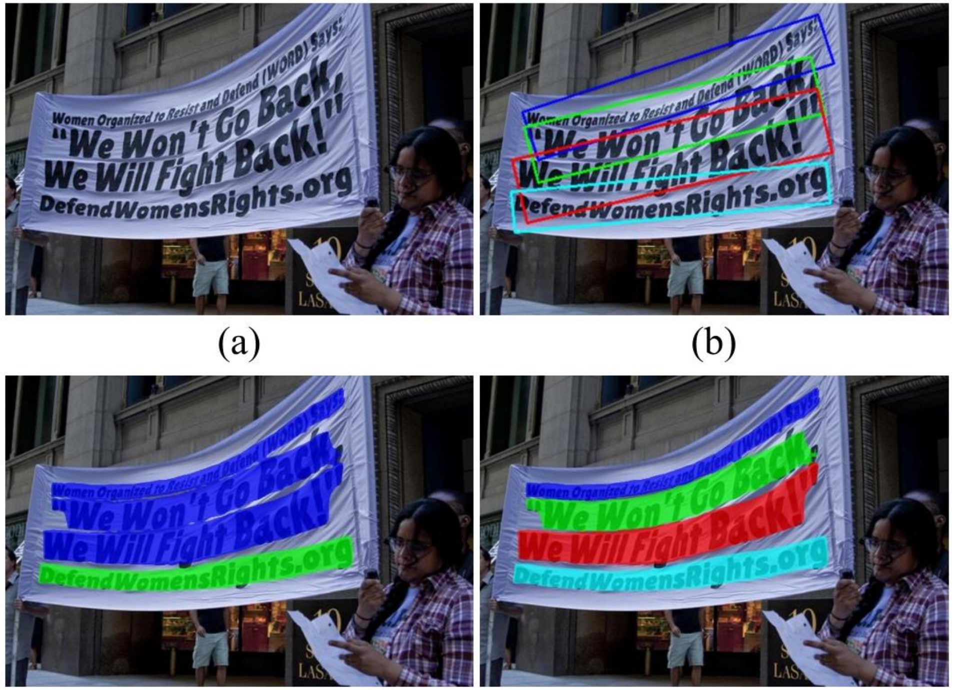
1 Introduction

For scene text detection in the wild, the existing CNN based algorithm can be divided into two categories:

- ➢ Regression-based approaches
    - Text targets are represented in the forms of rectangles or quadrangles with certain orientations
- ➢ Segmentation-based approaches
    - Locate text instance based on pixel-level classification

A novel kernel-based framework, namely, Progressive Scale Expansion Network (PSENet):

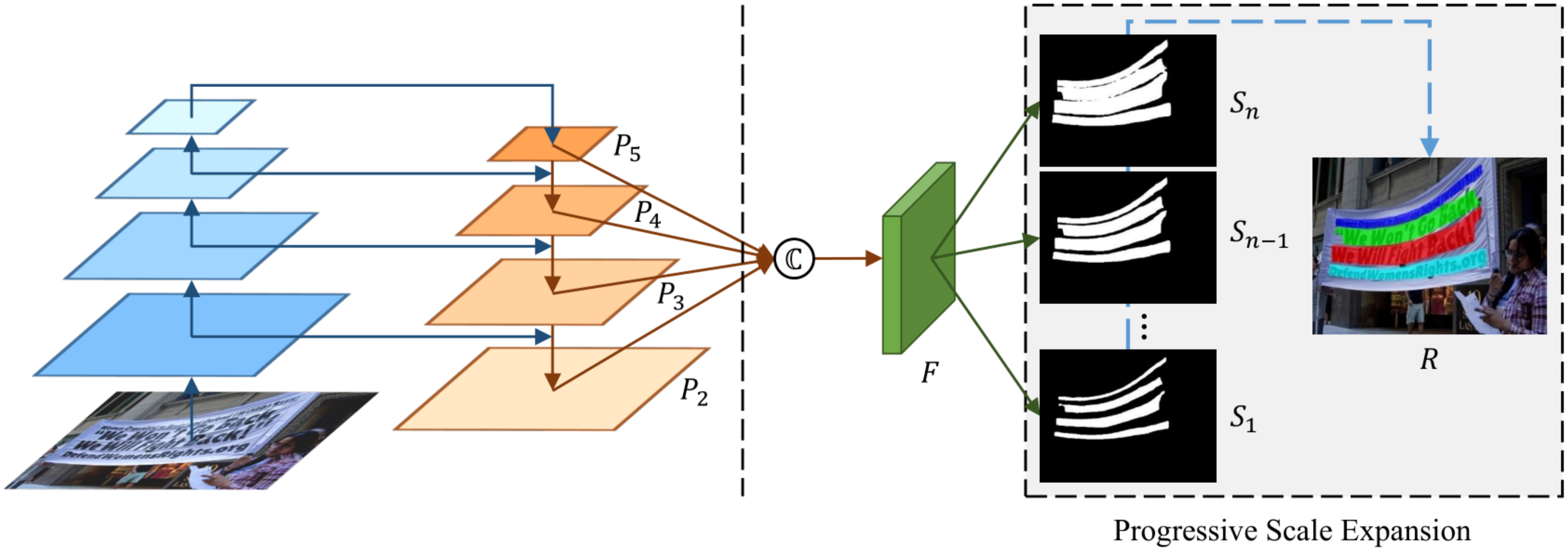
- ➢ Performs pixel-level segmentation
- ➢ Propose a progressive scale expansion algorithm
    - Based on Breadth-First-Search (BFS)



(a) (b) (c) (d)

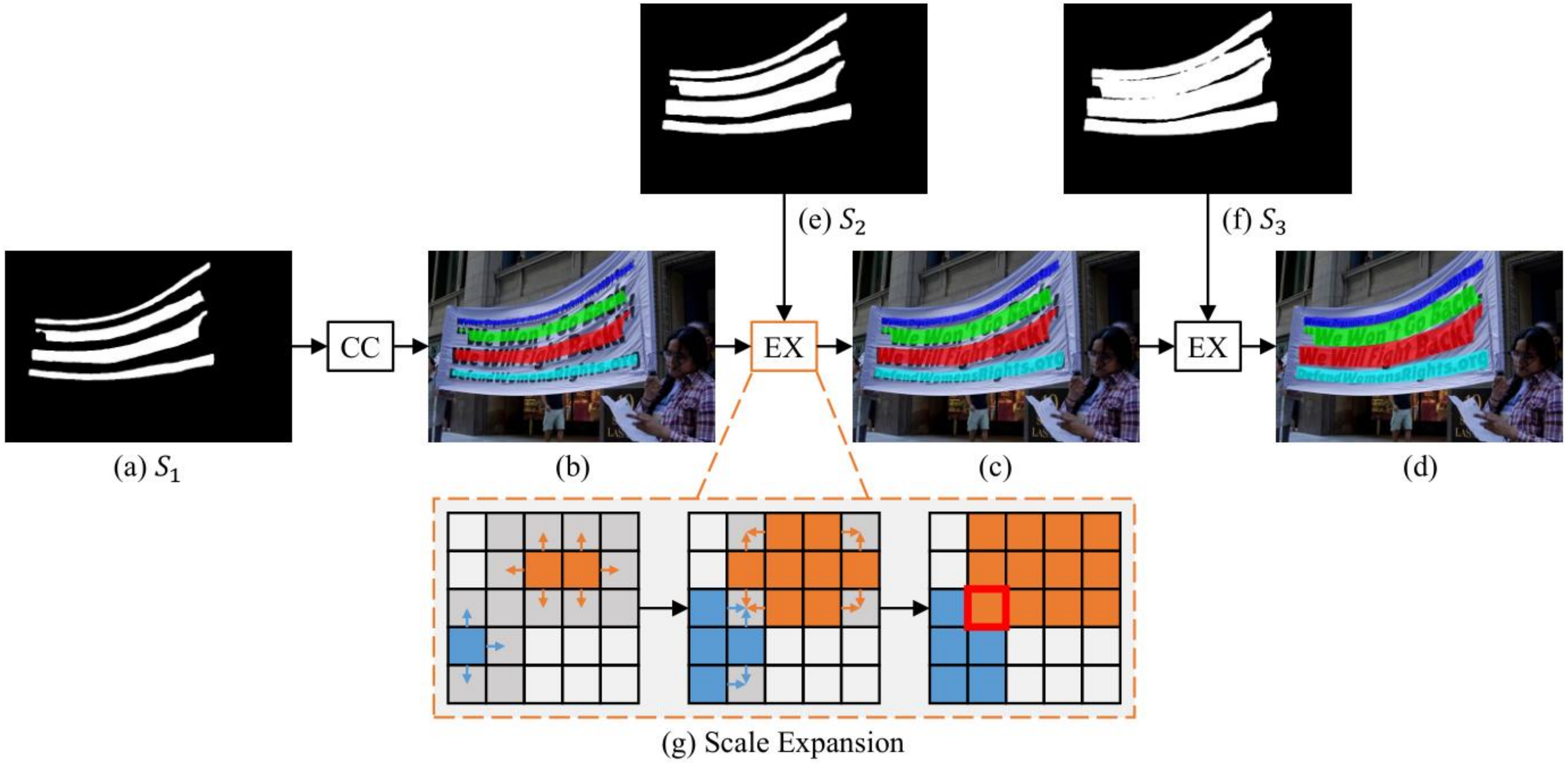
2 Overall Pipeline

$$\begin{aligned} F &= \mathbb{C}(P_2, P_3, P_4, P_5) \\ &= P_2 \parallel \mathrm{Up}_{\times 2}(P_3) \parallel \mathrm{Up}_{\times 4}(P_4) \parallel \mathrm{Up}_{\times 8}(P_5), \end{aligned}$$



3 Progressive Scale Expansion Algorithm

The confusing pixel can only be merged by one single kernel on a first-come-first-served basis.



4 Label Generation

Consider the scale ratio as r_i, the margin d_i between p_n and p_i can be calculated as:

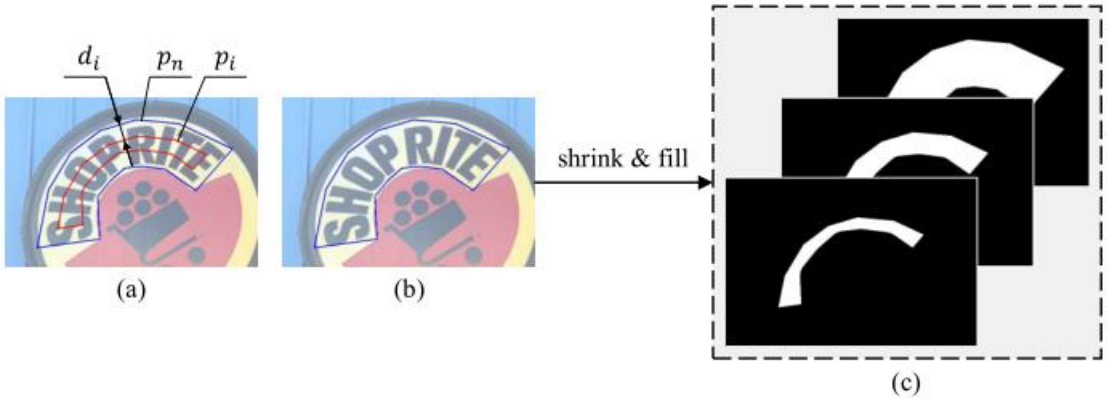
$$d_i = \frac{\text{Area}(p_n) \times (1 - r_i^2)}{\text{Perimeter}(p_n)},$$

Area(·) is the function of computing the polygon area, Perimeter(·) is the function of computing the polygon perimeter.

The scale ratio r_i for ground truth map G_i as:

$$r_i = 1 - \frac{(1 - m) \times (n - i)}{n - 1},$$

m is the minimal scale ratio, which is a value in (0,1].



5 Loss Function

Loss function:

$$L = \lambda L_c + (1 - \lambda) L_s,$$

Dice coefficient:

$$D(S_i, G_i) = \frac{2 \sum_{x,y} (S_{i,x,y} \times G_{i,x,y})}{\sum_{x,y} S_{i,x,y}^2 + \sum_{x,y} G_{i,x,y}^2},$$

L_c represent the loss for the complete text instances, focuses on segmenting the text and non-text region:

$$L_c = 1 - D(S_n \cdot M, G_n \cdot M),$$

M is the training mask.

L_s is the loss for shrunk text instances:

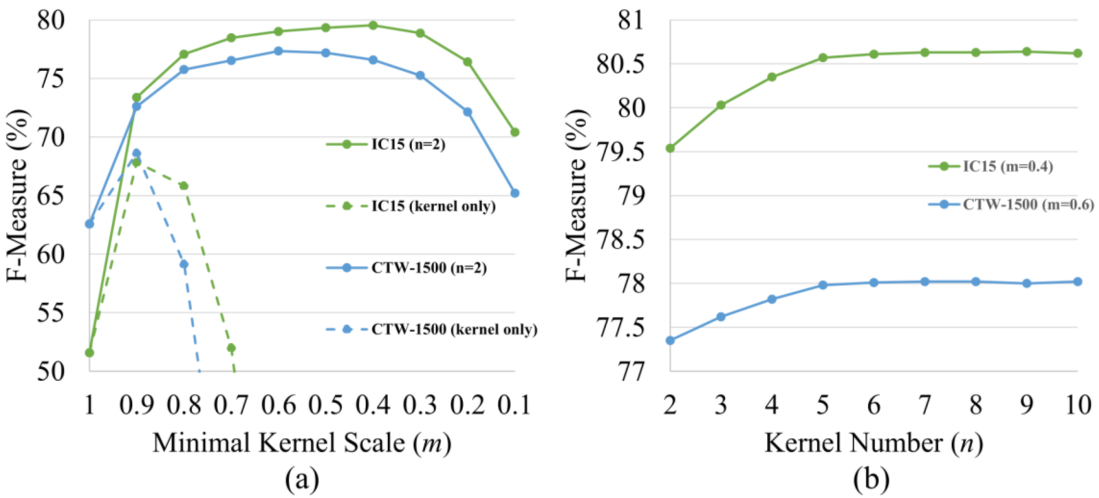
$$L_s = 1 - \frac{\sum_{i=1}^{n-1} D(S_i \cdot W, G_i \cdot W)}{n - 1},$$

$$W_{x,y} = \begin{cases} 1, & if\ S_{n,x,y} \geq 0.5; \\ 0, & otherwise. \end{cases}$$

W is a mask which ignores the pixels of the non-text region in S_n.

6 Ablation Study

- Influence of the minimal kernel scale.
  - When m is too large, separating the text instances lying closely to each other is hard.
  - When m is too small, PSENet will split a whole text line into different parts incorrectly.
- Influence of the kernel numbers.
  - The advantage of multiple kernels is that it can accurate reconstruct two text instances where they lying closely to each other.
- Influence of the backbone.
  - Adopt ResNet as backbone with three different depths of {50, 101, 152}.
  - test on the large scale dataset IC17-MLT.



Methods	P	R	F
PSENet (ResNet50)	73.7	68.2	70.8
PSENet (ResNet101)	74.8	68.9	71.7
PSENet (ResNet152)	75.3	69.2	72.2

Table 1. Performance grows with deeper backbones on IC17-MLT. “P”, “R” and “F” represent the precision, recall and F-measure respectively.

7 Comparisons with State-of-the-Art Methods

- Detecting **Curve Text** on **CTW1500** and **Total-Text**, which mainly contains the curve texts.
- Detecting **Oriented Text** on the **IC15**.

Method	Ext	CTW1500			
		P	R	F	FPS
CTPN [36]	-	60.4*	53.8*	56.9*	7.14
SegLink [32]	-	42.3*	40.0*	40.8*	10.7
EAST [43]	-	78.7*	49.1*	60.4*	**21.2**
CTD+TLOC [24]	-	77.4	69.8	73.4	13.3
TextSnake [26]	✓	67.9	85.3	75.6	-
PSENet-1s	-	80.57	75.55	78.0	3.9
PSENet-1s	✓	84.84	79.73	**82.2**	3.9
PSENet-4s	✓	82.09	77.84	79.9	8.4

Table 2. The single-scale results on CTW1500. "P", "R" and "F" represent the precision, recall and F-measure respectively. "1s" and "4s" means the width and height of output map is $1/1$ and $1/4$ of the input test image. * indicates the results from [24]. "Ext" indicates external data.

Method	Ext	Total-Text			
		P	R	F	FPS
SegLink [32]	-	30.3	23.8	26.7	-
EAST [43]	-	50.0	36.2	42.0	-
DeconvNet [2]	-	33.0	40.0	36.0	-
TextSnake [26]	✓	82.7	74.5	78.4	-
PSENet-1s	-	81.77	75.11	78.3	3.9
PSENet-1s	✓	84.02	77.96	**80.87**	3.9
PSENet-4s	✓	84.54	75.23	79.61	**8.4**

Table 3. The single-scale results on Total-Text. "P", "R" and "F" represent the precision, recall and F-measure respectively. "1s" and "4s" means the width and height of output map is $1/1$ and $1/4$ of the input test image. "Ext" indicates external data. Note that EAST and SegLink were not fine-tuned on Total-Text. Therefore their results are included only for reference.

Method	Ext	IC15			
		P	R	F	FPS
CTPN [36]	-	74.22	51.56	60.85	7.1
SegLink [32]	✓	73.1	76.8	75.0	-
SSTD [11]	✓	80.23	73.86	76.91	7.7
WordSup [13]	✓	79.33	77.03	78.16	-
EAST [43]	-	83.57	73.47	78.2	**13.2**
RRPN [28]	-	82.0	73.0	77.0	-
R^2CNN [16]	-	85.62	79.68	82.54	-
DeepReg [12]	-	82.0	80.0	81.0	-
PixelLink [4]	-	82.9	81.7	82.3	7.3
Lyu et al. [27]	✓	94.1	70.7	80.7	3.6
RRD [20]	✓	85.6	79.0	82.2	6.5
TextSnake [26]	✓	84.9	80.4	82.6	1.1
PSENet-1s	-	81.49	79.68	80.57	1.6
PSENet-1s	✓	86.92	84.5	**85.69**	1.6
PSENet-4s	✓	86.1	83.77	84.92	3.8

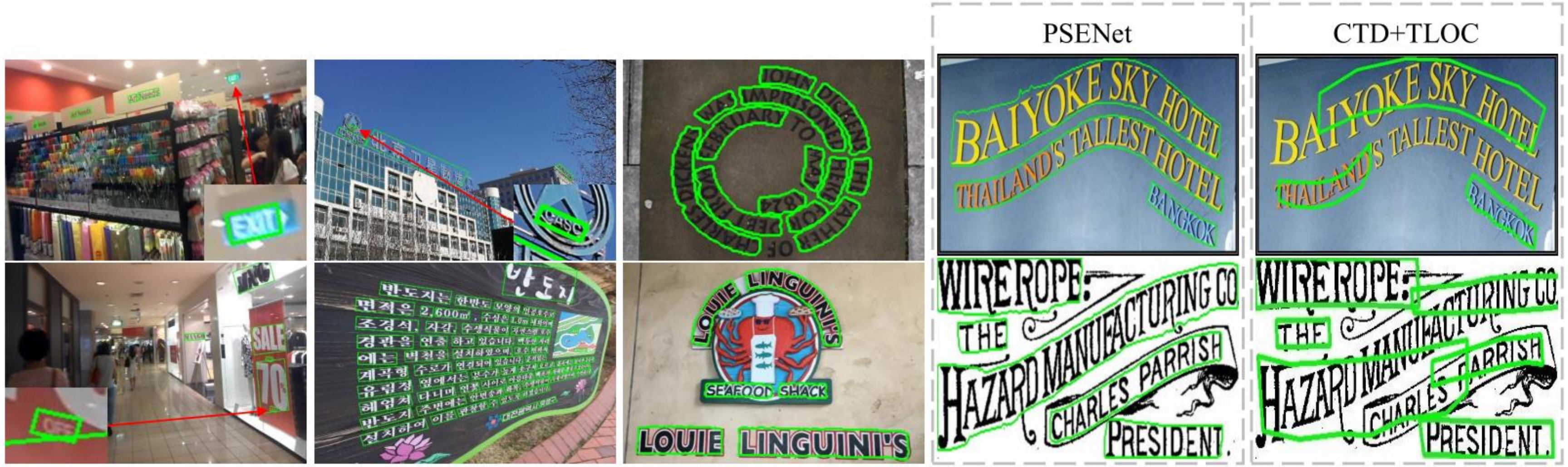
Table 4. The single-scale results on IC15. "P", "R" and "F" represent the precision, recall and F-measure respectively. "1s" and "4s" means the width and height of output map is $1/1$ and $1/4$ of the input test image. "Ext" indicates external data.

7 Comparisons with State-of-the-Art Methods

- Detecting **MultiLingual Text** on **IC17-MLT** benchmark.

Method	Ext	IC17-MLT		
		P	R	F
linkage-ER-Flow [1]		44.48	25.59	32.49
TH-DL [1]		67.75	34.78	45.97
TDN SJTU2017 [1]		64.27	47.13	54.38
SARI FDU RRPN v1 [1]		71.17	55.50	62.37
SCUT DLVClab1 [1]		80.28	54.54	64.96
Lyu et al. [27]	✓	83.8	55.6	66.8
PSENet (ResNet50)	-	73.77	68.21	70.88
PSENet (ResNet152)	-	75.35	69.18	**72.13**

Table 5. The single-scale results on IC17-MLT. "P", "R" and "F" represent the precision, recall and F-measure respectively. "Ext" indicates external data.



(a) ICDAR 2015 (b) ICDAR 2017 MLT (c) Total-Text (d) CTW1500

Figure 7. Detection results on three benchmarks and several representative comparisons of curve texts on CTW1500. More examples are provided in the **supplementary materials**.

8 Speed Analyze

- ResNet50 and ResNet18 are adopted as the backbone to trade off the speed and accuracy.
- Scale the long edge of {1280, 960, 640} as input to test the speed.

Method	Res	F	Time consumption			FPS
			backbone(ms)	head(ms)	PSE(ms)	
PSENet-1s (ResNet50)	1280	82.2	50	68	145	3.9
PSENet-4s (ResNet50)	1280	79.9	50	60	10	8.4
PSENet-4s (ResNet50)	960	78.33	33	35	9	13
PSENet-4s (ResNet50)	640	75.6	18	20	8	21.65
PSENet-4s† (ResNet18)	960	74.30	10	17	10	26.75

Table 6. Time consumption of PSENet on CTW-1500. The total time is consist of backbone, head of segmentation and PSE part. † indicates training from scratch. "Res" represents the resolution of the input image. "F" represent the F-measure.