Focus Entirety and Perceive Environment for Arbitrary-Shaped Text Detection

Xu Han, Junyu Gao, *Member, IEEE,* Chuang Yang, Yuan Yuan, *Senior Member, IEEE* and Qi Wang, *Senior Member, IEEE*

***Abstract*—Due to the diversity of scene text in aspects such as font, color, shape, and size, accurately and efficiently detecting text is still a formidable challenge. Among the various detection approaches, segmentation-based approaches have emerged as prominent contenders owing to their flexible pixel-level predictions. However, these methods typically model text instances in a bottom-up manner, which is highly susceptible to noise. In addition, the prediction of pixels is isolated without introducing pixel-feature interaction, which also influences the detection performance. To alleviate these problems, we propose a multi-information level arbitrary-shaped text detector consisting of a focus entirety module (FEM) and a perceive environment module (PEM). The former extracts instance-level features and adopts a top-down scheme to model texts to reduce the influence of noises. Specifically, it assigns consistent entirety information to pixels within the same instance to improve their cohesion. In addition, it emphasizes the scale information, enabling the model to distinguish varying scale texts effectively. The latter extracts region-level information and encourages the model to focus on the distribution of positive samples in the vicinity of a pixel, which perceives environment information. It treats the kernel pixels as positive samples and helps the model differentiate text and kernel features. Extensive experiments demonstrate the FEM's ability to efficiently support the model in handling different scale texts and confirm the PEM can assist in perceiving pixels more accurately by focusing on pixel vicinities. Comparisons show the proposed model outperforms existing state-of-the-art approaches on four public datasets.**

***Index Terms*—Scene text detection, arbitrary-shaped text, real-time detection.**

I. Introduction

OVER the past few years, research on scene text detection has gained increased concerns due to its various applications, including license plate detection, signboard reading, autonomous driving, and scene understanding. With the rapid development of object detection and image segmentation, scene text detection [1]–[8] achieves significant progress. However, accurately locating scene text remains tricky due to font, color, and scale variation. The irregular shape is still the most formidable challenge of them.

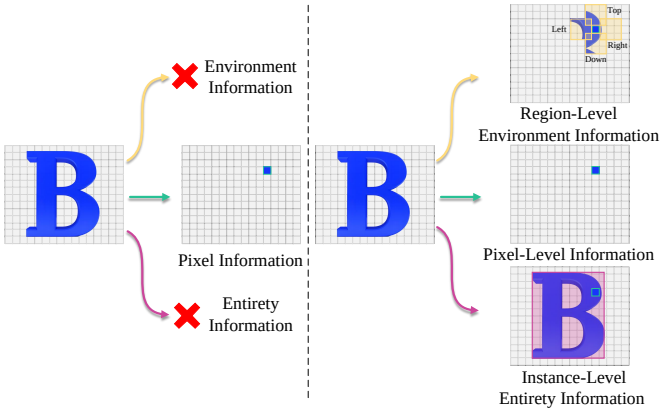


Fig. 1. Illustration of the multi-level information extraction for existing segmentation-based methods and ours. (a) Existing segmentation-based methods [9], [10], [11] only focus on pixel-level information. (b) Our method further extracts region-level and instance-level features to suppress the noise.

Among the numerous recent advanced approaches, segmentation-based approaches stand out as their flexible pixel prediction can cope with it. PSENet [9] produces multi-scale pixel-level predictions to identify different scales of kernels. DBNet [10] predicts the threshold and whether the pixel is a kernel or not. Although the above methods inherit and develop the advantage of pixel prediction, they still model text instances in a bottom-up manner, which is highly susceptible to noise. Moreover, these methods focus only on distinguishing text pixels from non-text pixels, ignoring essential features of the text instance. The fundamental purpose of text detection is to locate text instances. To address the above problem, we propose a focus entirety module (FEM), which helps the proposed model extract instance-level features and utilizes a top-down scheme to model texts which reduces the influence of noises. It assigns consistent information to the pixels within the same instance to strengthen the cohesion of pixels. In addition, the FEM encourages pixels to focus on the scale of instances and helps the model recognize features at different size instances to deal with text scale variations.

Furthermore, existing segmentation-based methods focus only on predicting single pixels isolated without introducing information interaction. For example, TextLeaf [4] focuses on

X. Han, C. Yang are with the School of Computer Science, and with the School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an 710072, P. R. China. (E-mail: hxu04100@gmail.com, omtcyang@gmail.com).

J. Gao, Y. Yuan, and Q. Wang are with the School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an 710072, P. R. China. (E-mail: gjy3035@gmail.com, y.yuan1.ieee@gmail.com, crabwq@gmail.com).

This work was supported by the National Natural Science Foundation of China under Grant U21B2041.

Qi Wang is the corresponding author.

the text kernel mask and rebuilds the instances by predicting the lateral and thin veins. CT-Net [12] predicts the kernel probability map and centripetal shift map to obtain detection results. However, it results in separate predictions of each pixel with no sufficient connection between them, influencing the detection performance. It is considered that the visual system often utilizes the distribution of the surrounding environment to determine the properties of an object that is challenging to judge. The model should focus on information about individual pixels and the surrounding environment to help construct the entire textual knowledge system. To be specific, we propose a perceive environment module (PEM), which extracts region-level features and facilitates predicting peripheral pixel interactions to obtain synergistic progress. It perceives the positive sample distribution around the pixel in four directions to recognize hard-to-identify pixels effectively. Furthermore, the PEM treats the kernel pixels as positive samples and helps the model differentiate text and kernel features, improving the incomplete kernel semantics.

As we can see from Fig. 1, existing segmentation-based methods [9]–[11] model text instances in a bottom-up manner that only focus on pixel-level features lack coarse global features, which is susceptible to noise interference. The proposed method is named focus entirety and perceive environment (FEPE), which models text structure from three levels: coarse global features (instance-level), fine-grained local features (pixel-level), and their intermediate state (region-level). Additionally, FEM and PEM can be removed during the testing phase. This means they improve the accuracy without affecting the inference speed and can be further integrated with other methods to improve their performance. The main contributions of this work are as follows:

1) A focus entirety module (FEM) is proposed to extract instance-level features and model texts in a top-down scheme that reduces the influence of noise. It assigns consistent entirety information to pixels within the same instance to improve their cohesion and emphasizes the scale of instances to which pixels belong, enabling the model to distinguish varying scale texts effectively.
2) A perceive environment module (PEM) is proposed to extract region-level information and encourage the model to focus on the distribution of positive samples in the vicinity of a pixel, which perceives environment information. It treats the kernel pixels as positive samples and helps the model differentiate text and kernel features, improving the incomplete kernel semantics.
3) An efficient and effective text detector is proposed based on the above modules named FEPE, which attend simultaneously to coarse global features and fine-grained local features. It achieves state-of-the-art (SOTA) performance on multiple public benchmarks, which include numerous horizontal, rotated, and irregular-shaped texts.

The rest of the paper is structured as follows. Some related work is presented in Section II. Section III describes the detail of FEM, PEM, and FEPE. Furthermore, we describe the multi-task loss used and the training details. In Section IV, ablation studies on four benchmarks strongly demonstrate the superiority of the proposed FEM and PEM. In addition, extensive experiment results are compared with state-of-the-art methods, proving the superiority and advancement of FEPE. Finally, the whole paper is summarized in Section V.

II. Related work

Deep learning has rapidly advanced in recent years, making significant progress in text detection. Existing methods are generally divided into regression-based methods, connected-component-based methods, and segmentation-based methods. The related works are briefly introduced as follows.

A. Regression-based methods

The majority of regression-based methods for text detection are inspired by object detection frameworks, such as Faster-RCNN [13], and refined based on the characteristics of the text. Liao *et al.* proposed TextBoxes [14], which detect texts by revising the anchor and convolution kernels. Then, TextBoxes++ was [15] proposed to cope with multi-directional text, which adds an angle parameter. Zhou *et al.* [16] divided text as rotated text and quadrangle text to predict different parameters based on FCN [17]. Liao *et al.* proposed RRD [18], which used rotation-sensitive features to detect oriented texts. He *et al.* proposed SSTD [19], which used an attention module and an auxiliary loss to obtain detection results. Most of the above methods are limited by sophisticated post-processing, which affects their development. Moreover, the irregular-shaped text is a tricky problem for the above methods. Dai *et al.* [20] proposed progressive contour regression to cope with it, which iterative update text contours. The initial result is horizontal text, which is gradually optimized to multi-directional and irregular-shaped text. FCENet [21] and ABCNet [22] represented text contours by Fourier Signature Vector and Bezier Curve, respectively. Although the above methods can deal with irregular-shaped text, the complicated structure influences the efficiency.

B. Connected-component-based methods

Connected-component-based methods locate and group characters or parts of instances to reconstruct instances. CRAFT [23] modeled text instance by judging the proximity of the characters to each other. DRRG [24] utilized graph convolutional networks (GCN) to infer the relationships between text parts. PixeLink [25] predicted pixel score map and the relationships with surrounding pixels to detect text. SegLink [26] represented text instances as segments and links that merged segments according to the predictions of links. Long *et al.* proposed TextSnake [27], which represents text like a snake. It utilized circles to describe text components. Although connected-component-based approaches work well when dealing with irregular-shaped texts, the complex merging process remains an open problem.

C. Segmentation-based methods

The critical goal of segmentation-based approaches is to predict whether a pixel is text. PSENet [9] represented text

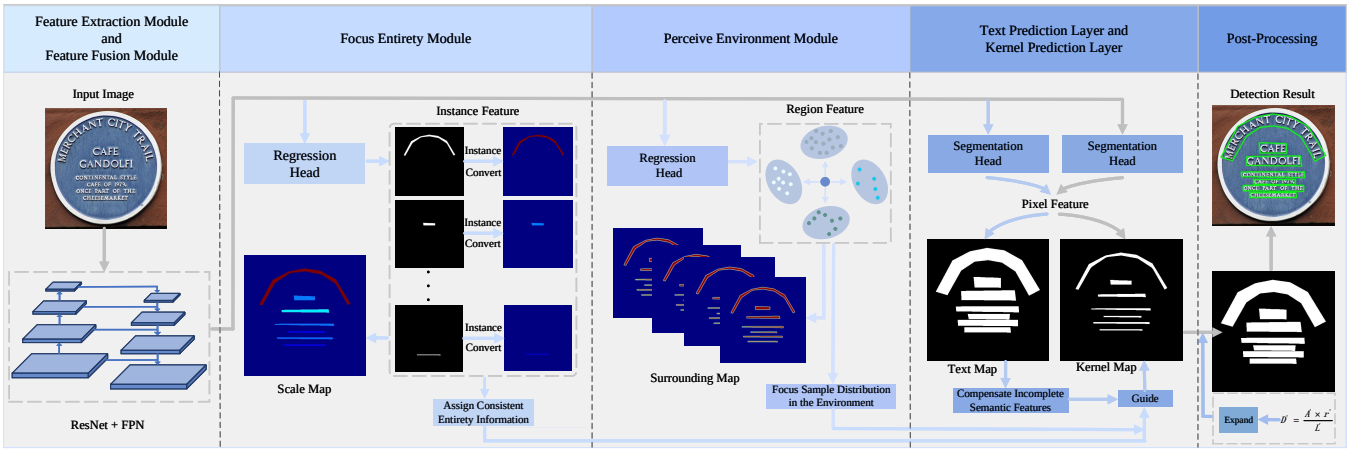


Fig. 2. The overall framework of the proposed FEPE. During the inference stage, only the feature extraction module, feature fusion module, kernel prediction layer, and post-processing are retained, and the others can be removed. D', r', A, and L' represent the expanding distance, expand factor, area, and perimeter of the kernel.

as different scale kernels and reconstructed text instances according to a progressive expansion algorithm. Lyu *et al.* [28] proposed an approach that generates candidate boxes by grouping corner points and assess them using region segmentation. Then, the candidate boxes were assessed by region segmentation and suppressed by NMS. TextField [29] predicted the direction field while segmenting the text. It utilized the direction field to separate geographically close texts. LeafText [4] treated text instance as leaf and utilized main, lateral, and thin veins to form text. The above approaches perform well for dealing with irregular-shaped texts but still lack efficiency. DBNet [10] segmented the score and regressed the threshold to surprise the result of DB. Benefiting from that DB module can be removed during inference and adopt a lightweight backbone, it achieved excellent performance while maintaining a high inference speed. On top of that, DBNet++ [11] introduced an attention mechanism that improves detection accuracy with minimal effect on speed. CM-Net [30] proposed a novel text kernel representation named concentric mask and learned some auxiliary features to assist in detecting text. PAN [31] adopted a lightweight backbone and utilized FFM and FPEM to strengthen features. It predicted similar vectors and proposed learnable post-processing for text restructuring.

III. Method

The overall pipeline of the proposed approach is introduced and illustrated first in this section. Then, we describe and visualize the label generation procedure. Furthermore, we present the focus entirety module (FEM) and perceive environment module (PEM) in detail. Finally, the multi-task constraints loss and training detail are described.

A. Overall Structure

The overall structure of FEPE is shown in Fig. 2, which consists of the feature extracting module, feature fusion module, kernel prediction layer, text prediction layer, focus entirety module, and perceive environment module. During the training stage, a multi-level feature map is obtained through the feature extracting and feature fusion modules. Then, the kernel prediction layer, text prediction layer, focus entirety module and perceive environment module output their prediction results. Only the kernel prediction layer is activated during the inference stage, while the text prediction layer, focus entirety module, and perceive environment module are deactivated. The FEM extracts instance-level features and adopts a top-down scheme to model texts which reduces the influence of noises. Specifically, it assigns consistent information to pixels of the same instance to encourage the clustering of these pixels. In addition, it emphasizes the scale information, enabling the model to distinguish varying scale texts effectively. The PEM defines the kernels as a positive sample and helps the model distinguish the different features between kernels and texts. It defines difference values between edge, internal, and external pixels, thereby helping the model comprehend the distance between contour and pixels. It strengthens the model's comprehension of kernel and text. Benefiting from the superiority of FEM and PEM, when the predictions deviate from the ground truth, these errors tend to be corrected, significantly improving detection accuracy.

The details of the above modules are described as follows. ResNet [32] with deformable convolution [33], [34] is selected as the feature extraction module. Multi-level feature maps are generated through it. The size of feature maps are $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, $\frac{1}{32}$ of the input image, respectively. We adopt the feature pyramid network (FPN) [35] to merge multi-level feature maps and obtain the fused feature map $\mathrm{F_f}$. This feature map contains both lower-level semantic features and high-level global features. We utilized two segmentation heads with the same architecture for the text kernel prediction layer and text prediction layer, which are shown as follows:

$$\mathrm{H_1} = \mathrm{ReLU_{BN}}(\mathrm{Conv_{3\times3,64}}(\mathrm{F_f})), \tag{1}$$

$$\mathrm{H_2} = \mathrm{ReLU_{BN}}(\mathrm{ConvT_{3\times3,64}}(\mathrm{H_1})), \tag{2}$$

$$\mathrm{H_3} = \mathrm{Sigmoid}(\mathrm{ConvT_{3\times3,1}}(\mathrm{H_2})), \tag{3}$$

where $\mathrm{H_3}$, Conv, and ConvT represent the output results, convolution operation, and transposed convolution operation, respectively. $\mathrm{ReLU_{BN}}$ represent the ReLU activation function

Algorithm 1: Text Kernel Label Generation

Data: text map M_t, minimum area threshold A_{min}, shrinking ratio δ, area of instance S, perimeter of instance L, width W and height H

Result: text kernel map M_k

initializing $M_k \in \mathbb{R}^{W,H}$;
for *i*th instance *in* M_t **do**
  $\text{offset}_i \leftarrow \frac{S_i}{L_i}(1-\delta^2)$;
  text kernel$_i \leftarrow$ shrinking contour inward by offset$_i$;
  if area of text kernel$_i > A_{min}$ **then**
    drawing text kernel$_i$ on M_k;
  end
end

and batch normalization layer [36]. The dimension of H_3 is $H \times W \times 1$. H and W represent the height and width of the input image.

The label generation processes of the kernel map, scale map, and surrounding map are shown in Fig. 5. The text kernel map is shrunk from the text map, and the shrinkage is calculated based on the area and perimeter of the instance. It can be described in detail as Algorithm 1.

B. Focus Entirety Module

Segmentation-based approaches are a case of the bottom-up method, which focuses heavily on local information and is susceptible to noise interference. According to this, a focus entirety module (FEM) is proposed to extract instance-level features and model texts in a top-down scheme that reduces the influence of noises. It assigns consistent entirety information to pixels within the same instance to improve their cohesion and emphasizes the scale of instances, enabling the model to distinguish varying scale texts effectively. The structure of FEM is as follows:

$$\mathrm{W}_1 = \mathrm{ReLU}_{\mathrm{BN}}(\mathrm{Conv}_{3\times3,64}(\mathrm{F_f})), \tag{4}$$

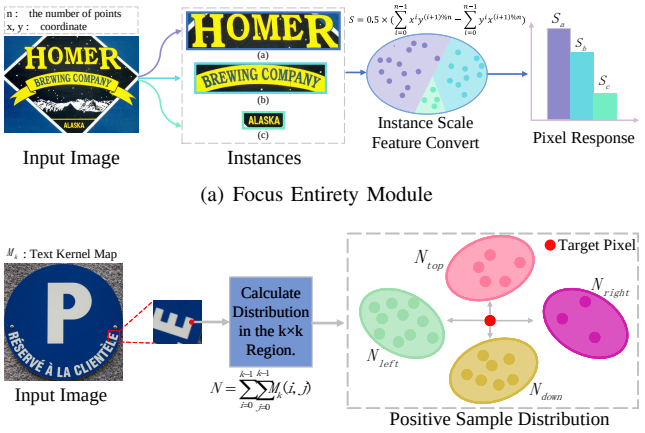
$$\mathrm{W}_2 = \mathrm{ReLU}_{\mathrm{BN}}(\mathrm{ConvT}_{3\times3,64}(\mathrm{W}_1)), \tag{5}$$

$$\mathrm{W}_3 = \mathrm{ReLU}(\mathrm{ConvT}_{3\times3,\mathrm{k}}(\mathrm{W}_2)), \tag{6}$$

where W_3 is a tensor with size of $H \times W \times 1$. It generates a scale map M_{sc}, which is defined as the area of the corresponding kernel (computed by the Algorithm 1). As shown in Fig. 3 (a), it focuses on the scale of the instance.

$$M_{sc}^i = \begin{cases} S^j, & \text{if} \quad i \in Kernel^j, \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

where i, S^j and $Kernel^j$ represent the ith pixel, the area of jth kernel and the jth kernel. FEM converts the instance scale feature and injects it into the pixel. Pixels belonging to different instances enjoy different response values. The specific generation process of M_{sc} is shown in Algorithm 2. As shown in Fig. 4, existing segmentation-based methods reconstruct text instances based on pixel-level knowledge, which lacks instance-level information. Unlike other methods, FEM infuses the model with information about the scale of the instance to help the pixel determine its attribution.



(a) Focus Entirety Module

(b) Perceive Environment Module

(c) An example of FEM

(d) An example of PEM

Fig. 3. The visualization of FEM and PEM. (a) FEM focuses on the scale of instance, the activation value of pixels belonging to large-scale is high. (b) PEM perceives the positive distribution of surroundings. The larger the positive sample the larger the label value. (c) The kernel regions are labeled in orange in the left image. Different instances are marked with a distinct color, and the value is the area of the corresponding instance. (d) The kernel regions are labeled in orange in the left image. The target pixel is marked with red. Its four surrounding map value is generated by the positive pixel number of the purple, green, grey, and blue region.

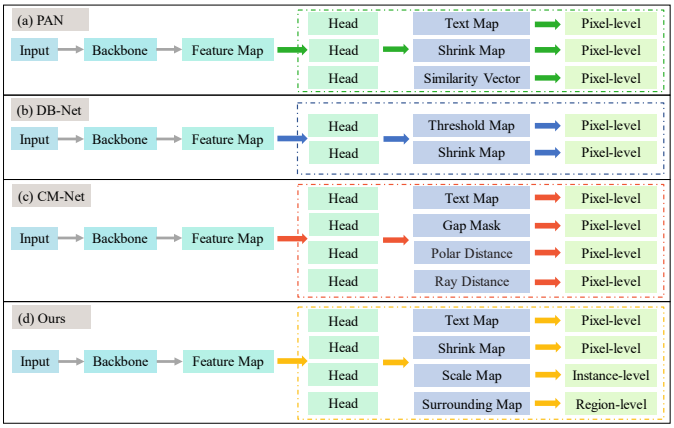


Fig. 4. The comparison with the overall pipeline of other advanced methods. It provides a comprehensive comparison that describes the hierarchy of features learned by each method.

C. Perceive Environment Module

As exhibited in Fig. 4, existing methods usually focus on information of isolated pixels, such as the probability of being text, kernel [9], the similarity vector [31], the distance to the text boundary [2], and the threshold map [10]. However, the

Algorithm 2: Scale Map and Surrounding Map Label Generation

Data: text kernel map M_k, area of text kernel instance S, environmental perception range μ, width W and height H, minimum area threshold A_{min}
Result: surrounding map M_{sr}, scale map M_{sc}

initializing $M_{sr} \in \mathbb{R}^{W,H,4}$ and $M_{sc} \in \mathbb{R}^{W,H}$;
for lth kernel instance *in* M_k **do**
  $\sigma \leftarrow$ area of lth kernel instance;
  if $\sigma > A_{min}$ **then**
    drawing shrink-mask$_k$ on M_{sc} with value σ;
  end
end
for lth pixel$_l^{i,j}$ *in* input image **do**
  for nth $\mathrm{M_{sr}}$ *on* pixel$_l^{i,j}$ **do**
    $(\theta_x^n, \theta_y^n) \leftarrow$ nth offset
    current position $(\rho_{\mathrm{x}}^{\mathrm{l,n}}, \rho_{\mathrm{y}}^{\mathrm{l,n}}) \leftarrow$ $(i,j) + (\theta_x^n, \theta_y^n)$;
    $\alpha \leftarrow \mathrm{clip}(\rho_x^{l,n} - (\mu+1)/2,\ 0,\ W)$;
    $\beta \leftarrow \mathrm{clip}(\rho_x^{l,n} + (\mu+1)/2,\ 0,\ W)$;
    $\varphi \leftarrow \mathrm{clip}(\rho_y^{l,n} - (\mu+1)/2,\ 0,\ H)$;
    $\omega \leftarrow \mathrm{clip}(\rho_y^{l,n} + (\mu+1)/2,\ 0,\ H)$;
    $M_{sr}^{i,j,n} \leftarrow \sum_{m=\alpha}^{\beta} \sum_{v=\varphi}^{\omega} (M_k(m,v))$;
  end
end

local textures of some objects in natural scenes are highly similar to the text. Focusing only on the pixel information is prone to misjudgment. The visual system tends to rely on surrounding objects to recognize complex objects. Hence, we propose a PEM to provide region-level information about the environment to enhance the model's ability to understand the relative position of pixels in the instance. It effectively improves the accuracy of scoring ambiguous pixels. The PEM treats the kernel pixels as positive samples to generate surrounding map M_{sr}, representing the number of positive samples in the $k \times k$ region in four directions (shown in Fig. 3 (b)). It helps the model differentiate text and kernel features, improving the incomplete kernel semantics. The specific generation process of M_{sr} is shown in the Algorithm. 2. The structure of PEM is as follows:

$$\mathrm{W}_1 = \mathrm{ReLU_{BN}}(\mathrm{Conv}_{3\times3,64}(\mathrm{F_f})), \tag{8}$$

$$\mathrm{W}_2 = \mathrm{ReLU_{BN}}(\mathrm{ConvT}_{3\times3,64}(\mathrm{W}_1)), \tag{9}$$

$$\mathrm{W}_3 = \mathrm{ReLU}(\mathrm{ConvT}_{3\times3,\mathrm{k}}(\mathrm{W}_2)), \tag{10}$$

where W_3 is a tensor with size of $H \times W \times 4$. When approaching the kernel boundary in a particular direction, the value of the surrounding map corresponding to that direction gradually decreases, which assists in determining the relative position of pixels within the text instance.

D. Optimization Function

In this paper, the proposed FEPE determines four predictions by kernel prediction layer, text prediction layer, focus entirety module, and perceive environment module (as shown in Fig. 2). A multi-task loss $\mathcal{L}$ is designed to optimize the proposed method. It includes four loss functions that text segmentation loss $\mathcal{L}_t$, kernel segmentation loss $\mathcal{L}_k$, surrounding map prediction loss $\mathcal{L}_{su}$, scale map prediction loss $\mathcal{L}_{sc}$, which supervise the corresponding features during training.

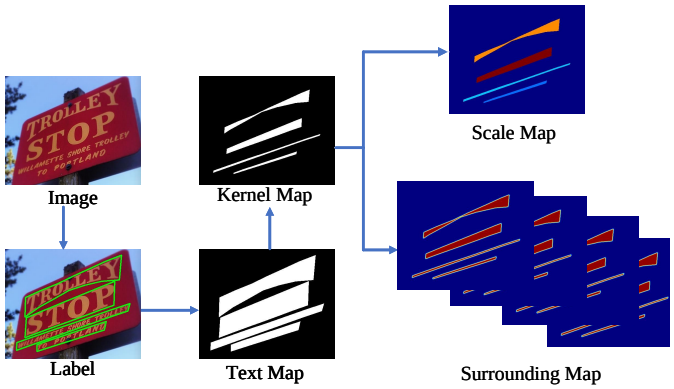


Fig. 5. The generation process of the text map, kernel map, scale map, and surrounding map used in the experiments.

$$\mathcal{L} = \lambda_1 \mathcal{L}_k + \lambda_2 \mathcal{L}_t + \lambda_3 \mathcal{L}_{su} + \lambda_4 \mathcal{L}_{sc}, \tag{11}$$

where λ_1, λ_2, λ_3 and λ_4 are the corresponding coefficients of L_k, L_t, L_{su}, and L_{sc}.

1) Kernel Segemention Loss: For kernel segmentation, binary cross-entropy (BCE) loss is utilized for supervision and is often used in binary classification problems. To alleviate the imbalance of positive and negative samples, hard negative mining is adopted in BCE loss:

$$\mathcal{L}_k = \sum_{i \in S} -K_i^y \times log(K_i^x) - (1 - K_i^y) \times log(K_i^x), \tag{12}$$

where S, K^x, and K^y are the selected training set, prediction kernel map, and ground truth of kernel. The sample ratio between positive and negative samples is 1:3.

2) Text Segemention Loss: Dice loss is commonly used for segmentation tasks. We adopt it to supervise the text area which is much smaller than the background. The $\mathcal{L}_t$ can be describe as follows:

$$\mathcal{L}_t = 1 - \frac{2 \times \sum(T^y \times T^x)}{\sum T^y + \sum T^x + \varepsilon}, \tag{13}$$

where T^y, T^x are the prediction and ground truth of the text map. ε is a minimal value used to avoid the denominator being 0, which is set to 10^{-6}.

3) Regression Loss: For the surrounding map and scale map, the ratio loss $\mathcal{L}_{ratio}$ [30] is used to optimize them. It can be described as follows:

$$\mathcal{L}_{ratio}(X, Y) = log\frac{max(X, Y)}{min(X, Y)}, \tag{14}$$

where X and Y are the prediction and ground truth, respectively. $\mathcal{L}_{su}$ and $\mathcal{L}_{sc}$ are based $\mathcal{L}_{ratio}$ which can be defined as follows:

$$\mathcal{L}_{su}(X_{su}, Y_{su}) = \mathcal{L}_{ratio}(X_{su}, Y_{su}), \tag{15}$$

TABLE I
ABLATION STUDY ON THE EFFECT OF FEM AND PEM ON DETECTION PERFORMANCE ON THE ICDAR2015 AND MSRA-TD500. "TEXT" REPRESENTS THE TEXT PREDICTION LAYER.

Backbone	Text	FEM	PEM	MSRA-TD500			ICDAR2015		
				Precision	Recall	F-measure	Precision	Recall	F-measure
ResNet18	×	×	×	79.4	76.8	78.1	87.7	75.2	81.0
	✓	×	×	80.1	77.0	78.5	89.2	74.6	81.3
	×	✓	×	84.1	80.8	82.4	88.4	78.3	83.0
	×	×	✓	82.9	79.7	81.3	88.2	78.5	83.1
	✓	✓	×	87.0	79.4	83.0	88.5	77.1	82.4
	✓	×	✓	86.2	79.2	82.5	88.0	78.5	83.0
	✓	✓	✓	87.7	80.6	84.0	87.3	79.4	83.2
ResNet50	×	×	×	85.8	79.9	82.7	87.5	79.5	83.3
	✓	×	×	86.2	82.3	84.2	89.3	78.3	83.4
	×	✓	×	86.9	83.3	85.1	88.2	80.5	84.1
	×	×	✓	89.0	82.0	85.3	88.2	80.4	84.1
	✓	✓	×	90.2	80.1	85.0	87.1	81.1	84.0
	✓	×	✓	89.2	82.0	85.4	88.0	80.0	83.8
	✓	✓	✓	88.1	83.5	85.8	88.5	80.4	84.2

TABLE II
ABLATION STUDY ON THE EFFECT OF FEM AND PEM ON DETECTION PERFORMANCE ON THE TOTAL-TEXT AND CTW1500. "TEXT" REPRESENTS THE TEXT PREDICTION LAYER.

Backbone	Text	FEM	PEM	Total-Text			CTW1500		
				Precision	Recall	F-measure	Precision	Recall	F-measure
ResNet18	×	×	×	86.0	76.4	80.9	81.9	79.4	80.6
	✓	×	×	87.2	78.3	82.5	82.6	80.8	81.7
	×	✓	×	85.2	80.4	82.7	83.4	80.6	82.0
	×	×	✓	87.1	78.9	82.8	83.8	80.9	82.3
	✓	✓	×	87.6	78.6	82.9	84.3	81.4	82.9
	✓	×	✓	87.1	79.1	82.9	83.7	81.4	82.6
	✓	✓	✓	89.4	78.8	83.7	85.1	81.6	83.3

$$\mathcal{L}_{sc}(X_{sc}, Y_{sc}) = \mathcal{L}_{ratio}(X_{sc}, Y_{sc}), \tag{16}$$

where X_{su} and Y_{su} are the prediction and label of the surrounding map, respectively. X_{sc} and Y_{sc} represent the prediction and label of the scale map, respectively.

IV. EXPERIMENT

In this section, we introduce the datasets. Then, the ablation studies are conducted on four public benchmarks to prove the superiority of the proposed method. Next, FEPE is compared with SOTA methods on different public benchmarks. Finally, we demonstrate the robustness of the method, and its shortcomings are also analyzed.

A. Datasets

CTW1500 [37] is a text dataset includes long curved text. It contains 1,000 training images and 500 testing images. Each text instance is labeled with 14 points.

ICDAR2015 [38] contains many samples from supermarkets, and many of these examples have low-resolution problems. Each instance is labeled by a quadrilateral consisting of four points. It contains 1500 images.

Total-Text [39] contains not only a large amount of horizontal text and multi-directional text but also a large amount of irregularly shaped text. The training and test sets have 1255 and 300 images, respectively.

SynthText [40] is a synthetic text dataset that includes 800,000 images for pre-training. It is generally used to pre-train the model to improve its performance.

ICDAR2017 MLT [41] is a multilingual text dataset. It consists of 7,200 training images, 1,800 validation images, and 9,000 testing images in nine languages.

MSRA-TD500 [42] is a Chinese-English bilingual scene text dataset with line-level annotations. We follow previous papers to utilize HUST-TR400 [43] for training.

B. Implementation Details

ResNet with deformable convolution and Feature Pyramid Network (FPN) are selected as the backbone. We choose two pre-training strategies: (1) Pretraining on ICDAR2017MLT for 400 epochs. (2) Pretraining on SynthText for four epochs. Afterward, the model is fine-tuned for 1,200 epochs. During the training phase, the batch size and initial learning rate are set to 16 and 0.007, respectively. The stochastic gradient descent (SGD) is used to train the model, while the weight decay and momentum are set to 0.0001 and 0.9, respectively. We use the "poly" strategy to adjust the learning rate, where the current learning rate is equal to the initial learning rate multiplied by $(1-\frac{iter}{max\_iter})^{power}$, and the power is set to 0.9. Slight random rotation, random cropping, and random flipping are used for data augmentation. All input images are resized to 640×640 during training. We evaluated the detection results following the metrics used in DBNet. During the inference stage, the prediction of the kernel map is binarized, and each kernel instance is obtained through contour extraction. Then, each text kernel expands a specific distance $D' = \frac{A' \times r'}{L'}$ to generate the text instance. r', A', and L' represent the expand

TABLE III
ABLATION STUDY ON THE IMPACT OF k ON DETECTION PERFORMANCE ON THE ICDAR2015 AND MSRA-TD500. k REPRESENTS THE AREA PERCEIVED BY PEM IN THE RANGE OF $k \times k$.

	Kernel	MSRA-TD500				ICDAR2015			
		Precision	Recall	F-measure	FPS	Precision	Recall	F-measure	FPS
FEPE with	3×3	87.6	77.8	82.4	62	88.0	78.5	83.0	48
	5×5	87.0	79.4	83.0	62	88.9	77.8	83.0	48
	7×7	84.7	77.5	81.6	62	89.0	77.4	82.8	48

TABLE IV
ABLATION STUDY ON THE IMPACT OF k ON DETECTION PERFORMANCE ON THE TOTAL-TEXT AND CTW1500. k REPRESENTS THE AREA PERCEIVED BY PEM IN THE RANGE OF $k \times k$.

	Kernel	Total-Text				CTW1500			
		Precision	Recall	F-measure	FPS	Precision	Recall	F-measure	FPS
FEPE with	3×3	88.0	77.6	82.4	50	84.5	80.5	82.5	55
	5×5	87.1	79.1	82.9	50	83.7	81.4	82.6	55
	7×7	87.5	77.3	82.1	50	82.9	81.5	82.2	55

TABLE V
THE QUALITATIVE ANALYSIS OF WHETHER THE MODEL ENHANCEMENT IS DUE TO THE EXTRA SUPERVISION, WHERE 'KERNEL', 'SCALE', AND 'SURROUNDING' REPRESENT THE KERNEL MAP, SCALE MAP, AND SURROUNDING MAP.

Baseline	PEM	FEM	P	R	F
Kernel	-	-	79.4	76.8	78.1
Kernel	Kernel	-	81.4	79.2	80.3
Kernel	Scale	-	87.0	79.4	83.0
Kernel	-	Kernel	80.4	74.6	77.5
Kernel	-	Surrounding	86.2	79.2	82.5

TABLE VI
THE DETECTION PERFORMANCE OF FEPE WITH DIFFERENT PRE-TRAINING CONDITIONS ON FOUR PUBLIC BENCHMARKS.

Datasets	Ext.	P	R	F
TotalText	None	89.4	78.7	83.7
	SynthText	90.8	79.5	84.8
	ICDAR2017	89.2	79.2	83.9
MSRA-TD500	None	87.7	80.6	84.0
	SynthText	89.4	82.8	86.0
	ICDAR2017	93.6	85.4	89.3
CTW1500	None	85.1	81.6	83.3
	SynthText	88.0	83.0	85.5
	ICDAR2017	89.0	82.2	85.5
ICDAR2015	None	88.0	78.5	83.0
	SynthText	87.3	79.4	83.2
	ICDAR2017	89.9	79.7	83.5

ratio, area, and perimeter of the kernel. The coefficients of the loss λ_1, λ_2, λ_3 and λ_4 are set to 6, 3, 1, and 0.5, respectively.

C. Ablation Study

The ablation study is conducted on four public benchmarks to show the effectiveness of the proposed PEM and FEM. All models are trained without pre-training.

1) Effectiveness of the FEM: The proposed FEM assigns consistent entirety information to pixels within the same instance to improve their cohesion and emphasizes the scale of instances to which pixels belong, enabling the model to distinguish varying scale texts effectively. Extensive experiments have proved the superiority of the proposed FEM. As seen in Table I, the proposed FEM brings a 4.3% and 2.0% performance improvement on MSRA-TD500 and ICDAR2015 when ResNet18 is adopted as the backbone. For ResNet50, the improvement of the method is 2.4% and 0.8%, respectively. In addition, when adopting ResNet18 as the backbone, FEM yields about 1.8% and 1.4% improvement on Total-Text and CTW-1500, respectively. The above experiment results demonstrate the superiority of the proposed FEM. As shown in Fig. 6, we display the prediction of the scale map that the redder means a higher value. As we can see, the larger instance obtains the higher value, which demonstrates the proposed FEM modeling the instance feature successfully. Moreover, the model's predictions for the scale map and kernel map are relatively consistent.

2) Influence of the PEM: As mentioned above, PEM extracts region-level features and encourages the model to focus on the distribution of positive samples in the vicinity of a pixel, which perceives environment information. It treats the kernel pixels as positive samples and helps the model differentiate text and kernel features, improving the incomplete kernel semantics. A series of experiments are conducted on MSRA-TD500, Total-Text, CTW1500, and ICDAR2015 datasets to validate the superiority of the proposed PEM. As shown in Table I, the proposed PEM improved F-measure by 3.2% and 2.1% on MSRA-TD500 and ICDAR2015 when ResNet18 is used as the backbone. Moreover, the method brings 2.6% and 0.8% performance improvements when using ResNet50 to extract features. For Total-Text and CTW1500, when adopting ResNet18 as the backbone, the proposed PEM achieves 1.9% and 1.7% performance gains, respectively, as shown in Table II. The above experiments demonstrate that PEM can help the proposed model to improve detection performance effectively. As shown in Fig. 7, we show the prediction of the left surrounding map, corresponding ground truth, which actually models regional features of texts. In addition, the prediction of the left surrounding map and the kernel map are consistent.

3) Influence of the choice of k: Table III and Table IV show the results under different $k \times k$ regions to validate the impact of k on detection performance on ICDAR2015, MSRA-TD500, Total-Text and CTW1500, respectively. When k is set to 5, the model achieves optimal performance, and we set it up like this in subsequent experiments. When the choice of

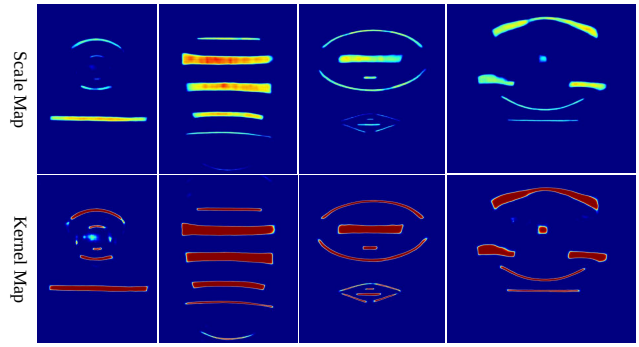


Fig. 6. The visualization of the prediction of the scale map and the corresponding kernel map. For the scale map, a redder color means a higher value.

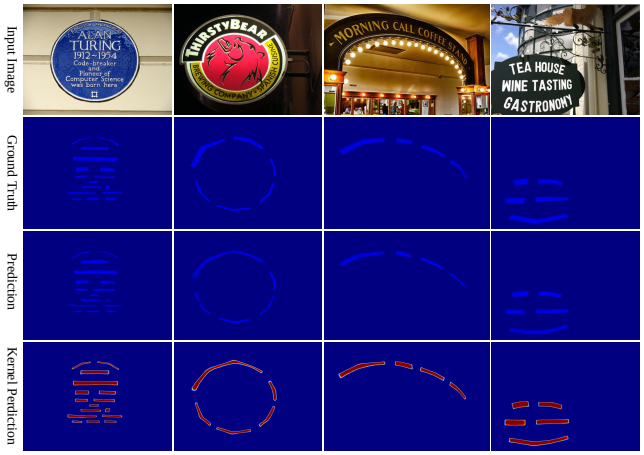
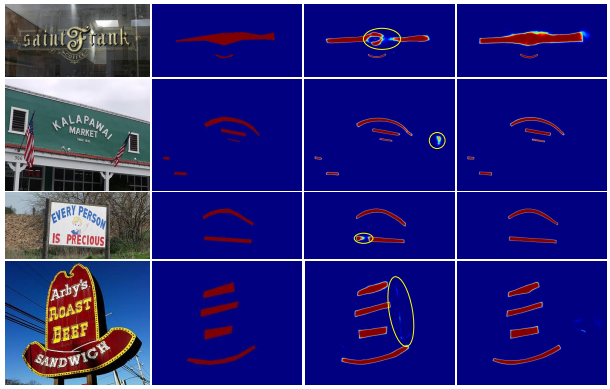


Fig. 7. Visualization of left surrounding maps predictions and ground truths, as well as kernel maps predictions.

k is too large, it will occupy too much weight of the model, and when it is too small, it will affect the model's ability to perceive the environment.

4) The qualitative analysis of the extra supervision.: To qualitatively analyze whether the model enhancement is due to the extra supervision, we perform corresponding experiments on MSRA-TD500. We change the label of FEM and PEM to kernel map. As shown in Table V, the extra supervision for the model is not always helpful for the model. Compared with using kernel maps, the scale map and surrounding map are better, which shows that the effectiveness of FEM and PEM is not because of extra supervision.

5) Influence of the pre-training: Pre-training using additional datasets has a significant impact on the detection results. The ICDAR2015 dataset is minimally affected by pre-training, resulting in only slight improvements of 0.2% and 0.5% after pre-training on SynthText and MLT, respectively. In contrast, the MSRA-TD500 dataset is the most affected by pre-training. After pre-training on SynthText, the performance improved by 2%, while pre-training on MLT resulted in an F-measure improvement of 5.3%. For the Total-Text dataset, pre-training on SynthText was more effective than pre-training on MLT, with performance improvements of 2.8% and 2%,



(a) Input images (b) Ground truth (c) Prediction of baseline (d) Prediction of FEPE

Fig. 8. The visual comparison between the ground truth (text kernel maps) and the predicted results. The input images are presented in (a). The ground truth is illustrated in (b), while (c) and (d) show the prediction results of the baseline (only predict text kernel maps and text maps) and the proposed FEPE, respectively. The baseline method faces difficulties correctly identifying long text regions with large gaps, which are misjudged as non-text regions in (c). Moreover, patterns resembling text texture are mistakenly classified as text regions, as shown in (c).

respectively. Pre-training on both datasets resulted in gains of 1.9% for CTW1500. Based on the experimental results, we can conclude that MLT yields more significant performance improvement for multi-directional text datasets, while SynthText is better suited for datasets that contain a large amount of irregular-shaped text.

6) Visual comprasion: We visualize and compare the detection performance of the proposed method with the baseline (only predict text kernel maps and text maps), as seen in Fig. 8. The baseline model misidentifies an instance as two text instances when confronted with texts containing large gaps (Fig. 8 first line). The proposed FEM improves the cohesiveness of pixels within the same text instance, thus exhibiting a better performance in this particular case. Moreover, the baseline model has a major drawback of misclassifying certain patterns that have similar textures with text. This issue is alleviated by the PEM, which confirms whether a given pixel belongs to text by emphasizing its perceptual surroundings.

D. Comparison with State-of-the-Art Methods

The comparison with SOTA approaches is built on four benchmarks containing various text types. ICDAR2015 and MSRA-TD500 are word-level and line-level multi-directional text datasets. Total-Text and CTW1500 are word-level and line-level irregular-shaped text datasets. The advantages of FEPE are further analyzed through comparison.

Evaluation on MSRA-TD500. MSRA-TD500 is a multi-directional line-level labeled text dataset that contains Chinese and English. During the inference stage, we resize the short side of the input images to 736. As we can see from Table. VII, the proposed FEPE achieves 86.0% (pre-trained on SynthText) and 89.5% (pre-trained on MLT) for F-measure when ResNet18 is used as backbone. Moreover, the method equipped with ResNet50 brings a 2.1% (pre-

TABLE VII
COMPARISON WITH EXISTING ADVANCED APPROACHES ON THE MSRA-TD500 AND CTW1500 DATASETS. "RED", "BLUE" AND "GREEN" REPRESENT THE OPTIMAL, SUB-OPTIMAL AND THE THIRD BEST PERFORMANCE, RESPECTIVELY.

Methods	Venue	Ext.	Backbone	MSRA-TD500				CTW1500			
				P	R	F	FPS	P	R	F	FPS
PAN [31]	ICCV'19	Synth	ResNet18	84.4	83.8	84.1	30.2	86.4	81.2	83.7	39.8
ContourNet [44]	CVPR'20	-	ResNet50	-	-	-	-	84.1	83.7	83.9	4.5
DRRG [45]	CVPR'20	MLT	VGG16	88.1	82.3	85.1	-	85.9	83.0	84.5	-
CTNet [12]	NeurIPS'21	Synth	ResNet18	90.0	82.5	86.1	34.8	88.3	79.9	83.9	40.8
FEMP [46]	TMM'21	MLT	ResNet50	86.0	83.4	84.7	1.6	88.5	82.9	**85.6**	1.4
PCR [20]	CVPR'21	MLT	DLA34	90.8	83.5	87.0	-	87.2	82.3	84.7	-
TextBPN [47]	ICCV'21	Synth	ResNet50	85.4	80.7	83.0	12.7	87.8	81.5	84.5	12.2
TextBPN [47]	ICCV'21	MLT	ResNet50	86.6	84.5	85.6	12.3	86.5	83.6	85.0	12.2
LPAP [48]	TOMM'22	Synth	ResNet50	87.9	77.7	82.5	-	84.6	80.3	82.4	-
TextDCT [49]	TMM'22	Synth	ResNet50	-	-	-	-	85.3	85.0	85.1	17.2
ASTD [5]	TMM'22	-	ResNet101	-	-	-	-	87.2	81.7	84.4	-
LEMNet [50]	TMM'22	-	ResNet50	85.6	84.8	85.2	-	86.6	83.8	85.2	-
ADNet [1]	TMM'22	Synth	ResNet50	92.0	83.2	87.4	-	88.2	83.1	**85.6**	-
CMNet [30]	TIP'22	-	ResNet18	89.9	80.6	85.0	41.7	86.0	82.2	84.1	**50.3**
PAN++ [51]	TPAMI'22	Synth	ResNet18	89.6	86.3	**87.9**	22.6	87.1	81.1	84.0	36.0
KPN [52]	TNNLS'22	MLT	ResNet50	-	-	-	-	84.4	84.2	84.3	16
ZTD [7]	TNNLD'23	Synth	ResNet18	91.6	82.4	86.8	**59.2**	88.4	80.2	84.1	**76.9**
FS [53]	TIP'23	-	ResNet18	90.0	80.4	84.9	35.5	84.6	77.7	81.0	35.2
FS [53]	TIP'23	-	ResNet50	89.3	81.6	85.3	25.4	85.3	82.5	83.9	25.1
DBNet++ [11]	TPAMI'23	Synth	ResNet18	87.9	82.5	85.1	**55**	84.3	81.0	82.6	49
DBNet++ [11]	TPAMI'23	Synth	ResNet50	91.5	83.3	87.2	29	87.9	82.8	85.3	26
LeafText [4]	TMM'23	Synth	ResNet50/18	92.1	83.8	86.1	-	87.1	83.9	**85.5**	-
FEPE	Ours	Synth	ResNet18	89.4	82.8	86.0	**62**	88.0	83.0	**85.5**	**55**
FEPE	Ours	MLT	ResNet18	93.8	85.6	**89.5**	**62**	89.0	82.2	**85.5**	**55**
FEPE	Ours	Synth	ResNet50	90.5	85.4	**88.0**	32	88.8	83.5	**86.0**	22

TABLE VIII
COMPARISON WITH EXISTING ADVANCED APPROACHES ON THE TOTAL-TEXT. "RED" AND "BLUE" REPRESENT THE OPTIMAL AND SUB-OPTIMAL PERFORMANCE, RESPECTIVELY.

Methods	Backbone	P	R	F	FPS
PSENet-1s [9]	ResNet50	84.0	78.0	80.9	3.9
TextSnake [27]	VGG16	82.7	74.5	78.4	-
Boundary [54]	ResNet50	85.2	82.2	84.3	-
DRRG [45]	VGG16	86.5	84.9	85.7	-
FCENet [21]	ResNet50	89.3	82.5	85.8	-
KPN [52]	ResNet50	88.0	82.3	85.1	22.7
PSE+STKM [55]	ResNet50	86.3	78.4	82.2	-
DB [10]	ResNet50	87.1	82.5	84.7	32
CM-Net [30]	ResNet18	88.5	81.4	84.8	**49.8**
PAN [31]	ResNet18	89.3	81.0	85.0	39.6
TextDCT [49]	ResNet50	87.2	82.7	84.9	15.1
ASTD [5]	ResNet101	85.4	81.2	83.2	-
CRAFT [23]	VGG16	87.6	79.9	83.6	-
OKR [56]	ResNet18	85.8	80.9	83.3	40.5
PAN++ [51]	ResNet18	89.9	81.0	85.3	38.3
NASK [57]	ResNet50	85.6	83.2	84.4	8.2
DBNet [10]	ResNet50	87.1	82.5	84.7	32
DBNet++ [11]	ResNet50	88.9	83.2	86.0	28
LeafText [4]	ResNet18	88.9	83.2	**87.3**	-
LPAP [48]	ResNet50	87.3	79.8	83.4	-
FEPE (Syn)	ResNet18	90.8	79.5	84.8	**50**
FEPE (Syn)	ResNet50	91.3	81.9	**86.4**	32

TABLE IX
COMPARISON WITH EXISTING ADVANCED APPROACHES ON THE ICDAR2015. "RED" AND "BLUE" REPRESENT THE OPTIMAL AND SUB-OPTIMAL PERFORMANCE, RESPECTIVELY.

Method	Backbone	P	R	F	FPS
EAST [16]	VGG16	83.6	73.5	78.2	13.2
PixelLink [25]	VGG16	85.5	82.0	83.7	-
PSE-1s [9]	ResNet50	86.9	84.5	85.7	1.6
TextSnake [27]	VGG16	84.9	80.4	82.6	1.1
Boundary [54]	ResNet50	88.1	82.2	85.0	-
FCENet [21]	ResNet50	90.1	82.6	86.2	-
KPN [52]	ResNet50	88.3	88.3	86.5	6.3
DBNet++ [11]	ResNet50	90.9	83.9	**87.3**	10
DBNet++ [11]	ResNet18	90.1	77.2	83.1	44
DBNet [10]	ResNet50	91.8	83.2	**87.3**	12
LOMO [59]	ResNet50	91.3	83.5	87.2	-
CM-Net [30]	ResNet18	86.7	81.3	83.9	34.5
PAN [31]	ResNet18	84.0	81.9	82.9	26.1
ZTD [7]	ResNet18	87.5	79.0	83.0	**48.3**
Spotter [60]	ResNet50	85.8	81.2	83.4	4.8
BiP-Net [61]	ResNet18	86.9	82.1	83.9	24.8
PAN++ [51]	ResNet50	91.4	83.9	**87.5**	12.6
ASTD [5]	ResNet101	88.8	82.6	85.6	-
LeafText [4]	ResNet50	88.9	82.3	86.1	-
LPAP [48]	ResNet50	88.7	84.4	86.5	-
FEPE (Syn)	ResNet18	87.3	79.4	83.2	**48**
FEPE (Syn)	ResNet50	89.8	84.9	**87.3**	12

trained on SynthText) improvement. The proposed approach significantly outperforms existing SOTA methods regarding both performance and speed. Benefiting from the FEM that strengthens the identification of features between instances at various scales, FEPE surpasses ADNet [1] by 0.3% on F-measure. Compared with DBNet++ [11], our approach has improved both speed and performance. As seen from Fig. 9 and Table VII, FEPE is particularly effective for detecting multi-directional long text instances.

Evaluation on Total-Text and CTW1500. The Total-Text and CTW1500 datasets contain lots of varying shape and orientation texts. During the testing stage, the short side of input images is resized to 800. As shown in Table VIII, our approach outperforms LPAP [48], ASTD [5], and TextDCT [49] by 3.0%, 3.2%, and 1.5%, respectively, while also maintaining a faster speed. The proposed PEM helps FEPE in perceiving the environment around a pixel to confirm whether the pixel is text or not. Even though LeafText [4] is

Fig. 9. Visualizations of various types of text detection results are presented, including horizontal text, rotated text, and irregular text. The first and second rows of samples are from ICDAR2015 and CTW1500, respectively, while the last two are from Total-Text and MSRA-TD500. The proposed method is able to handle text instances of arbitrary shapes effectively.

superior to ours, it needs complex post-processing and not mentation the speed, which limits it to apply in the real world significantly. Moreover, when adopting ResNet18 as the backbone, our method achieves competitive performance while maintaining fast speed. Unlike Total-Text, CTW1500 is a line-level annotated irregular text dataset. As shown in Table VII, FEPE achieves the F-measure of 85.5 % and 86.0% when adopting ResNet18 and ResNet50 as the backbone, which surpassing the existing SOTA method DBNet++ [11] by 2.9% and 0.7%, respectively, while maintaining a speed advantage. Even our approach using ResNet18 as the backbone is still superior to some existing SOTA methods using ResNet50. This further demonstrates the superiority of FEPE. We visualize some samples from Total-Text and CTW1500 in Fig. 9 to demonstrate the effectiveness of FEPE. Furthermore, in Fig. 10, we compare the visible results of our approach to SOTA methods. TextRay [58] is a regression method that fails to fit instances with particularly uneven aspect ratios accurately. FCENet [21] and PAN [31], which focus only on pixel information, incorrectly classify some patterns similar to text as text and have problems with adjacent text sticking in PAN [31]. Additionally, LPAP [48] misclassifies one text as two instances. Since FEPE focuses on instance-level features, it effectively addresses these problems and showcases the superiority of the proposed method.

Evaluation on ICDAR2015. This dataset contains images with complex backgrounds, low resolution, and dim lighting, making scene text detection challenging. The large variation in scale and multiple orientations are additional reasons for the difficulty in detecting instances. As shown in Table IX, when adopting ResNet50 as the backbone and resizing the short side to 1152, the proposed method achieves 89.8%, 84.9%, and 87.3% on precision, recall, and F-measure, respectively. The proposed FEPE surpasses the existing SOTA method LeafText [4] by 1.2% in terms of F-measure, even though LeafText uses a complex post-process without mentioning the speed. Moreover, the proposed method outperforms most existing SOTA approaches (such as KPN [52], LPAP [48], and FCENet [21]) on performance and speed. Although DBNet++ [11] achieves the same performance, mainly because it introduces an extra attention module that sacrifices speed. PAN++ [51] surpasses ours 0.2% in F-measure, which is mainly because it uses ICDAR2017-MLT to pre-train, but the proposed method uses SynthText. Using real datasets to pre-train generates better results than synthetic datasets. The objective evaluation metrics presented in Table IX and the visualization results in Fig. 9 effectively demonstrate that our method can cope with multi-directional texts.

E. Cross Dataset Text Detection

To show the shape robustness of the FEPE, we train it on one dataset and test it on another. Note that cross-train-test experiments adopt ResNet18 as the backbone. We divided the four datasets used in the experiments into two categories based on the annotation style (word-level or line-level). As

(a) vs. FCENet [21] (b) vs. TextRay [58] (c) vs. PAN [31] (d) vs. LPAP [48]

Fig. 10. Some visual comparisons with FCENet [21], TextRay [58], PAN [31], and LPAP [48]. The middle row displays the results for FCENet [21], PAN [31], DBNet [10], and DBNet++ [11], respectively. The top row is the labels corresponding to the images. The bottom row displays our detection results.

TABLE X
TWO GROUPS (WORD-LEVEL AND LINE-LEVEL) CROSS-DATASET EVALUATIONS, WHERE IC15, TOTAL, TD500, AND CTW REPRESENT ICDAR2015, TOTAL-TEXT, MSRA-TD500 AND CTW1500 DATASETS, RESPECTIVELY.

Training	Testing	Methods	P	R	F
IC15	Total	Textfield [29]	61.5	65.2	63.3
		CM-Net [30]	75.8	64.5	69.7
		FEPE(ours)	81.4	63.5	71.4
Total	IC15	Textfield [29]	77.1	66.0	71.1
		CM-Net [30]	76.5	68.1	72.1
		FEPE(ours)	82.9	72.5	77.3
TD500	CTW	Textfield [29]	75.3	70.0	72.6
		CM-Net [30]	77.2	69.7	72.8
		FEPE(ours)	85.3	74.8	79.7
CTW	TD500	Textfield [29]	85.3	75.8	80.3
		CM-Net [30]	85.8	77.1	81.2
		FEPE(ours)	85.5	86.3	80.7

shown in Tab. X, FEPE achieves 71.4% and 77.3% of F-measure when training on Total-Text and ICDAR2015 and testing on ICDAR2015 and Total-Text. Compared with the SOTA method CM-Net [30], the proposed FEPE surpasses 1.7% and 5.2% in terms of F-measure, which shows the generalization ability on word-level texts. On the line-level datasets, FEPE achieves 79.7% and 80.7% when training on MSRA-TD500 and CTW1500 and testing on CTW1500 and MSRA-TD500. It is also substantially superior to TextField [29]. Compared to the CM-Net [30], the FEPE substantially outperformed that test on CTW1500. It is slightly inferior to the test on MSRA-TD500. These experiments demonstrate that FEPE has excellent generalization for data of different shapes and that its data requirements are low compared to other methods.

F. Limitations

We demonstrate the superiority of the proposed FEM for instance-level feature extraction and the effectiveness of PEM for sensing the surrounding environment through ablation experiments. Also, the excellent performance on various datasets proves the advancedness of FEPE. In this phase, we further analyze the shortcomings and limitations of the FEPE. As shown in Fig. 11(a), four typical errors are selected for detailed analysis. The lack of FEPE's ability to detect vertical text in the figure is mainly because vertical text instances are too rare in the training set and even in life. The model does not have enough samples to learn the features of vertical text, which can be considered to compensate for this shortcoming in the subsequent dataset construction. The long text in Fig. 11(b) is truncated by an obstacle, resulting in one instance being misclassified by the model as two instances. This is the bottom-up approach of the segmentation method, which over-focuses on the underlying features and has an insufficient grasp of the holistic features of the instances. We can see in Fig. 11(c) that some patterns similar to the text texture are misclassified as text. As we can see from Fig. 11(d), the characters belonging to the same instance are wrongly divided into different instances due to different colors. On the contrary, characters of different text instances are classified as the same instance due to the same color. These problems arise mainly because the model focuses only on visual information, for the language features are unaware. It is our future work to alleviate these problems.

V. CONCLUSION

In this paper, an arbitrary-shaped scene text detector is proposed that consists of FEM and PEM. The former encour-

(a) Large amount of vertical texts.

(b) The long text is separated by a barrier in the middle.

(c) Patterns that are similar to the texture of the text.

(d) Text with different colors for the same instance character.

Fig. 11. Some limitations and drawbacks of the proposed FEPE include inadequate detection of vertical text, misclassification of the gaps in the long text as negative samples, and difficulties in accurately detecting text in texture and color interference. The left is our prediction in each group image, and the right is the corresponding ground truth. The incorrections in our results are used in yellow to mark.

ages the model to distinguish instances of different scales, enhance the sense of belonging of pixels to their respective instances, and increase the cohesion of pixels belonging to the same sample. The latter perceives the distribution of positive samples around each pixel to confirm whether the current pixel is a positive sample. The FEPE differs from existing segmentation-based methods which typically focus only on pixel-level information. The proposed FEM and PEM enable the model to learn instance-level and region-level information, thus partially compensating for the insufficient global information extraction of bottom-up methods. Extensive experiments prove that the proposed FEPE significantly surpasses existing SOTA methods on four public benchmarks. We will continue to explore the relationships of multi-level information of scene texts to structure text knowledge systems in the future.

REFERENCES

[1] Y. Qu, H. Xie, S. Fang, Y. Wang, and Y. ZhangSenior Member, "Adnet: Rethinking the shrunk polygon-based approach in scene text detection," *IEEE Transactions on Multimedia*, pp. 1–14, 2022.

[2] C. Yang, M. Chen, Y. Yuan, and Q. Wang, "Reinforcement shrink-mask for text detection," *IEEE Transactions on Multimedia*, 2022, early Access, doi: 10.1109/TMM.2022.3209022.

[3] B. Jiang, Z. Zhou, X. Wang, J. Tang, and B. Luo, "cmsalgan: Rgb-d salient object detection with cross-view generative adversarial networks," *IEEE Transactions on Multimedia*, vol. 23, pp. 1343–1353, 2021.

[4] C. Yang, M. Chen, Y. Yuan, and Q. Wang, "Text growing on leaf," *IEEE Transactions on Multimedia*, 2023, early Access, doi: 10.1109/TMM.2023.3244322.

[5] P. Dai, Y. Li, H. Zhang, J. Li, and X. Cao, "Accurate scene text detection via scale-aware data augmentation and shape similarity constraint," *IEEE Transactions on Multimedia*, vol. 24, pp. 1883–1895, 2022.

[6] Y. Wang, H. Xie, Z. Zha, Y. Tian, Z. Fu, and Y. Zhang, "R-net: A relationship network for efficient and accurate scene text detection," *IEEE Transactions on Multimedia*, vol. 23, pp. 1316–1329, 2020.

[7] C. Yang, M. Chen, Y. Yuan, and Q. Wang, "Zoom text detector," *IEEE Transactions on Neural Networks and Learning Systems*, 2023, early Access, doi: 10.1109/TNNLS.2023.3289327.

[8] X. Han, J. Gao, Y. Yuan, and Q. Wang, "Text kernel calculation for arbitrary shape text detection," *The Visual Computer*, pp. 1–14, 2023.

[9] W. Wang, E. Xie, X. Li, W. Hou, T. Lu, G. Yu, and S. Shao, "Shape robust text detection with progressive scale expansion network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9336–9345.

[10] M. Liao, Z. Wan, C. Yao, K. Chen, and X. Bai, "Real-time scene text detection with differentiable binarization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 11 474–11 481.

[11] M. Liao, Z. Zou, Z. Wan, C. Yao, and X. Bai, "Real-time scene text detection with differentiable binarization and adaptive scale fusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 1, pp. 919–931, 2023.

[12] T. Sheng, J. Chen, Z. Lian, and q. qzz, "Centripetaltext: An efficient text instance representation for scene text detection," *Advances in Neural Information Processing Systems*, vol. 34, pp. 335–346, 2021.

[13] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, pp. 91–99, 2015.

[14] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "Textboxes: A fast text detector with a single deep neural network," in *Thirty-first AAAI conference on artificial intelligence*, 2017.

[15] M. Liao, B. Shi, and X. Bai, "Textboxes++: A single-shot oriented scene text detector," *IEEE transactions on image processing*, vol. 27, no. 8, pp. 3676–3690, 2018.

[16] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang, "East: an efficient and accurate scene text detector," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 5551–5560.

[17] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[18] M. Liao, Z. Zhu, B. Shi, G.-s. Xia, and X. Bai, "Rotation-sensitive regression for oriented scene text detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[19] P. He, W. Huang, T. He, Q. Zhu, Y. Qiao, and X. Li, "Single shot text detector with regional attention," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 3047–3055.

[20] P. Dai, S. Zhang, H. Zhang, and X. Cao, "Progressive contour regression for arbitrary-shape scene text detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2021, pp. 7393–7402.

[21] Y. Zhu, J. Chen, L. Liang, Z. Kuang, L. Jin, and W. Zhang, "Fourier contour embedding for arbitrary-shaped text detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3123–3131.

[22] Y. Liu, H. Chen, C. Shen, T. He, L. Jin, and L. Wang, "Abcnet: Real-time scene text spotting with adaptive bezier-curve network," in *proceedings of the IEEE conference on computer vision and pattern recognition*, 2020, pp. 9809–9818.

[23] Y. Baek, B. Lee, D. Han, S. Yun, and H. Lee, "Character region awareness for text detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9365–9374.

[24] S. Zhang, X. Zhu, J. Hou, C. Liu, C. Yang, H. Wang, and X. Yin, "Deep relational reasoning graph network for arbitrary shape text detection," in

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2020, pp. 9699–9708.

[25] D. Deng, H. Liu, X. Li, and D. Cai, "Pixellink: Detecting scene text via instance segmentation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[26] B. Shi, X. Bai, and S. Belongie, "Detecting oriented text in natural images by linking segments," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2550–2558.

[27] S. Long, J. Ruan, W. Zhang, X. He, W. Wu, and C. Yao, "Textsnake: A flexible representation for detecting text of arbitrary shapes," in *Proceedings of the European conference on computer vision*, 2018, pp. 20–36.

[28] P. Lyu, C. Yao, W. Wu, S. Yan, and X. Bai, "Multi-oriented scene text detection via corner localization and region segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2018.

[29] Y. Xu, Y. Wang, W. Zhou, Y. Wang, Z. Yang, and X. Bai, "Textfield: Learning a deep direction field for irregular scene text detection," *IEEE Transactions on Image Processing*, vol. 28, no. 11, pp. 5566–5579, 2019.

[30] C. Yang, M. Chen, Z. Xiong, Y. Yuan, and Q. Wang, "Cm-net: Concentric mask based arbitrary-shaped text detection," *IEEE Transactions on Image Processing*, pp. 2864–2877, 2022.

[31] W. Wang, E. Xie, X. Song, Y. Zang, W. Wang, T. Lu, G. Yu, and C. Shen, "Efficient and accurate arbitrary-shaped text detection with pixel aggregation network," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 8440–8449.

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[33] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 764–773.

[34] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable convnets v2: More deformable, better results," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2019, pp. 9308–9316.

[35] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[36] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*. pmlr, 2015, pp. 448–456.

[37] L. Yuliang, J. Lianwen, Z. Shuaitao, and Z. Sheng, "Detecting curve text in the wild: New dataset and new solution," *arXiv preprint arXiv:1712.02170*, 2017.

[38] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu *et al.*, "Icdar 2015 competition on robust reading," in *2015 13th International Conference on Document Analysis and Recognition*. IEEE, 2015, pp. 1156–1160.

[39] C. Ch'ng and C. Chan, "Total-text: A comprehensive dataset for scene text detection and recognition," in *2017 14th IAPR international conference on document analysis and recognition*, vol. 1. IEEE, 2017, pp. 935–942.

[40] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2315–2324.

[41] N. Nayef, F. Yin, I. Bizid, H. Choi, Y. Feng, D. Karatzas, Z. Luo, U. Pal, C. Rigaud, J. Chazalon *et al.*, "Icdar2017 robust reading challenge on multi-lingual scene text detection and script identification-rrc-mlt," in *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, vol. 1. IEEE, 2017, pp. 1454–1459.

[42] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, "Detecting texts of arbitrary orientations in natural images," in *2012 IEEE conference on computer vision and pattern recognition*. IEEE, 2012, pp. 1083–1090.

[43] C. Yao, X. Bai, and W. Liu, "A unified framework for multioriented text detection and recognition," *IEEE Transactions on Image Processing*, vol. 23, no. 11, pp. 4737–4749, 2014.

[44] Y. Wang, H. Xie, Z.-J. Zha, M. Xing, Z. Fu, and Y. Zhang, "Contournet: Taking a further step toward accurate arbitrary-shaped scene text detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 753–11 762.

[45] S. Zhang, X. Zhu, J. Hou, C. Liu, C. Yang, H. Wang, and X. Yin, "Deep relational reasoning graph network for arbitrary shape text detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9699–9708.

[46] S. Zhang, Y. Liu, L. Jin, Z. Wei, and C. Shen, "Opmp: An omnidirectional pyramid mask proposal network for arbitrary-shape scene text detection," *IEEE Transactions on Multimedia*, vol. 23, pp. 454–467, 2021.

[47] S. Zhang, X. Zhu, C. Yang, H. Wang, and X. Yin, "Adaptive boundary proposal network for arbitrary shape text detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2021, pp. 1305–1314.

[48] Z. Fu, H. Xie, S. Fang, Y. Wang, M. Xing, and Y. Zhang, "Learning pixel affinity pyramid for arbitrary-shaped text detection," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 19, no. 1s, pp. 1–24, 2023.

[49] Y. Su, Z. Shao, Y. Zhou, F. Meng, H. Zhu, B. Liu, and R. Yao, "Textdct: Arbitrary-shaped text detection via discrete cosine transform mask," *IEEE Transactions on Multimedia*, pp. 1–14, 2022.

[50] M. Xing, H. Xie, Q. Tan, S. Fang, Y. Wang, Z. Zha, and Y. Zhang, "Boundary-aware arbitrary-shaped scene text detector with learnable embedding network," *IEEE Transactions on Multimedia*, vol. 24, pp. 3129–3143, 2021.

[51] W. Wang, E. Xie, X. Li, X. Liu, D. Liang, Z. Yang, T. Lu, and C. Shen, "Pan++: Towards efficient and accurate end-to-end spotting of arbitrarily-shaped text," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 5349–5367, 2022.

[52] S. Zhang, X. Zhu, J. Hou, C. Yang, and X. Yin, "Kernel proposal network for arbitrary shape text detection," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–12, 2022.

[53] F. Wang, X. Xu, Y. Chen, and X. Li, "Fuzzy semantics for arbitrary-shaped scene text detection," *IEEE Transactions on Image Processing*, vol. 32, pp. 1–12, 2023.

[54] H. Wang, P. Lu, H. Zhang, M. Yang, X. Bai, Y. Xu, M. He, Y. Wang, and W. Liu, "All you need is boundary: Toward arbitrary-shaped text spotting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 12 160–12 167.

[55] Q. Wan, H. Ji, and L. Shen, "Self-attention based text knowledge mining for text detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 5983–5992.

[56] H. Ma, C. Yang, Y. Yuan, and Q. Wang, "Optimal kernel for real-time arbitrary-shaped text detection," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023, pp. 1–5.

[57] M. Cao, C. Zhang, D. Yang, and Y. Zou, "All you need is a second look: Towards arbitrary-shaped text detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 2, pp. 758–767, 2022.

[58] F. Wang, Y. Chen, F. Wu, and X. Li, "Textray: Contour-based geometric modeling for arbitrary-shaped scene text detection," in *Proceedings of the 28th ACM International Conference on Multimedia*, ser. MM '20. Association for Computing Machinery, 2020, p. 111–119.

[59] C. Zhang, B. Liang, Z. Huang, M. En, J. Han, E. Ding, and X. Ding, "Look more than once: An accurate detector for text of arbitrary shapes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 552–10 561.

[60] M. Liao, P. Lyu, M. He, C. Yao, W. Wu, and X. Bai, "Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 2, pp. 532–548, 2021.

[61] C. Yang, M. Chen, Y. Yuan, and Q. Wang, "Bip-net: Bidirectional perspective strategy based arbitrary-shaped text detection network," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 2255–2259.

Xu Han received the B.E. degree in information and computing sciences from Northeast Agricultural University, Harbin, China, in 2021

He is currently pursuing the Ph.D. degree with the School of Computer Science and the School of Artificial Intelligence, Optics and Electronics (iOPEN). His research interests include computer vision, pattern recognition and text detection.

Junyu Gao received the B.E. degree and the Ph.D. degree in computer science and technology from the Northwestern Polytechnical University, Xi'an 710072, Shaanxi, P. R. China, in 2015 and 2021 respectively. He is currently an associate professor with the School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision and pattern recognition..

Chuang Yang received the B.E. degree in automation and the M.E. degree in control engineering from Civil Aviation University of China, Tianjin, China, in 2017 and 2020 respectively. He is currently working toward the Ph.D. degree in the School of Computer Science and School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision and machine learning.

Yuan Yuan (M'05-SM'09) is currently a Full Professor with the School of Computer Science and the School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an, China. She has authored or co-authored over 150 papers, including about 100 in reputable journals, such as the IEEE TRANSACTIONS and PATTERN RECOGNITION, as well as the conference papers in CVPR, BMVC, ICIP, and ICASSP. Her current research interests include visual information processing and image / video content analysis.

Qi Wang (M'15-SM'15) received the B.E. degree in automation and the Ph.D. degree in pattern recognition and intelligent systems from the University of Science and Technology of China, Hefei, China, in 2005 and 2010, respectively.

He is currently a Professor with the School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision, pattern recognition and remote sensing.