TextRSR: Enhanced Arbitrary-Shaped Scene Text Representation via Robust Subspace Recovery

Zhiwen Shao, Shengtian Jiang, Hancheng Zhu, Xuehuai Shi, Canlin Li, Lizhuang Ma, and Dit-Yan Yeung

***Abstract*—In recent years, scene text detection research has increasingly focused on arbitrary-shaped texts, where text representation is a fundamental problem. However, most existing methods still struggle to separate adjacent or overlapping texts due to ambiguous spatial positions of points or segmentation masks. Besides, the time efficiency of the entire pipeline is often neglected, resulting in sub-optimal inference speed. To tackle these problems, we first propose a novel text representation method based on robust subspace recovery, which robustly represents complex text shapes by combining orthogonal basis vectors learned from labeled text contours. These basis vectors capture basis contour patterns with distinct information, enabling clearer boundaries even in densely populated text scenarios. Moreover, we propose a dynamic sparse assignment scheme for positive samples that adaptively adjusts their weights during training, which not only accelerates inference speed by eliminating redundant predictions but also enhances feature learning by providing sufficient supervision signals. Building on these innovations, we present TextRSR, an accurate and efficient scene text detection network. Extensive experiments on challenging benchmarks demonstrate the superior accuracy and efficiency of TextRSR compared to state-of-the-art methods. Particularly, TextRSR achieves an F-measure of 88.5% at 37.8 frames per second (FPS) for CTW1500 dataset and an F-measure of 89.1% at 23.1 FPS for Total-Text dataset.**

***Index Terms*—Scene text detection, arbitrary-shaped text representation, robust subspace recovery, dynamic sparse assignment**

Manuscript received December, 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62472424, in part by the China Postdoctoral Science Foundation under Grant 2023M732223, in part by the Hong Kong Scholars Program under Grant XJ2023037/HKSP23EG01, and in part by the Research Impact Fund of the Hong Kong Government under Grant R6003-21. It was also supported in part by the National Natural Science Foundation of China under Grants 62402231, 72192821, and 62472282, in part by the Opening Fund of the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University under Grant VRLAB2024C03, and in part by the Science and Technology Planning Project of Henan Province under Grant 242102211003. (Corresponding authors: Shengtian Jiang, Hancheng Zhu, and Xuehuai Shi.)

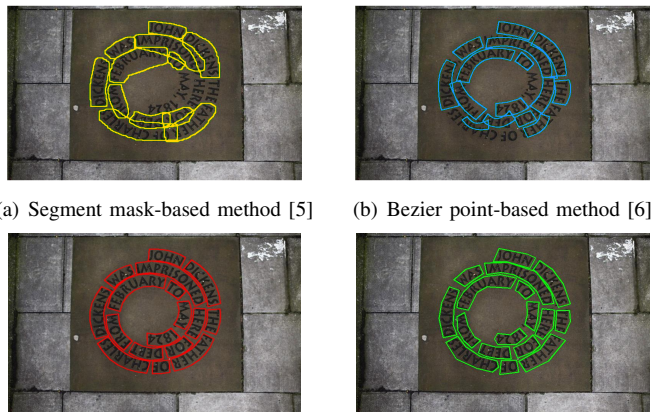
Z. Shao, S. Jiang, and H. Zhu are with the School of Computer Science and Technology, China University of Mining and Technology, Xuzhou 221116, China, and also with the Mine Digitization Engineering Research Center of the Ministry of Education, Xuzhou 221116, China. Z. Shao is also with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong, and also with the School of Computer Science, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: {zhiwen_shao; shengtian_jiang; zhuhancheng}@cumt.edu.cn).

X. Shi is with the School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210003, China (e-mail: xuehuai@njupt.edu.cn).

C. Li is with the School of Computer Science and Technology, Zhengzhou University of Light Industry, Zhengzhou 450002, China (e-mail: li-cl@zzuli.edu.cn).

L. Ma is with the School of Computer Science, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: ma-lz@cs.sjtu.edu.cn).

D.-Y. Yeung is with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong (e-mail: dyyeung@cse.ust.hk).



(a) Segment mask-based method [5] (b) Bezier point-based method [6]

(c) **TextRSR** (d) Ground truth

Fig. 1. Detection results of different methods in dense text scenarios. Both the segment mask-based method [5] and the Bezier point-based method [6] fail when adjacent texts are close, as shown in (a) and (b). In contrast, our method TextRSR accurately differentiates adjacent texts in such complex scenarios, as illustrated in (c), with the ground truth provided in (d).

I. Introduction

SCENE text detection is a widely researched topic in the field of computer vision, with numerous downstream applications, including image/video understanding [1], [2], visual search [3], and autonomous driving [4]. However, it remains a challenging task due to the complex geometric layout of texts, perspective distortions from shooting angles, and varying degrees of occlusion. Therefore, designing an effective text representation is a problem worthy of research.

Arbitrary-shaped scene text representations can be categorized into two main paradigms. One is the segmentation-based text representation which represents text shapes by grouping segmentation results at the pixel level [5], [7]–[12] or component level [13]–[15] through heuristic post-processing. The other approach is regression-based text representation, which models text shapes using contour points [16]–[19] or parameterized methods [6], [20]–[23].

Although both types of text representations achieve strong performance, they exhibit certain limitations. First, segmentation-based text representation methods [5], [7]–[12] may struggle with text-like background noise due to their limited global perception. Second, regression-based text representation methods [6], [24] may fail to accurately model highly curved texts with perspective distortions due to limited control points. Finally, both methods [5], [6] frequently have difficulty separating adjacent or overlapping texts due to ambiguous spatial positions of points or segmentation masks, as shown

in Fig. 1.

Moreover, regression-based text representation methods often overlook the overall efficiency of the pipeline, often resulting in sub-optimal inference performance. Based on how positive samples are assigned, existing positive sample assignment schemes adopted by regression-based text representation methods can be categorized into three types: dense assignment scheme [21], [22], [24], one-to-one assignment scheme [25]–[27], and dual assignment scheme [23]. The dense assignment schemes require the use of non-maximum suppression (NMS) to reduce numerous redundant predictions, and the process can be computationally intensive, particularly in dense text scenarios involving predictions of arbitrary shapes. The one-to-one assignment schemes adopt the set prediction mechanism from DETR [28] to avoid NMS, but due to the lack of sufficient supervision signals and positional priors, it typically requires stacking multiple decoders for iterative text contour refinement, resulting in a complex pipeline. The dual assignment scheme [23] combines both dense assignment and sparse assignment branches, in which the dense assignment branch provides sufficient supervision signals for training, while the sparse assignment branch accelerates inference speed. However, introducing the sparse branch also increases training complexity.

To tackle these problems, firstly inspired by the recent instance segmentation work [29], we propose a robust subspace recovery (RSR) method to robustly represent complex text shapes. Compared with most previous text representations, it offers distinct advantages. 1) Unlike Bezier point-based methods [6], [24], which require generating intermediate representations followed by discrete point regression in image space—a process prone to noise from occlusion that complicates distinguishing adjacent texts, our approach directly predicts the coefficients of RSR in parameter space, reducing noise susceptibility. 2) Besides, most parameterized text representation methods [5], [20]–[22] model each text instance independently. In contrast, our method models all text instances across the entire training set collectively, considering the shape relationships between different instances. 3) Our method captures fundamental contour patterns with well-differentiated information among patterns, allowing for clearer boundaries even in densely populated text scenarios where adjacent texts are close.

Our method RSR begins with constructing a text contour matrix containing all text contours in the training set. Based on robust subspace recovery [30], we utilize the projected Riemannian subgradient method (PRSGM) [31] to compute a robust M-dimensional subspace on the text contour matrix to find a set of orthogonal basis vectors. We then perform a sharing basis vectors conversion mechanism to reconstruct text contours by linearly combining these orthogonal basis vectors, as illustrated in Fig. 2.

Moreover, we propose a dynamic sparse assignment scheme (DSAS) for positive samples. During training, the weights of positive samples are adaptively adjusted to maintain a balance between feature learning and duplicate prediction removal. Specifically, we start with large weights for positive samples in the early training stage to provide ample supervised signals, enabling the network to learn feature more effectively. As training progresses, we gradually decrease the weights of positive samples to guide the network to reduce duplicate predictions, thereby decreasing the time of NMS during inference.

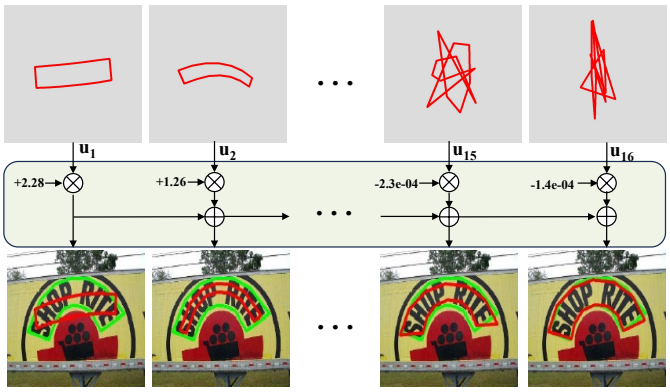


Fig. 2. Ilustration of the robust subspace recover text representation. $\mathbf{u}_1$, $\mathbf{u}_2$, $\cdots$, $\mathbf{u}_{15}$, and $\mathbf{u}_{16}$ are the orthogonal basis vectors that capture basic contour patterns with well-differentiated information. The text contour is approximated as a linear combination of these orthogonal basis vectors, with fitted curves shown progressively from left to right to demonstrate the effect of using an increasing number of the basis vectors. The ground truth contour is depicted in green.

Building on the above innovations, we present TextRSR, an accurate and efficient scene text detector. The main contributions of our work can be summarized as follows:

- We propose a text shape representation method named RSR, which utilizes robust subspace recovery approach to find a set of orthogonal basis vectors learned from labeled text contours, representing the text shape by linearly combining these orthogonal basis vectors.
- We introduce a dynamic sparse assignment scheme for positive samples that adaptively adjusts their weights during training, which simultaneously facilitates feature learning by providing sufficient supervision signals and accelerates inference speed by removing duplicate predictions.
- Extensive experiments are conducted on challenging benchmarks, demonstrating the superior accuracy and efficiency of our approach TextRSR compared to state-of-the-art methods. Particularly, TextRSR achieves 88.5% F-measure at 37.8 frames per second (FPS) and 89.1% F-measure at 23.1 FPS for CTW1500 and Total-Text datasets, respectively.

II. Related Work

Current text representation methods can be roughly divided into segmentation-based text representation methods and regression-based text representation methods. Besides, we introduce some positive sample assignment schemes adopted by regression-based text representation methods.

A. Segmentation-based Text Representation

Segmentation-based text representation methods represent text shapes by grouping segmentation results at the pixel or component level through holistic post-processing.

For pixel level methods, which can naturally represent arbitrary shape text, Pixellink [7] first estimates the connection relationships between pixels, subsequently extracting text bounding boxes by distinguishing links associated with different text entities. Tian et al. [8] conceptualizes each text instance as a cluster and used a two-step clustering strategy to segment dense text instances, ensuring that pixels within the same text unit typically belonged to the same cluster. TextField [9] learns a direction field that incorporates both the text mask and positional information relative to text boundaries, ultimately linking adjacent pixels to form candidate text regions. PSENet [10] predicts text instance kernels of varying scales, and then utilized a progressive expansion strategy to gradually enlarge these predefined kernels. DB [11] and DB++ [12] presents a differentiable binarization module that assigns elevated thresholds to text boundaries.

For component level methods, which represent the text with a set of text components, TextSnake [13] models text instances as a sequence of overlapping circular regions, predicting the text areas, centerlines, and various geometric attributes of these regions to reconstruct the text. Seglink++ [14] proposes instance-aware component grouping (ICG) method, which detects text by utilizing attractive and repulsive links between components to improve the separation of closely positioned text instances, and employs a modified minimum spanning tree algorithm for the final detection. Moreover, DRRG [32] and ReLaText [33] further deduce the relationships between local components using graph convolution networks.

However, these segmentation-based text representation methods, constrained by a local perspective, lack a global perception of the text, making it challenging to distinguish adjacent or overlapping text instances. Furthermore, they are often computationally intensive in post-processing, leading to sub-optimal inference speed.

B. Regression-based Text Representation

Compared with segmentation-based text representation methods, regression-based text representation methods regress the geometric information of text shape and position, thus avoiding intricate post-processing.

For horizontal and multi-oriented text, a simple rectangular or quadrilateral representation is generally sufficient. For arbitrary-shaped text, some methods [16]–[19] directly regress contour points in image space as text boundary and gradually increase the number of points as required to detect complex scenes.

Other approaches use parameterized curves or surfaces to represent the text contour. For instance, TextRay [20] employs Chebyshev polynomials under a polar coordinate system to approximate the text contour. FCENet [21] introduces the Fourier contour embedding (FCE) method, which approximates arbitrary-shaped text contours using compact Fourier signature vectors. ABCNet [24] uses two Bezier curves to represent the long edges of the text, thereby fitting the text contour, while TPSNet [22] leverages thin plate splines (TPS) to parameterize text contours using TPS fiducial points.

However, specific limitations are inherent in these methods. TextRay may struggle to represent text contours compactly and effectively due to inherent limitations in global geometric modeling, particularly when the text shapes are non-starconvex. FCENet may fail to capture partial corner pixels for extremely long or curved text. Additionally, ABCNet and TPSNet depend on intermediate representations, such as fiducial points, while discrete point regression in image space is susceptible to disruptions like occlusion or noise, which makes distinguishing adjacent text challenging. Our RSR approach addresses these challenges by predicting RSR coefficients within a parameter space, which effectively captures fundamental contour patterns with well-differentiated information. This capability enables the clear separation of closely positioned text, resulting in more distinct boundaries in text-dense scenarios.

Moreover, previous text representation methods tend to overlook structural relationships among various text shapes, limiting their ability to effectively model arbitrary-shaped texts. To address this issue, LRANet [23] employs singular value decomposition (SVD) to extract structural relationships among text contours and reconstruct text shapes using a few eigenvectors derived from labeled text contours. However, the traditional ℓ_2-based SVD solution is highly susceptible to outliers, which encompasses the ground-truth text boundary. Unlike LRANet, our method utilizes RSR technique to extract a set of orthogonal basis vectors from the text contour matrix and reconstruct text shapes through a linear combination of these orthogonal basis vectors.

C. Positive Sample Assignment Schemes

According to the allocation of positive samples, existing positive sample assignment schemes adopted by regression-based text representation methods can be divided into three types: dense assignment scheme [21], [22], [24], one-to-one assignment scheme [25], [27], and dual assignment scheme [23].

Most anchor-free CNN-based methods adopt the dense assignment schemes. For instance, [13], [21], [24] employ the "center sampling" strategy, where the text center region (TCR) is treated as the positive sample region. However, this approach has limitations: the model struggles to effectively capture interactions between the far-separated ends of long text sequences. To address this issue, TPSNet [22] proposes using Gaussian text center (GTC), where only predictions near the center point are retained for assigning positive samples. Despite these designs improvements, these models require non-maximum suppression (NMS) to eliminate redundant predictions. This process can be time-consuming, especially in dense text scenarios with arbitrary-shaped predictions. To address this problem, DETR-based [25], [27] methods adopt a one-to-one assignment scheme, which eliminates the need for NMS. However, these methods often require stacking multiple decoders for iterative text contour refinement, leading to a more complex pipeline.

On the other hand, LRANet [23] proposes a dual assignment scheme that attempts to address these challenges by combining a dense assignment branch to provide sufficient supervised signals during training with a sparse assignment branch to

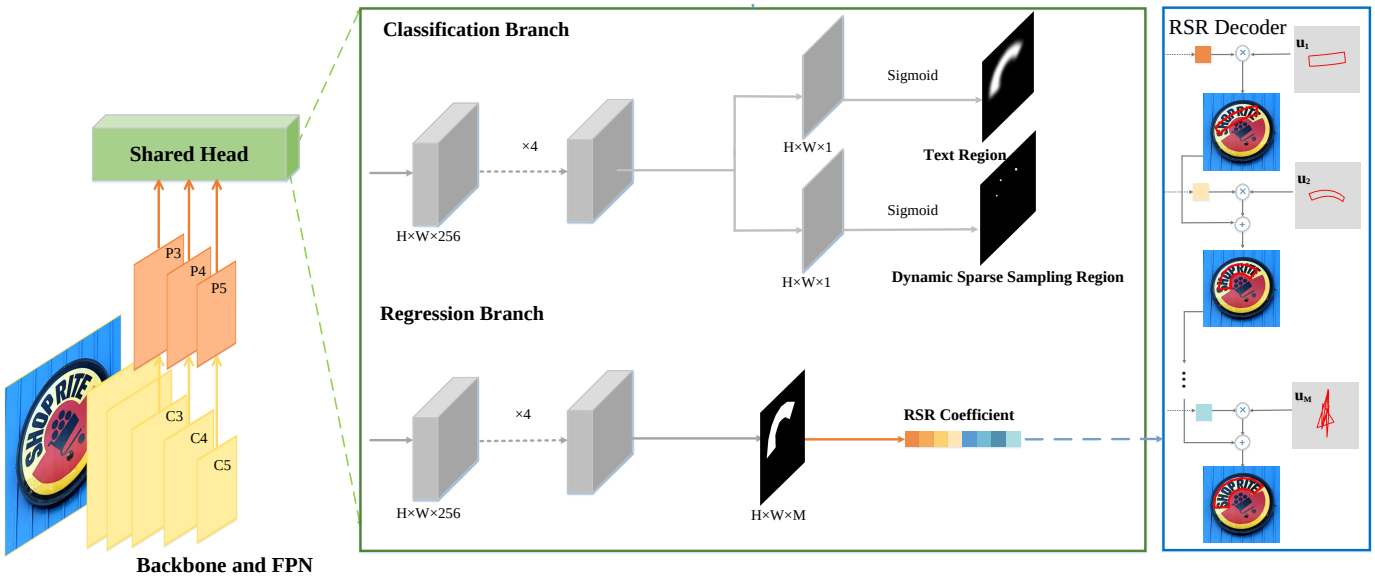


Fig. 3. The pipeline of the proposed TextRSR. Given an input image, multi-scale FPN features are extracted and fed into the shared head. In the shared head, the classification branch generates heatmaps for the text region (TR) and the dynamic sparse sampling region (DSSR), both of which are utilized to identify positive samples. Meanwhile, the regression branch predicts the RSR coefficients, which are used to linearly combine the basis vectors. In the RSR decoder, pixel-wise multiplication between the TR and DSSR predictions is performed to derive the final positive samples. The corresponding RSR coefficients for these samples are then decoded to reconstruct the text shape based on the predefined orthogonal basis vectors.

accelerate inference speed. However, the introduction of the sparse branch further increases training complexity.

To overcome the limitations of the above methods, we propose a dynamic sparse assignment scheme for positive samples. This approach adaptively adjusts the weights of positive samples during training, thereby enhancing feature learning while also accelerating inference speed by removing duplicate predictions without increasing additional parameters.

III. Arbitrary-Shaped Scene Text Detection via Robust Subspace Recovery

A. Overview

As illustrated in Fig. 3, following previous regression-based text detection networks [21]–[23], our TextRSR model is built on a compact one-stage fully convolutional architecture. It consists of three primary components: a feature extraction module, a detection head, and a RSR decoder for inference. The feature extraction module employs ResNet50 with a deformable convolutional network (DCN) as the backbone, and incorporates a feature pyramid network (FPN) to extract multi-scale feature maps. The detection head has two branches: one for classification and the other for regression. The classification branch uses four 3×3 convolutional layers to extract features, followed by two separate 3×3 convolutions for predicting the text region (TR) and the dynamic sparse sampling region (DSSR). The DSSR is the positive sample region within our dynamic sparse assignment scheme. In the regression branch, four 3×3 convolutional layers are used to extract features, followed by a single 3×3 convolution for predicting the RSR coefficients. In the RSR decoder, pixel-wise multiplication is performed between the predictions from TR and DSSR to obtain the final positive samples. The RSR coefficients corresponding to these samples are then decoded to reconstruct the text shape using orthogonal basis vectors, as defined by Eq. (4).

B. Robust Subspace Recovery Text Representation

Most existing text shape representation methods struggle to model arbitrary-shaped texts with compact layouts, particularly in dense text scenarios where adjacent texts are close due to ambiguous spatial position of points or segment masks. Furthermore, the interdependence of text contours motivates us to apply robust subspace projection to effectively capture contour patterns among the text contours. By capturing fundamental contour patterns with well-differentiated information among patterns, our method can achieve clearer boundaries even in densely populated text scenarios where adjacent texts are close.

First, we apply cubic spline interpolation to the ground truth text boundary, which consists of a variable number of vertices, to resample them into a fixed number of N vertices. Second, these resampled vertices are flattened into a single column vector, represented as $\mathbf{p} = [\mathbf{x}_1, \mathbf{y}_1, \cdots, \mathbf{x}_N, \mathbf{y}_N]^\top \in \mathbb{R}^{2N\times1}$. Then, a text contour matrix is constructed from these vectors, defined as $\mathbf{A} = [\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_L] \in \mathbb{R}^{2N\times L}$, where L denotes the number of labeled text instances in the training set.

Third, we need to compute a M-dimensional subspace $\mathbf{S}$ on the text contour matrix $\mathbf{A}$. We notice that there is a recently proposed method LRANet [23] using singular value decomposition (SVD) to obtain a low-rank representation. However, the traditional ℓ_2-based SVD solution is highly susceptible to outliers, including ground-truth contours of challenging texts. Therefore, we aim to robustly estimate the underlying low-dimensional subspace in the presence of outliers, known as robust subspace recovery (RSR) [30]. To achieve this, we employ the projected Riemannian subgradient method [31] to

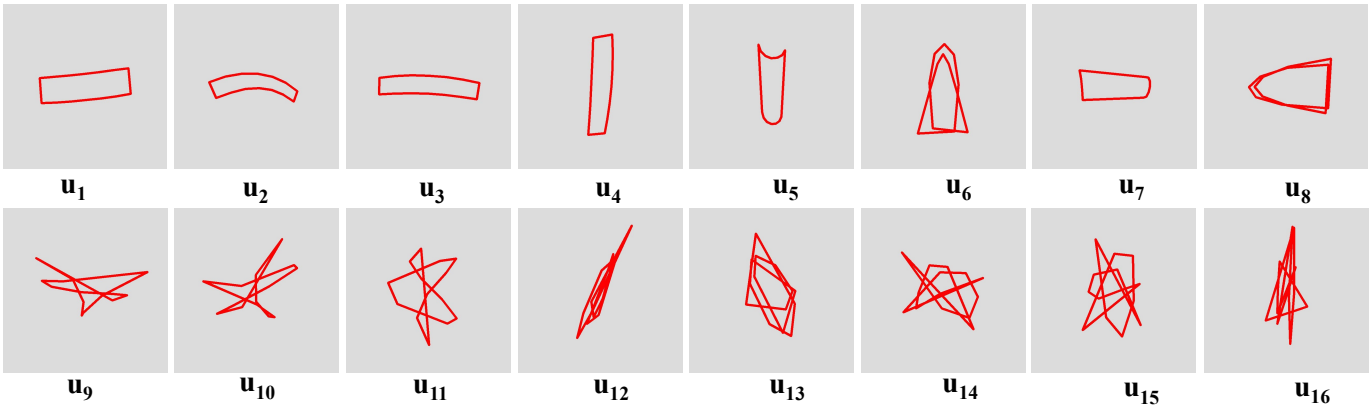


Fig. 4. Visualization of the sixteen basis vectors for the ICDAR2019-ArT [34] dataset revealing distinct patterns. The first five basis vectors capture typical contour patterns, while the remaining vectors focus on finer details of text shapes, leading to increasingly complex structures.

find a orthogonal column matrix $\mathbf{U}$ for the robust subspace $\mathbf{S}$ by solving the following optimization problem:

$$\min_{\mathbf{U}\in\mathbb{O}(2N,M)} \left\|(\mathbf{I}-\mathbf{U}\mathbf{U}^\top)\mathbf{A}\right\|_{1,2}, \tag{1}$$

where $\mathbb{O}(2N,M) := \left\{\mathbf{U}\in\mathbb{R}^{2N\times M} | \mathbf{U}^\top\mathbf{U}=\mathbf{I}_M\right\}$ denotes the set of $2N\times M$ orthogonal column matrices, and $\|\cdot\|_{1,2}$ is the mixed $\ell_{1,2}$-norm defined for any matrix $\mathbf{A}$ as the sum of the ℓ_2-norms of its rows.

Once the optimal $\mathbf{U}$ is computed by solving Eq. (1), the text contour matrix $\mathbf{A}$ can be approximated by projecting it onto the robust subspace $\mathbf{S}$:

$$\tilde{\mathbf{A}} = \mathbf{U}\mathbf{U}^\top\mathbf{A} = [\tilde{\mathbf{p}}_1,\ldots,\tilde{\mathbf{p}}_L] \approx \mathbf{A}, \tag{2}$$

where $\tilde{\mathbf{A}}$ is the projection of $\mathbf{A}$ onto the M-dimensional subspace $\mathbf{S}$ and $\tilde{\mathbf{p}}_i$ denotes the approximation of $\mathbf{p}_i$.

Next, we define the coefficient matrix $\mathbf{C} = \mathbf{U}^\top\mathbf{A} = [\mathbf{c}_1,\mathbf{c}_2,\ldots,\mathbf{c}_L] \in \mathbb{R}^{M\times L}$, allowing the matrix $\tilde{\mathbf{A}}$ to be expressed as:

$$\tilde{\mathbf{A}} = \mathbf{U}\mathbf{C} = [\mathbf{U}\mathbf{c}_1,\ldots,\mathbf{U}\mathbf{c}_L] = [\tilde{\mathbf{p}}_1,\ldots,\tilde{\mathbf{p}}_L]. \tag{3}$$

In Eq. (3), each approximated text contour $\tilde{\mathbf{p}}_i$ can be represented as a linear combination of the orthonormal basis vectors, as shown in the following equation:

$$\tilde{\mathbf{p}}_i = \mathbf{U}\mathbf{c}_i = [\mathbf{u}_1,\ldots,\mathbf{u}_M]\,\mathbf{c}_i. \tag{4}$$

These orthonormal basis vectors $\mathbf{u}_1,\mathbf{u}_2,\ldots,\mathbf{u}_M$ can describe basic contour patterns, as illustrated in Fig. 4.

Given any $2N$-dimensional text contour $\mathbf{p}$, we can project it onto the robust subspace $\mathbf{S}$ to obtain its approximation:

$$\tilde{\mathbf{p}} = \mathbf{U}\mathbf{c}, \tag{5}$$

where the coefficient vector $\mathbf{c}$ is given by:

$$\mathbf{c} = \mathbf{U}^\top\mathbf{p}. \tag{6}$$

Thus, in the robust subspace $\mathbf{S}$, a text contour $\mathbf{p}$ is approximately represented by the M-dimensional vector $\mathbf{c}$ obtained from Eq. (6). The approximation $\tilde{\mathbf{p}}$ of $\mathbf{p}$ can then be reconstructed via Eq. (4).

C. Dynamic Sparse Assignment Scheme

We propose a dynamic sparse assignment scheme for positive samples that adaptively adjusts their weights during training. This approach simultaneously enhances feature learning and accelerates inference speed by eliminating redundant predictions.

Specifically, we construct a prediction-aware matrix $\mathbf{S}$ that considers both classification and regression costs. Each element of this matrix is defined as:

$$s_{i,j} = \mathbb{1}^\infty\left[i\in\Omega_j\right]\times\left(\mathrm{FL}'\left(c_i\right)\right)^{1-\alpha}\times\sum_{n=0}^{N-1}\left\|\tilde{\mathbf{p}}_i^{(n)}-\mathbf{p}_j^{(n)}\right\|^\alpha, \tag{7}$$

where $s_{i,j}$ represents the matching cost between the predicted text contour of the i-th point and the ground-truth text contour of the j-th instance. The indicator function $\mathbb{1}^\infty\left[i\in\Omega_j\right]$ outputs 1 if point i lies within the text region Ω_j of the ground truth; otherwise, it returns ∞, ensuring that $S_{i,j}=\infty$. Here, c_i denotes the predicted classification score for the i-th point.

The classification cost, denoted as FL', is derived from the focal loss [35]. It is given by:

$$\mathrm{FL}'(x) = -\beta(1-x)^\gamma\log(x) + (1-\beta)x^\gamma\log(1-x), \tag{8}$$

where β serves as a weight factor to address the imbalance between positive and negative samples, while γ is a focusing parameter that down-weights the loss assigned to well-classified (easy) examples and thus focuses the learning on hard examples.

Finally, the term $\sum_{n=0}^{N-1}\left\|\tilde{\mathbf{p}}_i^{(n)}-\mathbf{p}_j^{(n)}\right\|^\alpha$ computes the regression cost, which is calculated as the L1 distance between the predicted and ground-truth text contours. The parameter α balances the relative importance between classification and regression costs.

Previous works [25], [27] typically formulate the selection of positive samples as a bipartite matching problem, often resolved using the Hungarian algorithm [36]. For simplicity, our method directly selects the K positive samples with the lowest matching costs based on $s_{i,j}$.

To minimize the likelihood of duplicate predictions, we assign dynamic soft labels to these selected positive samples.

Assuming the network undergoes X training epochs, the classification loss for each positive sample i in the j-th epoch is defined as:

$$l_i^j = -w_i^j \log(c_i) - (1 - w_i^j) \log(1 - c_i), \tag{9}$$

where c_i denotes the predicted classification score for the i-th point. The weights w_i^j and $(1-w_i^j)$ correspond to the positive and negative contributions of this point during the j-th epoch, respectively. The weight w_i^j is dynamically defined as:

$$\begin{aligned} w_i^j &= T^j \times \frac{c_i}{\max(c_k)}, \\ T^j &= T^{\max} + \frac{T^{\min} - T^{\max}}{X - 1} \times j, \end{aligned} \tag{10}$$

where T^j is a time-dependent variable assigned uniformly across all samples in the j-th epoch. The parameters $T^{\max}$ and $T^{\min}$ control the weights of the selected K positive sample points in the initial and final epochs, respectively. We ensure that the weights are positively correlated with the classification scores, meaning that points with higher prediction scores have a more significant impact on the positive signals.

Using c_i directly as the weight can lead to instability during training, especially for hard samples with much lower predicted scores compared to easy samples. To address this, we normalize the weights by taking the ratio of c_i to the maximum classification score, $\max\{c\}$, ensuring that all sample weights are scaled uniformly.

Adjusting T^j dynamically is crucial as it manages the balance between feature learning and duplication removal at different training stages. In the early phases of training, T^j is set to a higher value to provide ample positive supervision signals for robust feature representation learning, allowing the network to converge quickly. As training progresses, T^j is gradually decreased, reducing the positive weights of these points and enabling the network to effectively eliminate duplicate predictions.

D. Objective Function

In our TextRSR framework, the optimization objective of the network is formulated as:

$$\mathcal{L} = \mathcal{L}_{cls} + \mathcal{L}_{reg}, \tag{11}$$

where $\mathcal{L}_{cls}$ and $\mathcal{L}_{reg}$ are the losses for the classification branch and the regression branch, respectively.

The classification loss consists of the text region loss $\mathcal{L}_{TR}$ and the dynamic sparse sampling region loss $\mathcal{L}_{DSSR}$:

$$\mathcal{L}_{cls} = \mathcal{L}_{TR} + \mathcal{L}_{DSSR}, \tag{12}$$

where $\mathcal{L}_{TR}$ and $\mathcal{L}_{DSSR}$ are the cross-entropy loss and focal loss, respectively. To solve the sample imbalance problem, OHEM [37] is adopted for $\mathcal{L}_{TR}$.

Our regression loss is defined as:

$$\mathcal{L}_{reg} = \sum_i \mathbb{1}[i \in \text{DSSR}] l_1\left(\tilde{\mathbf{p}}_i, \mathbf{p}_i\right), \tag{13}$$

where the indicator function $\mathbb{1}\left[i \in \text{DSSR}\right]$ outputs 1 if point i lies within the dynamic sparse sampling region (DSSR); otherwise, it returns 0. l_1 denotes the smooth-L1 loss.

IV. Experiments

A. Datasets and Settings

1) Datasets: In this work, we follow [22] and use SynthText-150K for pre-training our model. We evaluate the performance of our model on four datasets: CTW1500, Total-Text, ICDAR2019-ArT, and ICDAR2015, reporting results both with and without pre-training.

- **SynthText-150K** [24] is a large-scale synthetic text image dataset, containing nearly 150,000 images that consist of both straight and curved texts.
- **CTW1500** [38] is a natural scene text detection data set that pays special attention to curved text. It contains 1000 training images and 500 test images.
- **Total-Text** [39] is a comprehensive dataset, especially for arbitrarily shaped text. It contains 1255 training images and 300 testing images. Instances of text in images come in many different orientations, such as horizontal, multi-directional, and curved.Text areas are annotated by a non-fixed number of polygons.
- **ICDAR2019-ArT** [34] is a complex large-scale multi-lingual arbitrary shape text detection dataset. It includes 5,603 images for training and 4,563 for testing. The text regions in this dataset are annotated using polygons with an adaptive number of key points, providing a flexible representation of the text boundaries.
- **ICDAR2015** [40] is an incidental scene text dataset which contains 1000 training images and 500 test images. Text instances in the images appear in random scale, orientation, location, viewpoint, and blurring. The annotations are in the form of quadrilateral bounding-boxes represented by 8 coordinates of four clockwise corners.

2) Implementation Details: We implement our TextRSR model based on MMOCR [41] with PyTorch library [42]. The backbone network is ResNet50, pre-trained on ImageNet, with DCN applied in stages 2, 3, and 4, followed by FPN. The text scale ranges for P3, P4, and P5 in the FPN are set to $[0, 0.25]$, $[0.2, 0.65]$, and $[0.55, 1]$ of the image size, respectively. The dimension M of the robust subspace is set to 16, while the sparse sampling number K in the dynamic sparse assignment scheme is 3. Besides, $T^{\max}$ and $T^{\min}$ are set to 0.8 and 0.2, respectively. In Eq. (8), β and γ are set to 0.25 and 2.0, respectively. When training from scratch, stochastic gradient descent (SGD) is used as the optimizer, with an initial learning rate of 0.001, weight decay of 0.0005, and momentum of 0.9. Each dataset is trained independently using its respective training set. The batch size is 8, and the models are trained for 500 epochs. To ensure comprehensive comparisons, we pre-train the model on SynthText-150K for 10 epochs, followed by fine-tuning for 500 epochs on all datasets with an initial learning rate of 0.002. Data augmentation techniques include random rotation, scaling, flipping, and cropping.

During testing, the shorter sides of the test images are resized to 800, 1000, 1600, and 1200 for CTW1500, Total-Text, ICDAR2019-ArT, and ICDAR2015, respectively, while maintaining the original aspect ratio. All experiments are conducted on an NVIDIA RTX 3090 GPU. In the following

TABLE I
PERFORMANCE GAINS OF RSR AND DSAS ON CURVE TEXTS.

Dataset	Method	Recall	Precision	F-measure
CTW1500	Baseline	85.1	79.5	82.2
	RSR	**87.9**	81.2	84.4
	RSR+DSAS	87.3	**86.2**	**86.8**
Total-Text	Baseline	86.2	83.3	84.7
	RSR	88.3	**86.3**	87.3
	RSR+DSAS	**90.1**	**86.3**	**88.2**

TABLE II
PERFORMANCE COMPARISON OF THE SAME NETWORK USING DIFFERENT TEXT REPRESENTATIONS ON CTW1500.

Representation	Dim	R	P	F
Contour points	28	84.4	82.6	83.5
Bezier points	16	83.4	84.7	84.1
RSR	14	86.7	86.1	86.4
RSR	**16**	**87.3**	**86.2**	**86.8**

sections, we omit % for simplicity in the recall (R), precision (P), and F-measure (F) results.

B. Ablation Study

In this section, to validate the proposed method RSR and DSAS in our TextRSR, we conduct ablation studies on both CTW1500 and Total-Text datasets without pre-training on SynthText-150K.

1) Effectiveness of RSR and DSAS: Table I shows the results of using RSR and DSAS over the baseline model on CTW1500 and Total-Text datasets. The baseline model is derived from the TPSNet model [22] by removing the TPS text representation method. In place of this, the model represents the text shape by directly regressing 14 contour points and employs the popular text center region (TCR) [13], [21] as the positive sample assignment scheme. The results indicate that replacing contour points with RSR to represent text shapes improves F-measure scores by 2.2 and 2.6 on CTW1500 and Total-Text, respectively. When both RSR and DSAS are applied, F-measure scores further increase by 4.6 and 3.7 on CTW1500 and Total-Text, respectively.

2) Different Text Representation Methods: To further validate our proposed text representation method, RSR, we investigate the effects of different text representations within the TextRSR network structure. As shown in Table II, our RSR outperforms both contour points and Bezier points representations by 3.3 and 2.7 in F-measure, respectively. This improvement is primarily due to our approach RSR, which captures fundamental contour patterns with distinct information and predicts the coefficients of RSR in parameter space. In contrast, the other two methods regress discrete points in the image space, making them more susceptible to noise, such as occlusion, which makes it challenging to clearly distinguish adjacent text. Moreover, our RSR still surpasses the other two methods by at least 2.3 in F-measure, with a lower representation dimension 14.

3) Different Positive Sample Assignment Schemes: To further examine our proposed positive sample assignment scheme, DSAS, we conduct comparisons with dense assignment scheme, one-to-one assignment scheme, and dual assignment scheme. To ensure fairness in comparison, we use the TCR dense positive sample assignment scheme from TPSNet [22] and the one-to-one assignment scheme from DETR [28] as baselines. Besides, we compare with the recent dual assignment technique [23]. As shown in Table III, our dynamic sparse assignment scheme achieves a faster inference speed (23.1 vs. 15.1), attributed to a significant reduction in duplicate predictions, while also delivering a 3.9 improvement in F-measure over the dense assignment. Additionally, when compared to the one-to-one assignment, our method shows a 1.0 increase in F-measure, while preserving high inference speed. Moreover, compared to the dual assignment scheme, our method achieves improvements of 0.2 in F-measure and 0.7 in FPS. These results indicate that our approach strikes a good balance between effective representation learning and high inference speed.

4) Dimension of Robust Subspace: To verify the generalization ability of the robust subspace dimension, i.e., M,

TABLE III
PERFORMANCE OF TEXTRSR WITH DIFFERENT POSITIVE SAMPLE ASSIGNMENT SCHEMES ON TOTAL-TEXT.

Assignment Scheme	R	P	F	FPS
Dense [22]	84.9	83.7	84.3	15.1
One to one [28]	86.1	88.4	87.2	**23.5**
Dual [23]	86.6	**89.3**	88.0	22.4
DSAS	**90.1**	86.3	**88.2**	23.1

TABLE IV
PERFORMANCE OF TEXTRSR WITH DIFFERENT SUBSPACE DIMENSIONS ON CTW1500 AND TOTAL-TEXT.

Dim	CTW1500			Total-Text		
	R	P	F	R	P	F
14	86.7	86.1	86.4	85.6	89.5	87.5
16	**87.3**	86.2	**86.8**	**90.1**	86.3	**88.2**
18	85.5	**87.3**	86.4	86.2	**89.8**	88.0

TABLE V
PERFORMANCE OF TEXTRSR WITH DIFFERENT NUMBER OF DYNAMIC SPARSE SAMPLES ON TOTAL-TEXT.

K	R	P	F	FPS
5	86.6	89.8	**88.2**	21.4
3	**90.1**	86.3	**88.2**	23.1
1	84.1	**90.4**	87.1	**23.8**

TABLE VI
F-MEASURE OF TEXTRSR WITH DIFFERENT T^{max} AND T^{min} VALUES ON TOTALTEXT.

T^{min} \ T^{max}	0.9	**0.8**	0.7
0.3	87.5	87.9	87.3
0.2	87.5	**88.2**	**87.7**
0.1	**87.9**	87.6	87.5

TABLE VII
COMPARISONS WITH STATE-OF-THE-ART METHODS ON CTW1500 AND TOTAL-TEXT. EXT DENOTES USING EXTRA DATA TO PRE-TRAIN, AND SYN, MLT, ART, AND MIXT REFER TO THE FOLLOWING DATASETS: SYNTHTEXT [43], ICDAR2017-MLT [44], ICDAR2019-ART [34], AND A MIXED DATASET COMPRISING SYNTHCURVE [24], COCO-TEXT [45], AND ICDAR2019-MLT [46], RESPECTIVELY. §DENOTES SEGMENTATION-BASED TEXT REPRESENTATION METHODS, AND †DENOTES REGRESSION-BASED TEXT REPRESENTATION METHODS.

Method	Paper	Ext	CTW1500				Total-Text			
			Recall	Precision	F-measure	FPS	Recall	Precision	F-measure	FPS
TextSnake§ [13]	ECCV'18	Syn	85.3	67.9	75.6	-	74.5	82.7	78.4	-
SegLink++§ [14]	PR'19	Syn	79.8	82.8	81.3	-	80.9	82.1	81.5	-
TextField§ [9]	TIP'19	Syn	79.8	83.0	81.4	6.0	79.9	81.2	80.6	6.0
MSR§ [47]	IJCAI'19	Syn	78.3	85.0	81.5	4.3	74.8	83.8	79.0	4.3
PSENet-1s§ [10]	CVPR'19	MLT	79.7	84.8	82.2	3.9	78.0	84.0	80.9	3.9
ATRR§ [18]	CVPR'19	-	80.2	80.1	80.1	10.0	76.2	80.9	78.5	-
CRAFT§ [15]	CVPR'19	Syn	81.1	86.0	83.5	-	79.9	87.6	83.6	-
PAN§ [48]	ICCV'19	Syn	81.2	86.4	83.7	**39.8**	81.0	89.3	85.0	**39.6**
DRRG§ [32]	CVPR'20	MLT	83.0	85.9	84.5	-	84.9	86.5	85.7	-
DB§ [11]	AAAI'20	Syn	80.2	86.9	83.4	22.0	82.5	87.1	84.7	32.0
ReLaText§ [33]	PR'21	Syn	83.3	86.2	84.8	10.6	83.1	84.8	84.0	3.2
DB++§ [12]	TPAMI'22	Syn	82.8	87.9	85.3	26.0	83.2	88.9	86.0	28.0
TextDCT§ [5]	TMM'22	Syn	85.3	85.0	85.1	17.2	82.7	87.2	84.9	15.1
Wang *et al.*§ [49]	TIP'23	-	82.5	85.3	83.9	25.1	79.9	88.7	84.1	24.3
LOMO† [50]	CVPR'19	Syn	69.6	89.2	78.4	4.4	75.7	86.6	81.6	4.4
TextRay† [20]	MM'20	ArT	80.4	82.8	81.6	-	77.9	83.5	80.6	-
ContourNet† [51]	CVPR'20	-	84.1	83.7	83.9	4.5	83.9	86.9	85.4	3.8
OPMP† [52]	TMM'21	-	80.8	85.1	82.9	1.4	82.7	87.6	85.1	1.4
Dai *et al.*† [53]	TMM'21	-	80.4	86.2	83.2	0.6	81.2	85.4	83.2	0.7
PCR† [16]	CVPR'21	MLT	82.3	87.2	84.7	-	82.0	88.5	85.2	-
FCENet† [21]	CVPR'21	-	83.4	87.6	85.5	-	82.5	89.3	85.8	-
ABCNet V2† [6]	TPAMI'21	MixT	83.8	85.6	84.7	-	84.1	89.2	87.0	-
TextBPN [19]	ICCV'21	Syn	81.4	87.8	84.5	12.1	84.6	90.2	87.3	-
TPSNet† [22]	MM'22	Syn	85.1	87.7	86.4	17.9	86.8	89.5	88.1	14.3
CT-Net† [27]	TCSVT'23	Syn	83.8	88.5	86.1	11.2	85.0	90.8	87.8	10.1
LRANet† [23]	AAAI'24	Syn	85.5	89.4	87.4	37.2	87.8	90.3	89.0	22.1
TextRSR†	Ours	-	87.3	86.2	86.8	37.8	**90.1**	86.3	88.2	23.1
TextRSR†	Ours	Syn	**87.5**	**89.5**	**88.5**	37.8	86.5	**91.7**	**89.1**	23.1

we conduct experiments on both CTW1500 and Total-Text datasets. The results are listed in Table IV. We can see that as M increases to 16, the F-measure improves by 0.4 and 0.7 in CTW1500 and Total-Text, respectively. However, when M is further increased to 18, the F-measure decreases by 0.4 and 0.2 in CTW1500 and Total-Text, respectively. These observations suggest that the robust subspace dimension M generalizes well across different datasets, with $M = 16$ yielding optimal performance.

5) Different Number of Dynamic Sparse Sampling: Here we further investigate the impact of the dynamic sparse sampling number K on model performance, as presented in Table V. When $K = 3$, the model achieves the most balanced performance, with an F-measure of 88.2 and an FPS of 23.1, indicating an effective trade-off between accuracy and inference speed. Increasing K to 5 does not yield any further improvement in the F-measure, while the FPS decreases to 21.4. This suggests that setting $K = 3$ is sufficient to provide adequate supervised signals for feature learning. Conversely, setting $K = 1$ increases precision to 90.4 and maximizes FPS at 23.8, but lowers the F-measure to 87.1 due to decreased recall.

6) T^{max} and T^{min}: These two parameters control the weights of positive sample points in the first and last epoch, respectively, during the training process. As shown in Table VI, the highest F-measure of 88.2 is achieved when $T^{\text{max}} = 0.8$ and $T^{\text{min}} = 0.2$, suggesting this combination offers an optimal balance for accurate model performance. When T^{min} is set to 0.1, the F-measure peaks at 87.9 when $T^{\text{max}} = 0.9$, but it decreases slightly to 87.6 and 87.5 for $T^{\text{max}} = 0.8$ and 0.7, respectively. Similarly, at $T^{\text{min}} = 0.3$, the F-measure ranges from 87.5 at $T^{\text{max}} = 0.9$ to 87.3 at $T^{\text{max}} = 0.7$, showing a gradual decline. Overall, the combination of $T^{\text{max}} = 0.8$ and $T^{\text{min}} = 0.2$ stands out as the most effective, providing the highest F-measure of 88.2, highlighting its potential as the optimal setting for the best model performance on the Total-Text dataset.

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

We compare our TextRSR with previous works on three benchmarks, including two benchmarks for curved texts and one benchmark for large-scale multi-lingual arbitrary shape texts. Some qualitative results are visualized in Fig. 5, which demonstrates the effectiveness of our TextRSR on long, curve, adjacent, and dense texts. For a fair comparison, we only record our model's single-scale testing results on all datasets. Moreover, for text spotting methods [6], [22], [54], [55], we only show the detection results without recognition module.

1) Evaluation on Long Curved Text Benchmark: As shown in Table VII, we first compare our results with state-of-the-art methods on the long curved text dataset CTW1500. TextRSR achieves recall, precision, and F-measure scores of 87.5, 89.5,

(a) CTW1500

(b) Total-Text

(c) ICDAR2019-ArT

(d) ICDAR2015

Fig. 5. Examples of text detection results obtained with the proposed TextRSR approach on benchmark datasets.

TABLE VIII
COMPARISON WITH OTHER PARAMETERIZED TEXT SHAPE METHODS ON CTW1500. IOU RESULTS FOR OTHER METHODS ARE SOURCED FROM [23].

Method	Dim	IoU
Chebyshev [20]	44	83.6
DCT [5]	32	88.5
Fourier [21]	22	91.5
Bezier [24]	16	97.6
TPS [22]	22	97.9
LRA [23]	14	98.0
RSR	16	**98.3**

TABLE IX
COMPARISON WITH STATE-OF-THE-ART METHODS ON ICDAR2019-ART. EXT DENOTES USING EXTRA DATA TO PRE-TRAIN.

Method	Paper	Ext	R	P	F
PSENet-1s [56]	CVPR'19	✓	52.2	75.9	61.9
CRAFT [15]	CVPR'19	✓	68.9	77.3	72.9
PAN [48]	ICCV'19	✓	**79.4**	61.1	69.1
TextRay [20]	MM'20	✓	58.6	76.0	66.2
ContourNet [51]	CVPR'20		62.1	73.2	67.2
PCR [16]	CVPR'21	✓	66.1	84.0	74.0
TPSNet [22]	MM'22		70.9	81.0	75.6
TPSNet [22]	MM'22	✓	73.3	84.3	**78.4**
EMA [57]	TIP'22	✓	68.7	80.8	74.3
Wang *et al.* [49]	TIP'23	✓	60.5	78.5	68.4
TextRSR	Ours		71.1	82.9	76.6
TextRSR	Ours	✓	71.6	**85.3**	77.8

and 88.5, respectively, outperforming all comparison methods in performance. Some qualitative results on CTW1500 dataset are depicted in Fig. 5(a). Notably, even without pre-training, TextRSR still surpasses all comparison methods in F-measure. Besides, thanks to our concise network architecture and the proposed positive sample assignment scheme (DSAS), TextRSR achieves a high inference speed of 37.8 FPS, which indicates that our model is capable of real-time text detection. Furthermore, as shown in Table VIII, compared to other parameterized text shape representation methods, our approach achieves the highest IoU of 98.3, while maintaining a competitive dimension for the representation.

2) Evaluation on Curved Text Benchmark: As shown in Table VII, we first compare our results with state-of-the-art methods on the curved text dataset Total-Text. TextRSR achieves recall, precision, and F-measure scores of 86.5, 91.7, and 89.1, respectively, outperforming all comparison methods in performance. Qualitative results on the Total-Text dataset are presented in Fig. 5(b). Compared to parameterized text representation methods [5], [20]–[23] , our TextRSR outperforms the best performing method, LRANet [23] by 0.1 in F-measure and 1.0 in FPS. Moreover, TextRSR significantly

(a) Ground truth (b) TextSnake (c) PAN (d) PSENet (f) DRRG (g) FCENet (h) ABCNet V2 (i) TPSNet (j) **TextRSR**

Fig. 6. Qualitative comparison with previous methods on selected adjacent text samples from the CTW1500 dataset. Ground truth is shown in column (a), marked in red. Segmentation-based text representation methods [10], [13], [48], [58] are presented in columns (b)–(f), highlighted in yellow, while regression-based text representation methods [6], [21], [22] are shown in columns (g)–(j), marked in green.

outperforms segmentation-based text representation methods [9]–[12], [19], [32], [48] by at least 3.1 in F-measure.

3) Evaluation on Large-Scale Multi-Lingual Arbitrary-Shaped Text Benchmark: To demonstrate the generalization ability of our proposed method, we evaluate our model on the ICDAR2019-ArT dataset, which contains numerous multilingual curved text instances from complex scenes. As shown in Table IX, even without pre-training, TextRSR achieves recall, precision, and F-measure scores of 71.1, 82.9, and 76.6, respectively. These results outperform most pre-trained methods in terms of F-measure, highlighting TextRSR's adaptive ability to large-scale datasets. However, with pre-training, there remains a gap in the F-measure compared to the previous best method TPSNet [22] (77.8 vs. 78.4). We attribute this discrepancy to the domain mismatch between the basis vectors learned from the real-world ICDAR2019-ArT dataset and those from the synthetic SynthText-150K dataset. Specifically, the contour patterns captured by these basis vectors exhibit differences across domains, which may account for the relatively modest performance gain from pre-training compared to TPSNet [22] (1.2 vs. 2.8). Some qualitative results on the ICDAR2019-ArT dataset are illustrated in Fig. 5(c).

4) Visual Comparison: As shown in the Fig. 6, we visualize both segmentation-based [10], [13], [48], [58] and regression-based text representation methods [21], [22] on the adjacent text samples from CTW1500 test dataset using MMOCR [41]. Note that most models for Total-Text or ICDAR2019-ArT have not been released by MMOCR, and the visual results for ABCNet V2 [6] are reproduced by AdelaiDet [59].

The first row depicts dense text scenarios where adjacent texts are extremely close, even overlapping. In contrast to all other text representation methods, which struggle with text adhesion (also shown in the last row), our TextRSR effectively distinguishes adjacent texts, even in such complex scenarios. This is primarily because our method RSR captures fundamental contour patterns with well-differentiated information across these patterns.

Moreover, when texts exhibit highly complex shapes in a compact layout, as shown in the middle row, our TextRSR still performs well. This is mainly due to our method's ability to model all text instance shapes across the entire training set collectively, accounting for the shape relationships between different instances.

D. TextRSR for Scene Texts in Quadrilateral Formats

To evaluate the applicability of our method to texts in quadrilateral annotations, we compare with previous works on ICDAR2015. As shown in Table X, our method, pre-trained on SynthText-150K, achieves an F-measure of 88.7, surpassing the state-of-the-art methods [22], [27] by 0.1. Without pre-training, our method still outperforms ContourNet [51], achieving an F-measure of 87.5 compared to 86.9. Besides, the visualization results shown in Fig. 5(d) demonstrate the effectiveness of our method in handling texts in quadrilateral formats.

E. TextRSR with Swin Transformer as Backbone

We conduct experiments by replacing the original ResNet50 backbone with a more advanced Swin Transformer-T model. As illustrated in Table XI, TextRSR using the Swin Transformer-T backbone achieves consistent performance improvements on both CTW1500 and Total-Text datasets. Specifically, with SynthText-150K pretraining, the F-measure on CTW1500 improves from 88.5 to 88.8, and on Total-Text improves from 89.1 to 89.5. These results indicate that TextRSR is compatible with Transformer-based architectures and

TABLE X
COMPARISONS WITH STATE-OF-THE-ART WORKS ON ICDAR2015.

Method	Paper	Ext	R	P	F	FPS
ATRR§ [18]	CVPR'19	-	86.0	89.2	87.6	-
PSENet-1s§ [14]	CVPR'19	MLT	84.5	86.9	85.7	1.6
CRAFT§ [15]	CVPR'19	Syn	84.3	89.8	86.9	8.6
PAN§ [48]	ICCV'19	Syn	81.9	84.0	82.9	**26.1**
DRRG§ [32]	CVPR'20	MLT	84.7	88.5	86.6	-
TextMountain§ [60]	PR'21	Syn	84.1	87.3	85.7	10.4
TextDCT§ [5]	TMM'22	Syn	84.8	88.9	86.8	7.5
DBNet++§ [12]	TPAMI'22	Syn	83.9	90.9	87.3	10.0
Wang *et al.* [49]	TIP'23	-	82.7	89.8	86.1	12.1
CBNet§ [61]	IJCV'24	MLT	85.4	91.0	88.1	-
SPCNet† [62]	AAAI'19	MLT	85.8	88.7	87.2	-
LOMO† [50]	CVPR'19	Syn	83.5	**91.3**	87.2	3.4
ContourNet† [51]	CVPR'20	-	86.1	87.6	86.9	3.5
Boundary† [63]	MM'20	Syn	82.2	88.1	85.0	-
R-Net† [64]	TMM'21	Syn	82.8	88.7	85.6	21.4
FCENet† [21]	CVPR'21	-	82.6	90.1	86.2	-
MOST† [65]	CVPR'21	Syn	87.0	89.1	88.2	10.0
TPSNet† [22]	MM'22	Syn	86.6	90.7	88.6	11.6
EMA† [57]	TIP'22	Syn	82.4	89.4	85.8	21.6
CT-Net† [27]	TCSVT'23	Syn	86.4	90.9	88.6	6.5
TextRSR†	Ours	-	86.8	88.2	87.5	16.6
TextRSR†	Ours	Syn	**87.3**	90.1	**88.7**	16.6

TABLE XI
PERFORMANCE OF OUR TEXTRSR WITH DIFFERENT BACKBONES ON CTW1500 AND TOTAL-TEXT.

Method	Backbone	Ext	CTW1500				Total-Text			
			R	P	F	FPS	R	P	F	FPS
TextRSR	Res50	-	87.3	86.2	86.8	**37.8**	**90.1**	86.3	88.2	**23.1**
TextRSR	Res50	Syn	87.5	**89.5**	88.5	**37.8**	86.5	**91.7**	89.1	**23.1**
TextRSR	Swin-T	-	87.9	86.5	87.2	32.6	89.1	87.9	88.5	19.4
TextRSR	Swin-T	Syn	**89.5**	88.1	**88.8**	32.6	88.1	90.9	**89.5**	19.4

benefits from their stronger feature extraction capabilities, although with a slight decrease in inference speed.

F. RSR vs. SVD

There is a recent parameterized text representation method LRANet [23]. It utilizes singular value decomposition (SVD) to extract eigenvectors and represents text contours through a linear combination of these eigenvectors. However, traditional ℓ_2-based SVD technique may suffer from a lack of robustness when confronted with outliers.

To fairly evaluate the robustness of our proposed RSR method against traditional SVD in the presence of training outliers (TOs), we replace RSR with SVD within our TextRSR framework, in which the new structure is named LRANet*. Since existing scene text datasets contain very few outliers, we design an algorithm to inject synthetic outliers into the CTW1500 training set. Specifically, for each image, we randomly select 0 to 3 text instances, and for each selected instance, randomly replace 0 to 3 contour points with nearest neighboring points from adjacent instances using KD-Tree algorithm [66]. The results of these two parameterized text representation methods are presented in Table XII.

It can be observed that TextRSR consistently outperforms LRANet* across both detection performance (F-measure) and contour reconstruction accuracy (IoU), under conditions with and without training outliers. At the same dimension of 16, TextRSR achieves an improvement of 0.3 in F-measure and 0.1 in IoU over LRANet* without training outliers. When training outliers are introduced, the performance gap further widens, in which TextRSR surpasses LRANet* by 0.6 in F-measure and 0.5 in IoU. Moreover, when the dimension is reduced to 14, the advantage becomes more evident, with TextRSR outperforming LRANet* by 1.1 in F-measure and 0.9 in IoU.

TABLE XII
PERFORMANCE COMPARISON BETWEEN LRANET* AND TEXTRSR ON CWT1500. TOS REFERS TO GENERATED TRAINING OUTLIERS.

Method	Dim	TOs	R	P	F	IoU
LRANet*	14	✓	84.2	85.8	85.0	96.1
	16	✓	85.4	86.4	85.9	97.3
	16	-	85.2	**87.8**	86.5	98.2
TextRSR	14	✓	87.0	85.2	86.1	97.0
	16	✓	**87.4**	85.6	86.5	97.8
	16	-	87.3	86.2	**86.8**	**98.3**

Notably, TextRSR maintains stable performance regardless of the presence of training outliers, with a slight difference in F-measure (0.3) and IoU (0.5) at the dimension of 16. In contrast, LRANet* exhibits larger performance degradation when training outliers are introduced, with a drop of 1.4 in F-measure and 0.9 in IoU at the same dimension. This performance gap can be attributed to RSR's robust subspace learning mechanism, which effectively suppresses the influence of outlier points during basis vector computation, whereas SVD-based decomposition tends to overfit noisy annotations due to its sensitivity to outliers.

Their visual results across various text scenarios are illustrated in Fig. 7, including adjacent text (column 1), occluded text (column 2), multi-oriented text (column 3), and curved text (columns 4 and 5). It is demonstrated that TextRSR is superior to LRANet* in the presence of training outliers.

G. Limitations

Based on the experimental results, our TextRSR demonstrates robust performance across various challenging scenarios, especially in densely populated scenes with closely spaced adjacent text. However, certain limitations persist, particularly in cases involving low-contrast texts and object-like texts, as illustrated in Fig. 8. These scenarios remain challenging, not only for our method but also for the state-of-the-art methods [6], [10], [13], [16], [21], [22], [48], [58].

V. CONCLUSION

In this paper, we have proposed TextRSR, an accurate and efficient detector for arbitrary-shaped scene text. A novel RSR text representation has been introduced to enhance arbitrary-shaped text representation by leveraging a robust subspace recovery approach. This method learns a set of orthogonal basis vectors from labeled text contours, capturing fundamental contour patterns while maintaining well-differentiated information among them. By representing text shapes through linear combinations of these orthogonal basis vectors, TextRSR enables clearer boundaries in densely populated text

Fig. 7. Qualitative comparisons on challenging samples from the CTW1500 dataset. The top and bottom rows show the detection results of LRANet* and our TextRSR, respectively.

(a) (b)

Fig. 8. Failure cases, in which red contours are ground truths while green contours are predicted results. (a) Low-contrast texts. (b) Object-like texts.

scenarios where adjacent texts are close. Furthermore, we have presented a dynamic sparse assignment scheme for positive samples, which adaptively adjusts their weights during training. This scheme not only enhances feature learning by providing sufficient supervision signals but also accelerates inference speed by reducing redundant predictions. Extensive experiments conducted on several challenging benchmarks have demonstrated the superior accuracy and efficiency of TextRSR compared to state-of-the-art methods. Given its proven effectiveness and efficiency, we intend to extend TextRSR to scene text spotting in future work.

REFERENCES

[1] C. Yi and Y. Tian, "Scene text recognition in mobile applications by character descriptor and structure configuration," *IEEE Transactions on Image Processing*, vol. 23, no. 7, pp. 2972–2982, 2014.

[2] Z. Shao, H. Zhu, Y. Zhou, X. Xiang, B. Liu, R. Yao, and L. Ma, "Facial action unit detection by adaptively constraining self-attention and causally deconfounding sample," *International Journal of Computer Vision*, vol. 133, no. 4, pp. 1711–1726, 2025.

[3] B. Xiong and K. Grauman, "Text detection in stores using a repetition prior," in *IEEE Winter Conference on Applications of Computer Vision*, 2016, pp. 1–9.

[4] Z. Hong, Y. Petillot, D. Lane, Y. Miao, and S. Wang, "Textplace: Visual place recognition and topological localization through reading scene texts," in *IEEE International Conference on Computer Vision*, 2019, pp. 2861–2870.

[5] 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*, vol. 25, pp. 5030–5042, 2023.

[6] Y. Liu, C. Shen, L. Jin, T. He, P. Chen, C. Liu, and H. Chen, "Abcnet v2: Adaptive bezier-curve network for real-time end-to-end text spotting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp. 8048–8064, 2021.

[7] D. Deng, H. Liu, X. Li, and D. Cai, "Pixellink: Detecting scene text via instance segmentation," in *AAAI Conference on Artificial Intelligence*, 2018, pp. 6773–6780.

[8] Z. Tian, M. Shu, P. Lyu, R. Li, C. Zhou, X. Shen, and J. Jia, "Learning shape-aware embedding for scene text detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4234–4243.

[9] 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.

[10] 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 *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9336–9345.

[11] M. Liao, Z. Wan, C. Yao, K. Chen, and X. Bai, "Real-time scene text detection with differentiable binarization," in *AAAI Conference on Artificial Intelligence*, 2020, pp. 11 474–11 481.

[12] 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, 2022.

[13] S. Long, J. Ruan, W. Zhang, X. He, W. Wu, and C. Yao, "Textsnake: A flexible representation for detecting text of arbitrary shapes," in *European Conference on Computer Vision*, 2018, pp. 20–36.

[14] J. Tang, Z. Yang, Y. Wang, Q. Zheng, Y. Xu, and X. Bai, "Seglink++: Detecting dense and arbitrary-shaped scene text by instance-aware component grouping," *Pattern Recognition*, vol. 96, p. 106954, 2019.

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

[16] P. Dai, S. Zhang, H. Zhang, and X. Cao, "Progressive contour regression for arbitrary-shape scene text detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 7393–7402.

[17] 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 *AAAI Conference on Artificial Intelligence*, 2020, pp. 12 160–12 167.

[18] X. Wang, Y. Jiang, Z. Luo, C.-L. Liu, H. Choi, and S. Kim, "Arbitrary shape scene text detection with adaptive text region representation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6449–6458.

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

[20] F. Wang, Y. Chen, F. Wu, and X. Li, "Textray: Contour-based geometric modeling for arbitrary-shaped scene text detection," in *ACM International Conference on Multimedia*, 2020, pp. 111–119.

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

[22] W. Wang, Y. Zhou, J. Lv, D. Wu, G. Zhao, N. Jiang, and W. Wang, "Tpsnet: Reverse thinking of thin plate splines for arbitrary shape scene text representation," in *ACM International Conference on Multimedia*, 2022, pp. 5014–5025.

[23] Y. Su, Z. Chen, Z. Shao, Y. Du, Z. Ji, J. Bai, Y. Zhou, and Y.-G. Jiang, "Lranet: Towards accurate and efficient scene text detection with low-rank approximation network," in *AAAI Conference on Artificial Intelligence*, 2024, pp. 4979–4987.

[24] 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 *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9809–9818.

[25] Z. Raisi, G. Younes, and J. Zelek, "Arbitrary shape text detection using transformers," in *International Conference on Pattern Recognition*, 2022, pp. 3238–3245.

[26] X. Zhang, Y. Su, S. Tripathi, and Z. Tu, "Text spotting transformers," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2022, pp. 9519–9528.

[27] Z. Shao, Y. Su, Y. Zhou, F. Meng, H. Zhu, B. Liu, and R. Yao, "Ct-net: arbitrary-shaped text detection via contour transformer," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 3, pp. 1815–1826, 2024.

[28] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European Conference on Computer Vision*, 2020, pp. 213–229.

[29] T. Ding, J. Zhou, T. Chen, Z. Zhu, I. Zharkov, and L. Liang, "Ada-contour: Adaptive contour descriptor with hierarchical representation," *arXiv preprint arXiv:2404.08292*, 2024.

[30] G. Lerman and T. Maunu, "An overview of robust subspace recovery," *Proceedings of the IEEE*, vol. 106, no. 8, pp. 1380–1410, 2018.

[31] Z. Zhu, T. Ding, D. Robinson, M. Tsakiris, and R. Vidal, "A linearly convergent method for non-smooth non-convex optimization on the grassmannian with applications to robust subspace and dictionary learning," *Advances in Neural Information Processing Systems*, pp. 9442 – 9452, 2019.

[32] S. X. Zhang, X. Zhu, J. B. Hou, C. Liu, and X. C. Yin, "Deep relational reasoning graph network for arbitrary shape text detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9696–9705.

[33] C. Ma, L. Sun, Z. Zhong, and Q. Huo, "Relatext: Exploiting visual relationships for arbitrary-shaped scene text detection with graph convolutional networks," *Pattern Recognition*, vol. 111, p. 107684, 2021.

[34] C. K. Chng, Y. Liu, Y. Sun, C. C. Ng, C. Luo, Z. Ni, C. Fang, S. Zhang, J. Han, E. Ding *et al.*, "Icdar2019 robust reading challenge on arbitrary-shaped text-rrc-art," in *International Conference on Document Analysis and Recognition*, 2019, pp. 1571–1576.

[35] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2980–2988.

[36] R. Stewart, M. Andriluka, and A. Y. Ng, "End-to-end people detection in crowded scenes," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2325–2333.

[37] A. Shrivastava, A. Gupta, and R. Girshick, "Training region-based object detectors with online hard example mining," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 761–769.

[38] Y. Liu, L. Jin, S. Zhang, C. Luo, and S. Zhang, "Curved scene text detection via transverse and longitudinal sequence connection," *Pattern Recognition*, vol. 90, pp. 337–345, 2019.

[39] C. K. Ch'ng and C. S. Chan, "Total-text: A comprehensive dataset for scene text detection and recognition," in *International Conference on Document Analysis and Recognition*, vol. 1, 2017, pp. 935–942.

[40] 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 *International Conference on Document Analysis and Recognition*, 2015, pp. 1156–1160.

[41] Z. Kuang, H. Sun, Z. Li, X. Yue, T. H. Lin, J. Chen, H. Wei, Y. Zhu, T. Gao, W. Zhang, K. Chen, W. Zhang, and D. Lin, "Mmocr: A comprehensive toolbox for text detection, recognition and understanding," in *ACM International Conference on Multimedia*, 2021, pp. 3791–3794.

[42] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2019, pp. 8024–8035.

[43] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2315–2324.

[44] 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 *International Conference on Document Analysis and Recognition*, vol. 1, 2017, pp. 1454–1459.

[45] A. Veit, T. Matera, L. Neumann, J. Matas, and S. Belongie, "Coco-text: Dataset and benchmark for text detection and recognition in natural images," *arXiv preprint arXiv:1601.07140*, 2016.

[46] N. Nayef, Y. Patel, M. Busta, P. N. Chowdhury, D. Karatzas, W. Khlif, J. Matas, U. Pal, J.-C. Burie, C.-l. Liu *et al.*, "Icdar2019 robust reading challenge on multi-lingual scene text detection and recognition—rrc-mlt-2019," in *International Conference on Document Analysis and Recognition*, 2019, pp. 1582–1587.

[47] C. Xue, S. Lu, and W. Zhang, "Msr: Multi-scale shape regression for scene text detection," in *International Joint Conference on Artificial Intelligence*, 2019, pp. 989 – 995.

[48] 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 *IEEE International Conference on Computer Vision*, 2019, pp. 8440–8449.

[49] 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, 2022.

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

[51] 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 *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 753–11 762.

[52] 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, 2020.

[53] 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.

[54] X. Liu, D. Liang, S. Yan, D. Chen, Y. Qiao, and J. Yan, "Fots: Fast oriented text spotting with a unified network," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5676–5685.

[55] P. Lyu, M. Liao, C. Yao, W. Wu, and X. Bai, "Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes," in *European Conference on Computer Vision*, 2018, pp. 67–83.

[56] 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 *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9336–9345.

[57] M. Zhao, W. Feng, F. Yin, X.-Y. Zhang, and C.-L. Liu, "Mixed-supervised scene text detection with expectation-maximization algorithm," *IEEE Transactions on Image Processing*, vol. 31, pp. 5513–5528, 2022.

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

[59] Z. Tian, H. Chen, X. Wang, Y. Liu, and C. Shen, "AdelaiDet: A toolbox for instance-level recognition tasks," https://git.io/adelaidet, 2019.

[60] Y. Zhu and J. Du, "Textmountain: Accurate scene text detection via instance segmentation," *Pattern Recognition*, vol. 110, p. 107336, 2021.

[61] X. Zhao, W. Feng, Z. Zhang, J. Lv, X. Zhu, Z. Lin, J. Hu, and J. Shao, "Cbnet: A plug-and-play network for segmentation-based scene text detection," *International Journal of Computer Vision*, vol. 132, no. 8, pp. 3119–3138, 2024.

[62] E. Xie, Y. Zang, S. Shao, G. Yu, C. Yao, and G. Li, "Scene text detection with supervised pyramid context network," in *AAAI Conference on Artificial Intelligence*, 2019, pp. 9038–9045.

[63] M. Xing, H. Xie, Q. Tan, S. Fang, Y. Wang, Z.-J. Zha, and Y. Zhang, "Boundary-aware arbitrary-shaped scene text detector with learnable embedding network," *IEEE Transactions on Multimedia*, vol. 24, pp. 3129–3143, 2022.

[64] 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.

[65] M. He, M. Liao, Z. Yang, H. Zhong, J. Tang, W. Cheng, C. Yao, Y. Wang, and X. Bai, "Most: A multi-oriented scene text detector with localization refinement," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8813–8822.
[66] M. Skrodzki, "The k-d tree data structure and a proof for neighborhood computation in expected logarithmic time," *arXiv preprint arXiv:1903.04936*, 2019.

Zhiwen Shao is currently an Associate Professor with the China University of Mining and Technology, China. He received the B.Eng. degree and the Ph.D. degree in Computer Science and Technology from the Northwestern Polytechnical University, China and the Shanghai Jiao Tong University, China in 2015 and 2020, respectively. His research interests lie in computer vision and affective computing. He has served as an Area Chair for ACM MM, an Associate Editor for TVC, and a Publication Chair for CGI.

Shengtian Jiang is currently a master student at the School of Computer Science and Technology, China University of Mining and Technology, China. His research interests include text detection, text recognition, and layout analysis.

Hancheng Zhu is currently an Associate Professor with the School of Computer Science and Technology, China University of Mining and Technology, China. He received the B.S. degree from the Changzhou Institute of Technology, Changzhou, China, in 2012, and the M.S. and Ph.D. degrees from the China University of Mining and Technology, Xuzhou, China, in 2015 and 2020, respectively. His research interests include image aesthetics assessment and affective computing.

Xuehuai Shi is currently a Lecturer with the School of Computer Science, Nanjing University of Posts and Telecommunications, China. He received the Ph.D. degree from the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, China. His research interests include virtual reality, real-time rendering, and augmented reality.

Canlin Li is currently a Professor with the School of Computer Science and Technology, Zhengzhou University of Light Industry, China. He received the B.S. degree from National University of Defense Technology, China in 1998, and the M.S. degree from Zhejiang University, China in 2004, and the Ph.D. degree from Shanghai Jiao Tong University, China in 2010. His research interests include image processing, pattern recognition, artificial intelligence, and visual media computing.

Lizhuang Ma is currently a Distinguished Professor with the School of Computer Science, Shanghai Jiao Tong University, China. He is the recipient of the National Science Fund for Distinguished Young Scholars. He received the B.S. and Ph.D. degrees from the Zhejiang University, China in 1985 and 1991, respectively. His research interests include computer graphics, digital media technology, and theory and applications for computer graphics, CAD/CAM.

Dit-Yan Yeung is currently a Chair Professor with the Department of Computer science and Engineering, The Hong Kong University of Science and Technology, Hong Kong. He received his B.Eng. degree in Electrical Engineering and MPhil degree in Computer Science from the University of Hong Kong, Hong Kong, and Ph.D. degree in Computer Science from the University of Southern California, USA. His research interests are primarily in computational and statistical approaches to machine learning and artificial intelligence.