HGR-Net: Hierarchical Graph Reasoning Network for Arbitrary Shape Scene Text Detection

Hengyue Bi, Canhui Xu, Cao Shi, Guozhu Liu, Honghong Zhang, Yuteng Li, and Junyu Dong, *Member, IEEE*

***Abstract*— As a prerequisite step of scene text reading, scene text detection is known as a challenging task due to natural scene text diversity and variability. Most existing methods either adopt bottom-up sub-text component extraction or focus on top-down text contour regression. From a hybrid perspective, we explore hierarchical text instance-level and component-level representation for arbitrarily-shaped scene text detection. In this work, we propose a novel Hierarchical Graph Reasoning Network (HGR-Net), which consists of a Text Feature Extraction Network (TFEN) and a Text Relation Learner Network (TRLN). TFEN adaptively learns multi-grained text candidates based on shared convolutional feature maps, including instance-level text contours and component-level quadrangles. In TRLN, an inter-text graph is constructed to explore global contextual information with position-awareness between text instances, and an intra-text graph is designed to estimate geometric attributes for establishing component-level linkages. Next, we bridge the cross-feed interaction between instance-level and component-level, and it further achieves hierarchical relational reasoning by learning complementary graph embeddings across levels. Experiments conducted on three publicly available benchmarks SCUT-CTW1500, Total-Text, and ICDAR15 have demonstrated that HGR-Net achieves state-of-the-art performance on arbitrary orientation and arbitrary shape scene text detection.**

***Index Terms*— Scene text detection, arbitrary shape text, hierarchical relation modeling, graph convolutional network.**

I. Introduction

TEXT instances in scene images carry valuable information and provide concise expression for the understanding of human reading. Scene text detection is a fundamental and prerequisite task in automatic scene text reading and has been utilized in various real-world multimedia applications, such as image retrieval, office automation, and robot navigation. Scene text detection is still challenging, due to aspect ratio variability, perspective distortion diversity, and arbitrary shapes of the text instances. To cope with the aforementioned challenges, early works use specific handcraft features for scene text detection. Constrained by the complexity of redundant intermediate processes and heavy heuristics rules, these methods can hardly handle intricate situations, especially arbitrarily-shaped scene text detection.

Benefitting from prosperity of object detection and segmentation with deep learning, the process of arbitrarily-shaped scene text detection has been greatly accelerated. Recent researches could be roughly summarized as two groups: bottom-up methods [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15] and top-down methods [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37]. Bottom-up methods apply deep neural network to densely extract segments or regional attributes, and subsequently reconstruct the extracted components to different text instances by post-processing. Consequently, bottom-up methods usually neglect modeling global geometric context from overall layout. Besides, complicated post-processing step is inevitable and time-consuming.

For top-down methods, previous works [16], [19], [20], [22] try to crop oriented text instances based on modified region proposal networks. Although these methods exploit text orientation angle information for proposal generation, they might be insensitive to detect arbitrary shape texts due to limited respective fields. Recent works [26], [27], [28], [29] are proposed to directly learn parameterized text contours for curved and oriented text detection. Interestingly, one of the notable characteristics is that text instances can be regarded as an ordered series of homogeneous components [38]. Thus, top-down methods tend to overlook implicit text geometric distribution in natural scenes due to insensibility to textual attributes of sub-text components.

Therefore, either exploiting geometric attributes from partial text components or modeling global context between holistic text instances is of importance for arbitrary shape scene text detection. However, few efforts have been made on both component-level and instance-level detection to fully exploit potential information among text layout and locality. Though, Mask TextSpotter [21] is proposed to handle text instances with arbitrary shapes via an instance-level detector and another separate character-level recognizer, it might ignore intuitive and vital interaction between two levels. Motivated by these

Manuscript received 1 October 2022; revised 26 February 2023; accepted 22 June 2023. Date of publication 17 July 2023; date of current version 21 July 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 61806107 and Grant 61702135, in part by the Shandong Key Laboratory of Wisdom Mine Information Technology, and in part by the Opening Project of State Key Laboratory of Digital Publishing Technology. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Xiangqian Wu. *(Hengyue Bi and Canhui Xu contributed equally to this work.)* *(Corresponding author: Guozhu Liu.)*

Hengyue Bi, Canhui Xu, Cao Shi, Guozhu Liu, Honghong Zhang, and Yuteng Li are with the School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao 266100, China (e-mail: lgz_0228@163.com).

Junyu Dong is with the Faculty of Information Science and Engineering, Ocean University of China, Qingdao 266100, China.

Digital Object Identifier 10.1109/TIP.2023.3294822

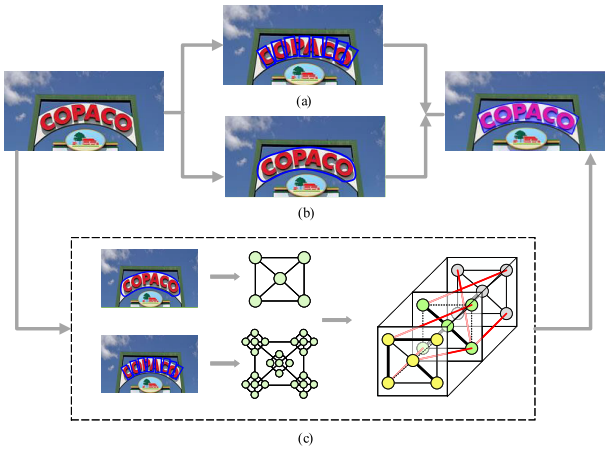


Fig. 1. Illustration of different text detection methods. Detection results are shown with blue bounding boxes. (a) bottom-up methods. (b) top-down methods. (c) HGR-Net: HGR-Net generates multi-grained text candidates and exploits intra-text and inter-text graph embeddings for hierarchical graph reasoning.

issues, we construct an intra-text graph and an inter-text graph. As shown in Figure 1(c), graph learners receive the constructed graph data, which updates two-level graph embeddings by capturing global contextual information and implicit spatial relationship, respectively. Moreover, we bridge the interaction between component-level and instance-level for hierarchical relational reasoning.

In this paper, we propose a novel Hierarchical Graph Reasoning Network (HGR-Net) for accurate and robust scene text detection, where HGR-Net mainly consists of two sub-networks: Text Feature Extraction Network (TFEN) and Text Relation Learner Network (TRLN). To begin with, motivated by ABC-Net [26] and TextSnake [2], the input text instances will be regarded as a combination of instance-level Bezier curves, component-level serial quadrangles, and bounding boxes with annotated categories. Hence, TFEN utilizes shared convolutional features to generate multiple descriptors with geometric attributes, and provides aligned feature representation. Then, TRLN integrates three modules including Inter-Text Global Contextual Module (Inter-Text GCM), Intra-Text Spatial Layout Module (Intra-Text SLM), and Hierarchical Relational Reasoning Module (HRRM). For the given multi-grained text candidates, Intra-Text SLM constructs an intra-text graph to establish linkages from spatial layout perspective, while Inter-Text GCM constructs an inter-text graph to explore instance-level contextual information. HRRM achieves hierarchical relational reasoning by learning complementary graph embeddings with cross-feed interaction.

The main contributions of this paper are summarized as follows:

- We propose a novel end-to-end Hierarchical Graph Reasoning Network (HGR-Net) for arbitrary shape scene text detection, which has achieved state-of-the-art performance on various publicly available benchmarks.
- TFEN can adaptively extract multi-grained text candidates and textual contents based on shared convolutional feature maps.
- TRLN integrates Intra-Text SLM to establish component-level spatial relationship and Inter-Text GCM to explore instance-level contextual information. And it further achieves relation aggregation and cross-feed interaction with aid of hierarchical design, which could restrain the effects of occluded, redundant and ambiguous situations.
- To the best of our knowledge, HGR-Net is one of the very first attempts to perform hierarchical linkage likelihood deduction across levels for arbitrary shape scene text detection, while keeping a proper real-time inference speed.

II. Related Work

Undoubtedly, scene text detection, as a preliminary step of text reading, has attracted considerable attention due to challenges on localizing diverse and complex text instances. Hence, efforts have been made on increasing efficiency and improving accuracy. In this section, we briefly review related studies, including early works based on engineered features and recent approaches based on deep learning. We encourage readers to refer to recent comprehensive and detailed surveys on scene text detection methods [38], [39].

The early researches mainly focus on connected component analysis (CCA)-based methods [40], [41], [42], [43] and sliding window classification (SLC)-based methods [44], [45], [46], [47] to deal with regular horizontal labeled datasets, such as ICDAR03 [48], ICDAR11 [49], and ICDAR13 [50]. CCA-based methods first detect text components via a variety of methods, such as edge detection or extreme region extraction, and then stack potential candidates into holistic text instances by utilizing heuristic rules. And SLC-based methods adopt multiple sliding windows to extract positive text candidates and group them into text regions. However, early works heavily rely on pre-processing and post-processing steps, which are insufficient to handle more intricate situations, such as arbitrary orientation benchmarks [41], [51], [52] and arbitrary shape benchmarks [24], [53].

Recently, benefitting from the advancement of deep learning, the performance of scene text detection has been greatly improved. Bottom-up methods [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15] mainly utilize fully convolutional network (FCN) to densely generate text prediction maps, and then post-processing algorithms are applied for grouping inclined region candidates. TextSnake [2] describes text instances as a sequence of ordered sliding round disks, and uses FCN to estimate geometric relation of sub-text components for final prediction. TextDragon [5] is designed to describe text instances as a series of quadrangles to handle arbitrarily-shaped text detection. DRRG [12] firstly uses CNN to predict geometric attributes of divided text components, and then GCN is to perform linkage deduction for stacking the components together. CentripetalText [13] divides text instances into combinations of text kernels and centripetal shifts which are implemented and clustered for text contour generation. Text instances are generated by estimating linkages among text units. However, bottom-up methods usually neglect geometric context of holistic text instances. And inevitable component aggregation complicates the overall pipelines.

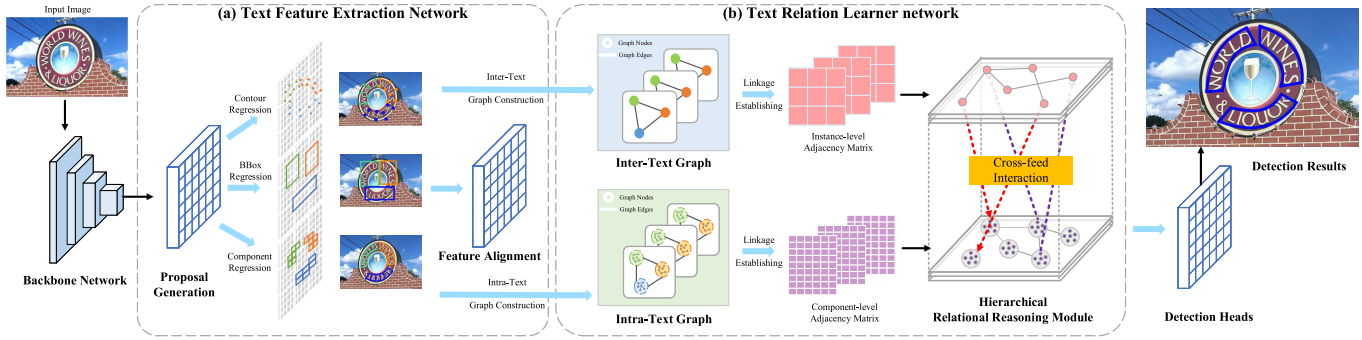


Fig. 2. The overall architecture of HGR-Net. It consists of two pipelines, including Text Feature Extraction Network (TFEN) and Text Relation Learner Network (TRLN). (a) TFEN generates multi-grained text candidates and extracts regional feature representation based on shared convolutional feature maps. (b) TRLN first utilizes multi-grained textual attributes to construct an instance-level inter-text graph and a component-level intra-text graph. Then, Hierarchical Relational Reasoning Module (HRRM) guides graph linkages to learn cross-feed interaction.

Top-down methods [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37] treat scene text detection as a special type of object detection. Previous methods [16], [19], [20], [22] use modified anchor-mechanisms to detect horizontal and arbitrarily-oriented text instances. Textboxes++ [20] adopts FCN to detect arbitrarily-oriented scene text instances. Based on SSD framework [54], Textboxes++ [20] modifies axis-aligned bounding box generation and utilizes quadrilateral bounding boxes to represent text instances with arbitrary orientation. In RRPN [22], Rotation Region Proposal Network (RRPN) and Rotation Region-of-Interest (RROI) are adapted to predict inclined region proposals. Dai et al. [32] put forward an arbitrarily-oriented detector with Anchor-free Corner Evolution (ACE), which progressively predicts compact quadrilateral text boxes by dynamic information gathering. As the aspect ratios and layouts vary significantly in the wild, horizontal and arbitrarily-oriented text detectors are incapable of fitting text instances with arbitrary shapes. Thus, recent works [26], [27], [28], [29] have focused on contour regression to face with arbitrarily-shaped benchmarks. ABCNet [26] regresses control points for approximating curved text contours progressively, so as to detect text instances with Bezier curve adaptation. FCE-Net [28] first formulates irregular text instances with Fourier signature vectors, and then reconstructs contour points in the image spatial domain. TextBPN [27] consists of a boundary proposal module and an adaptive boundary deformation module, which roughly generates text candidates and progressively performs boundary refinement. However, recent contour-based methods ignore implicit geometric distribution of exploiting homogeneous text components.

Additionally, with the rise and development of Transformer [55] in computer vision committee, top-down Transformer-based detectors [30], [33], [34], [35] have been proposed to detect text instances in the wild, which usually adopt ViT [56] or DETR [57] and their variances as the basic framework. For instance, Zhou et al. [35] present Aggregated Text TRansformer (ATTR) for exploiting interaction across different scales. A set of images in different scales are firstly projected into sequences of flatten patches. Then, the transformer encoder extracts multi-scale feature representation and generates binary text masks as text embeddings. The scale-wise decoder aggregates features across different scales and output detection queries. Long et al. [36] present a novel network Unified Detector for simultaneous scene text detection and layout analysis. MaX-DeepLab [58] is taken as the backbone network to extract image features and queries for detection and layout analysis branches, respectively. Because of lacking inductive biases and relying on large-scale training, Transformer-based methods are faced with the challenge of high computational cost.

From a hybrid perspective, we propose a novel detector HGR-Net for arbitrarily-shaped text detection in a hierarchical manner. Each input text instance is regarded as a combination of instance-level Bezier curve and component-level ordered quadrangles. HGR-Net first estimates geometric attributes of multi-grained text candidates, and then constructs an inter-text graph and an intra-text graph for establishing linkages. To achieve hierarchical relational reasoning, it further bridges the cross-feed interaction and conducts complementary graph embedding updating.

III. Our Approach

The overall architecture of HRG-Net is shown in Figure 2. It is composed of two pipelines, including Text Feature Extraction Network (TFEN) and Text Relation Learner Network (TRLN). Functionally, Text Feature Extraction Network (TFEN) generates multi-grained text candidates and extracts regional feature representation based on shared convolutional feature maps. Text Relation Learner Network (TRLN) constructs an inter-text graph and an intra-text graph. Inter-text graph is to capture instance-level contextual information with position-awareness from visual connections, and intra-text graph is to estimate component-level geometric attributes for establishing spatial linkages. After two-level graph construction, Hierarchical Relational Reasoning Module (HRRM) guides graph linkages to cross-feed interaction, which aims at learning complementary graph embeddings with integrated spatial-related and contextual information.

A. Text Feature Extraction Network

To detect text instances with arbitrary shapes, Text Feature Extraction Network (TFEN) is proposed to exploit multiple

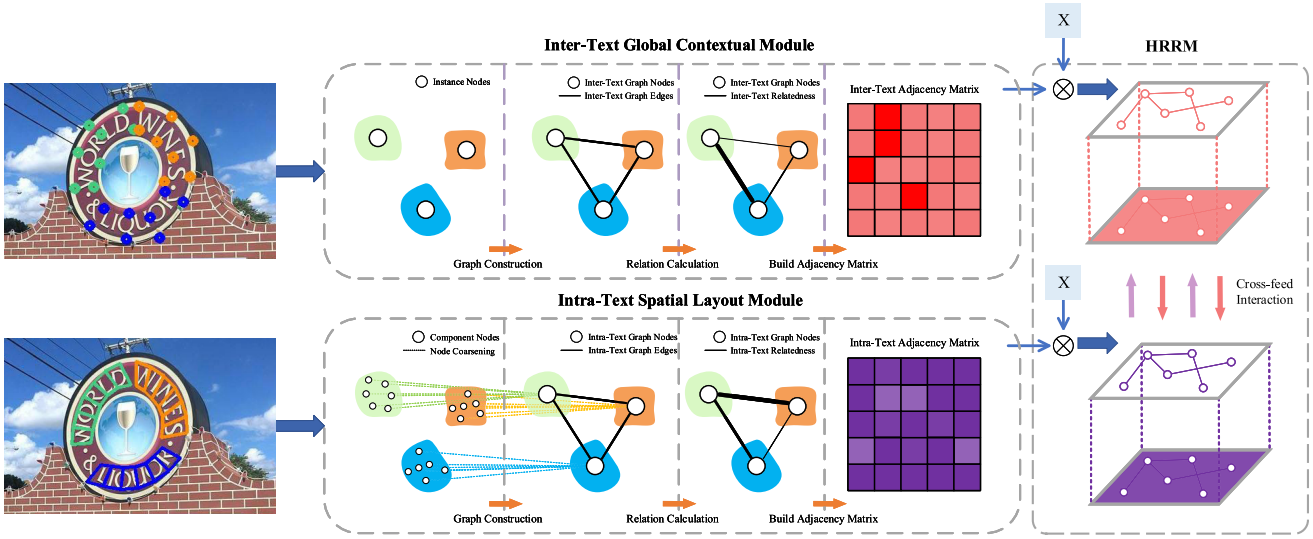


Fig. 3. Illustration of Text Relation Learner Network. Inter-Text Global Contextual Module (Inter-Text GCM) constructs inter-text graph to encode instance-level contextual information, and Intra-Text Spatial Layout Module (Intra-Text SLM) constructs intra-text graph to capture spatial layout relationship from component-level geometric attributes. After graph construction, Hierarchical Relational Reasoning Module (HRRM) learns graph embeddings by cascading the association between inter-text graph and intra-text graph.

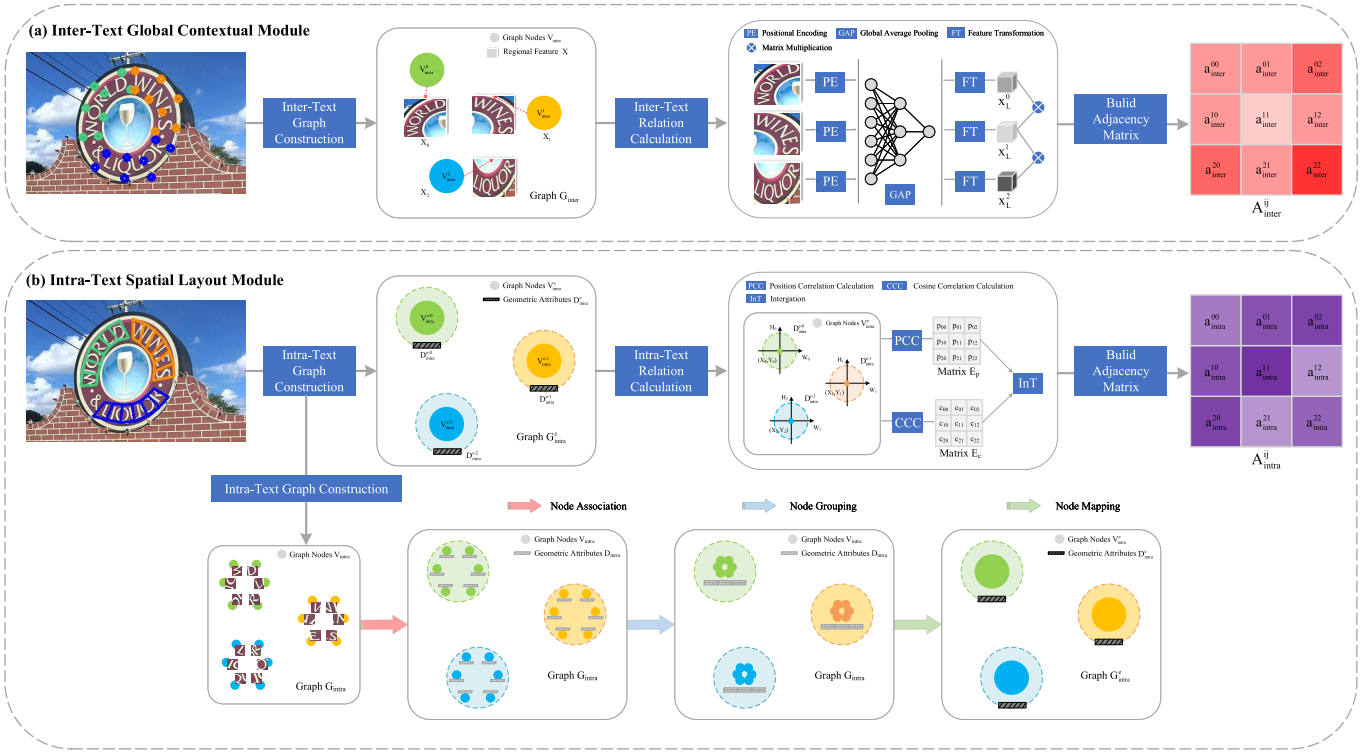


Fig. 4. Illustration of Inter-Text GCM and Intra-Text SLM. (a) Inter-Text GCM explores contextual information with position awareness, and inter-text adjacency matrix A_{inter} can be calculated. (b) Intra-Text SLM first constructs the initial intra-text graph. And then graph nodes with geometric attributes are aggregated, and the coarsened intra-text graph can be obtained. The captured spatial relationship is applied to build intra-text adjacency matrix A_{intra}.

descriptors with geometric attributes. After extracting convolutional feature maps by ResNet-50 [59] with FPN [60], TFEN generates text candidates for the input images, which learns to regress control points of parameterized Bezier curves, quadrilateral boxes of text components, and bounding boxes with categories. Subsequently, feature alignment is applied to extract regional feature representation.

Conventional anchor-based detectors adopt a variety of anchor boxes generated by Region Proposal Networks (RPN) [61] as reference to perform box regression and classification. However, horizontal anchor boxes proposed by RPN are mainly applied to regular object detection, which are insufficiently robust for irregular text instances. Recent ABC-Net [26] models text instances by Bezier curves, leading to considerable improvement in arbitrarily-shaped scene text detection. Inspired by ABC-Net, parameterized Bezier curve is employed to represent text instances, and instance-level branch is developed to regress control points of cubic Bezier curves.

To explore component-level branch, we refer to TextSnake [2] and TextDragon [5] to divide each text instance into a series of ordered quadrilateral boxes.

We define a multi-task loss function for an input scene text image, which is formulated as,

$$L = L_{\text{cls}} + \lambda_1 L_{\text{bbox}} + \lambda_2 L_{\text{contour}} + \lambda_3 L_{\text{component}}, \quad (1)$$

where L_{cls} and L_{bbox} are the loss functions for classification and bounding box regression, which are identical as in [61]. Anchors are classified as either positive samples or negative samples. Specifically, for the positive anchors, we adapt $smooth_{L_1}$ loss function for text contour and component regression. It is defined as,

$$\begin{aligned} L_{\text{contour}}(c_i^*, c_i) &= \frac{1}{T}\sum_i p_i^* smooth_{L_1}\left(c_i^* - c_i\right), \\ L_{\text{component}}(f_i^*, f_i) &= \frac{1}{K}\sum_i p_i^* smooth_{L_1}\left(f_i^* - f_i\right), \\ smooth_{L_1}(x) &= \begin{cases} 0.5\ x^2, & \text{if } |x| < 1, \\ |x| - 0.5, & \text{otherwise.} \end{cases} \end{aligned} \quad (2)$$

Here, i is the anchor index, and p_i represents predicted probability of positive anchor i. Vector c_i is the predicted control points, and vector c_i^* corresponds to the control points of labeled ground truth contour. Vector f_i is the predicted quadrilateral boxes, and vector f_i^* corresponds to the labeled ground truth components. T is the number of pair-wise control points, and K corresponds to the number of ordered quadrilateral boxes. Thus, for each positive anchor, the parameterizations of tuples (c_i, c_i^*) and (f_i, f_i^*) can be calculated as follows,

$$\begin{aligned} f_{ix} &= (x_i^f - x_i^a)/w_i^a, \quad f_{iy} = (y_i^f - y_i^a)/h_i^a, \\ f_{iw} &= \log(w_i^f/w_i^a), \quad f_{ih} = \log(h_i^f/h_i^a), \\ f_{ix}^* &= (x_i^{f*} - x_i^a)/w_i^a, \quad f_{iy}^* = (y_i^{f*} - y_i^a)/h_i^a, \\ f_{iw}^* &= \log(w_i^{f*}/w_i^a), \quad f_{ih}^* = \log(h_i^{f*}/h_i^a), \end{aligned} \quad (3)$$

$$\begin{aligned} c_{ix} &= (x_i^c - x_i^a)/w_i^a, \quad c_{iy} = (y_i^c - y_i^a)/h_i^a, \\ c_{ix}^* &= (x_i^{c*} - x_i^a)/w_i^a, \quad c_{iy}^* = (y_i^{c*} - y_i^a)/h_i^a, \end{aligned} \quad (4)$$

where x_i^a, y_i^a, w_i^a, and h_i^a denote coordinate, width, and height of anchor i, respectively. Variables (x_i^c, y_i^c) and (x_i^{c*}, y_i^{c*}) are predicted control points and labeled control points. Variables x_i^f and x_i^{f*} are predicted components and labeled components (likewise for y_i^f, w_i^f, and h_i^f).

Motivated by Mask TextSpotter [21], RoIAlign is adapted to extract text feature representation, and the extracted aligned features are subsequently shared with parallel instance-level and component-level branches, which not only simplifies overall pipeline, but also reduces training complexity.

B. Text Relation Learner Network

In real world scenarios, it is natural for humans to recognize scene texts by exploiting hierarchical relationship empirically, which is beneficial for detecting scene texts in various complex situations. However, few studies have focused on jointly learning instance-level and component-level information. The integration of partial and holistic cues could bring better understanding of the detection [62]. To this end, in Figure 3, Text Relation Learner Network (TRLN) constructs an Inter-Text Global Contextual Module (Inter-Text GCM) for encoding global contextual information from instance-level visual connection. And an Intra-Text Spatial Layout Module (Intra-Text SLM) encodes spatial layout relationship from geometric attributes among text components. After graph construction, Hierarchical Relational Reasoning Module (HRRM) is to learn complementary graph embeddings for inferring linkages.

1) Inter-Text Global Contextual Module: Illustration of Inter-Text Global Contextual Module (Inter-Text GCM) is depicted in Figure 4(a). We firstly define a dynamic graph $G_{inter} = (V_{inter}, E_{inter})$ to explore global context information with position awareness, where $V_{inter} \in \mathbb{R}^N$ are inter-text graph nodes, $E_{inter} \in \mathbb{R}^{N\times N}$ represent inter-text graph edges, and N is the number of instance-level candidates. Given regional features $X \in \mathbb{R}^{N\times C\times W\times H}$ extracted from RoIAlign, learnable Position Encoding (PE) [55] is applied on the feature matrix X. Global Average Pooling (GAP) is then used to squeeze spatial dimension.

$$X_g = \text{GAP}(X^{enc}) \in \mathbb{R}^{N\times C_2\times H_2\times W_2}. \quad (5)$$

Here, N is the number of text instances, X_g is the output feature matrix with channel dimension C_2, GAP is implemented by 2D adaptive average pooling, and X^{enc} is the feature matrix X with positional encoding. To encode contextual information, the reshaped feature matrix $X_g^r \in \mathbb{R}^{N\times C_2H_2W_2}$ are further transformed into a latent space by non-linear function $\phi(\cdot)$, which can be formulated as,

$$X_L = \phi(X_g^r) \in \mathbb{R}^{N\times L}, \quad (6)$$

where L is the dimension of transformed latent space. Thus, we calculate the adjacency matrix A_{inter}, which is defined as,

$$A_{inter} = X_L \otimes X_L^T \in \mathbb{R}^{N\times N}, \quad (7)$$

where $X_L^T \in \mathbb{R}^{L\times N}$ is the transpose matrix of X_L.

2) Intra-Text Spatial Layout Module: The intra-text graph $G_{intra} = (V_{intra}, E_{intra})$ is to establish spatial-related linkages among generated text components, where $V_{intra} \in \mathbb{R}^{N_2}$ represent intra-text graph nodes, and $E_{intra} \in \mathbb{R}^{N_2\times N_2}$ are intra-text graph edges, and N_2 is the number of component-level candidates. As shown in Figure 4(b), graph nodes $V_{intra} \in \mathbb{R}^{N_2}$ with geometric attributes $D_{intra} \in \mathbb{R}^{N_2\times d}$ belong to same text instances will be associated and grouped up. We then aggregate graph nodes $V_{intra} \in \mathbb{R}^{N_2}$ to high-level $V_{intra}^e \in \mathbb{R}^N$, and the mean-pooling operation is used to map geometric attributes $D_{intra} \in \mathbb{R}^{N_2\times d}$ to $D_{intra}^e \in \mathbb{R}^{N\times d_2}$. Thus, we generate the coarsened intra-text graph $G_{intra}^e = (V_{intra}^e, E_{intra}^e)$.

Based on G_{intra}^e, position correlation matrix $E_p(i, j)$ between graph node i and j is computed as follows,

$$E_p(i, j) = \tau(ED((x_i, y_i), (x_j, y_j))) \in \mathbb{R}^{N\times N}, \quad (8)$$

where $ED(\cdot)$ is the Euclidean distance calculation. $\tau(\cdot)$ not only normalizes input variables to $[0, 1]$, but also extends a negative proportion. The intuition behind is that more correlated components are closer in spatial layout [63].

Moreover, $E_c(i, j)$ is the cosine correlation matrix to describe angle distance between graph nodes. It can be written by,

$$E_c(i, j) = \frac{(x_i, y_i) \cdot (x_j, y_j)}{\|(x_i, y_i)\| \left\|(x_j, y_j)\right\|} \in \mathbb{R}^{N \times N}. \tag{9}$$

After that, we integrate position correlation matrix $E_p(i, j)$ and cosine correlation matrix $E_c(i, j)$ to build component-level adjacency matrix A_{intra} for coarsened intra-text graph G^e_{intra}. It can be written by,

$$A_{intra} = E_p(i, j) \cup E_c(i, j) \in \mathbb{R}^{N \times N}. \tag{10}$$

3) Hierarchical Relational Reasoning Module: After graph construction, GCN learns to update graph embeddings by propagating information across the edges of the graph. Constrained by the flat structure [64], it is unable to deduce linkages in a hierarchical way. Therefore, Hierarchical Relational Reasoning Module (HRRM) is presented to bridge information interaction across instance-level and component-level, which can provide complementary graph embeddings by aggregating and inferring mutual information. After graph reasoning, feature fusion is used to integrate visual feature representation, intra-text graph embeddings and inter-text graph embeddings.

For component-level graph reasoning, we use $H^{(l)}_{intra}$ to denote the hidden feature matrix.

$$H^{l+1}_{intra} = \sigma\left(L_{intra} H^{(l)}_{intra} W^l_{intra}\right), \tag{11}$$

$$H^0_{intra} = X, \tag{12}$$

$$L_{intra} = \tilde{D_{intra}}^{-\frac{1}{2}} \tilde{A_{intra}} \tilde{D_{intra}}^{-\frac{1}{2}}. \tag{13}$$

Here, σ presents the $LeakyReLU$ activation function, W^l_{intra} is the learned weight transformation matrix of component-level reasoning, $L_{intra} \in \mathbb{R}^{N \times N}$ is the normalized Laplacian, $\tilde{A}_{intra} = A_{intra} + I$ is the intra-text adjacency matrix with added self-connections, and $\tilde{D_{intra}} \in \mathbb{R}^{N \times N}$ is the degree matrix.

For instance-level graph reasoning, intra-text graph embeddings are mapped into the component-to-instance feature space via an MLP architecture, which are fused with regional features X. The injected textual attributes provide auxiliary inner-text information. We formulate this process as follows,

$$H^{l+1}_{inter} = \sigma\left(L_{inter} H^{(l)}_{inter} W^l_{inter}\right), \tag{14}$$

$$H^0_{inter} = X \oplus \mathrm{MLP}(H^{l+1}_{intra}), \tag{15}$$

$$L_{inter} = \tilde{D_{inter}}^{-\frac{1}{2}} \tilde{A_{inter}} \tilde{D_{inter}}^{-\frac{1}{2}}, \tag{16}$$

where σ presents the $LeakyReLU$ activation function, W^l_{inter} is the learned weight transformation matrix of instance-level reasoning, $L_{inter} \in \mathbb{R}^{N \times N}$is the normalized Laplacian, $\tilde{A}_{inter} = A_{inter} + I$ is the inter-text adjacency matrix with added self-connections, $\tilde{D}_{inter} \in \mathbb{R}^{N \times N}$ is the degree matrix, and $\oplus$ is channel-wise concatenation operation.

Intra-text graph embeddings, inter-text graph embeddings and visual feature representation will be incorporated to provide complementary text representation for result prediction. This process can be summarized,

$$Z = \mathrm{FC}(X \oplus H^{l+1}_{intra} \oplus H^{l+1}_{inter}). \tag{17}$$

Here, Z, $\oplus$, and FC are the enhanced feature representation, channel-wise concatenation, and fully-connected layers liked [65], respectively. The dimensions of graph embeddings H^{l+1}_{intra} and H^{l+1}_{inter} are consistent with the dimensions of visual feature representation X.

C. Ground Truth Generation

Common arbitrarily-shaped scene text datasets, such as, SCUT-CTW1500 [24] and Total-text [53], use polygonal labels to annotate curved text contours. We extend multi-grained annotations on the original ground truth, including bounding boxes with categories, component-level ordered quadrangles, and instance-level control points of parameterized Bezier curves. Mathematically, given the annotated text instance with curved contour, it consists of a group of points $\left\{(X^p_k, Y^p_k)\right\}^K_{k=1}$, where (X^p_k, Y^p_k) represents the k^{th} point. Firstly, the corresponding horizontal bounding box $bbox_i$ can be formulated as,

$$bbox_i = \left(min(X^p_k), min(Y^p_k), max(X^p_k), max(Y^p_k)\right), i \in N, \tag{18}$$

where N is the number of annotated text instances. Then, we adapt a flexible yet simple component-level annotation generation method [2] to produce a series of ordered quadrangles. For each text instance, we calculate the vertices of top edges $E^t_n = \left\{e^t_k\right\}^{f+1}_{k=1}$ and bottom edges $E^b_n = \left\{e^b_k\right\}^{f+1}_{k=1}$. Thus, a series of quadrangles $component_i$ can be written by,

$$component_i = \left\{(e^t_k, e^t_{k+1}, e^b_k, e^b_{k+1})\right\}^f_{k=1}, i \in N, \tag{19}$$

where f is the number of text components within a text instance, and we empirically set it to 6. Finally, parameterized Bezier curves [26] are adopted to describe much more free-form text contours, which are formulated as $contour_i = \{B_k\}^m_{k=1}, n \in N$, where B_k represents the k^{th} control point and m is set to 8.

IV. Experiments

To validate the effectiveness of the proposed HGR-Net, we evaluate it on three scene text benchmarks, including one arbitrarily-oriented dataset ICDAR15 [66], two arbitrarily-shaped datasets SCUT-CTW1500 [24] and Total-Text [53].

A. Public Datasets

Total-Text is a widely used scene text dataset for arbitrary shape scene text detection released by Ch'ng et al. [53] in 2017, and each text instance is labeled with word-level polygon. It has 1255 images for training and 300 images for testing, including horizontal, arbitrarily-oriented, and arbitrarily-shaped text instances.

SCUT-CTW1500 is another vital arbitrarily-shaped scene text dataset, which is provided by Liu et al. [24]. It consists of 1500 images and over 10000 text line annotations, including

TABLE I

DETECTION RESULTS BETWEEN HGR-NET AND THIRTY STATE-OF-THE-ART METHODS ON SCUT-CTW1500, TOTAL-TEXT, AND ICDAR15. BU AND TD REPRESENT BOTTOM-UP AND TOP-DOWN METHODS, RESPECTIVELY. THE BEST RESULT OF EACH COLUMN IS HIGHLIGHTED IN **BOLD**

Method	Category	SCUT-CTW1500			Total-Text			ICDAR15		
		Recall	Precision	H-mean	Recall	Precision	H-mean	Recall	Precision	H-mean
Seglink [67]	BU	48.4	38.3	42.8	30.3	23.8	26.7	76.8	73.1	75.0
TextSnake [2]	BU	85.3	67.9	75.6	74.5	82.7	78.4	73.9	83.2	78.3
LSAE [9]	BU	77.8	82.7	80.1	-	-	-	-	-	-
TextDragon [5]	BU	82.8	84.5	83.6	75.7	85.6	80.3	-	-	-
TextField [4]	BU	79.8	83.0	81.4	79.9	81.2	80.6	80.1	84.3	82.4
Seglink++ [6]	BU	79.8	82.8	81.3	80.9	82.1	81.5	80.3	83.7	82.0
CRAFT [7]	BU	81.1	86.0	83.5	79.9	87.6	83.6	84.3	89.8	86.9
PAN [8]	BU	81.2	86.4	83.7	81.0	89.3	85.0	81.9	84.0	82.9
DRRG [12]	BU	83.0	85.9	84.5	84.9	86.5	85.7	84.7	88.5	86.6
CentripetalText [13]	BU	79.9	**88.3**	83.9	82.5	90.5	86.3	-	-	-
DMPNet [68]	TD	61.7	63.9	62.7	-	-	-	-	-	-
CTD+TLOC [24]	TD	69.8	77.4	73.4	71.0	74.0	73.0	77.1	84.5	80.6
EAST [16]	TD	49.7	78.7	60.4	50.0	36.2	42.0	78.3	83.3	80.7
Textboxes++ [20]	TD	-	-	-	-	-	-	78.5	87.8	82.9
RRPN [19]	TD	-	-	-	-	-	-	77.0	84.0	80.0
ATTR [69]	TD	80.2	80.1	80.1	76.2	80.9	78.5	83.3	90.4	86.8
MSR [70]	TD	79.0	84.1	81.5	73.0	85.2	78.6	-	-	-
PSENet-1s [71]	TD	79.7	84.8	82.2	78.0	84.0	80.9	85.5	88.7	87.1
LOMO [72]	TD	76.5	85.7	80.8	79.3	87.6	83.3	83.5	91.3	87.2
Mask TTD [18]	TD	79.0	79.7	79.4	74.5	79.1	76.7	**87.6**	86.6	87.1
Counter-Net [73]	TD	84.1	83.7	83.9	83.9	86.9	85.4	86.1	87.6	86.9
DB [74]	TD	80.2	86.9	83.4	82.5	87.1	84.7	82.7	88.2	85.4
Mask TextSpotter [21]	TD	-	-	-	82.4	88.3	85.2	87.3	86.6	87.0
ABC-Net [26]	TD	83.8	85.6	84.7	84.1	90.2	87.0	86.0	**90.4**	88.1
Boundary [75]	TD	-	-	-	85.0	88.9	87.0	-	-	-
PCR [29]	TD	82.3	87.2	84.7	82.0	88.5	85.2	-	-	-
FCE-Net [28]	TD	83.4	87.6	85.5	82.5	89.3	85.8	-	-	-
TBTD [30]	TD	-	-	-	-	-	-	78.3	89.8	83.7
ACE [32]	TD	-	-	-	-	-	-	82.6	82.1	82.4
TextBPN [27]	TD	80.6	87.7	84.0	83.3	**90.8**	86.9	-	-	-
HGR-Net	-	**85.9**	86.1	**86.0**	**86.5**	88.6	**87.5**	87.2	90.2	**88.7**

1000 training and 500 testing images. Different from Total-Text, it contains both Chinese and English text instances, and each text instance is annotated by a 14 vertices polygon.

ICDAR 2015 Incidental Text (ICDAR15) consists of a total of 1500 images collected from natural scenes, including 1000 training images and 500 testing images. Multi-orientation text instances are contained in ICDAR15 with word-level quadrangle annotations.

B. Implementation Details

HGR-Net is implemented by Pytorch framework. A common setting of ResNet-50 together with FPN is utilized as the basic backbone. For training phase, HGR-Net is trained using stochastic gradient descent (SGD) with a momentum of 0.9 and weight decay of 0.0001, and the image batch size is set to 2. HGR-Net is firstly pre-trained on SynthText150K dataset [26] (150K images including straight text images and curved text images) and ICDAR17-MLT dataset [77] (7200 images with multi-language annotation) for 270K iterations, with an initial learning rate of 0.01 which is decayed by 10 at the $160K^{th}$ iteration and $220K^{th}$ iteration. Then, HGR-Net is fine-tuned on the corresponding training set of the target datasets and the maximum iteration here is 160K. The initial learning rate is 0.001, and it is decayed by 10 at the $80K^{th}$ iteration and $120K^{th}$ iteration. The whole training process takes about 2 days. Data augmentation during training includes random crop and random scale training. We first crop input images randomly. And the cropped images are assured to avoid cropping text instances. The short edges within image are then resized to a range from 640 to 896, and the long edges are kept within 1600. For inference phase, we evaluate HGR-Net on SCUT-CTW1500, Total-Text, and ICDAR15 benchmarks. For our different multi-grained proposal generation strategy, we adopt 3 scales and 3 aspect ratios, and 100 proposals are posted after NMS.

All experiments are performed on single Nvidia GTX 2080Ti GPU and an ubuntu20.04 workstation with an Intel(R) Xeon(R) Silver 4210 2.20GHz CPU 64GB RAM.

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

In this section, HGR-Net is compared with thirty state-of-the-art methods on both arbitrarily-shaped and arbitrarily-oriented scene text benchmarks, which is shown in Table I.

(a) Total-Text

(b) SCUT-CTW1500

(c) ICDAR15

Fig. 5. Qualitative detection results of HGR-Net, which are depicted with blue masks. (a) Results on Total-Text. (b) Results on SCUT-CTW1500. (c) Results on ICDAR15.

TABLE II

COMPARISON WITH DRRG [12], ABC-NET [26], AND FCE-NET [28] ON SCUT-CTW1500 AND TOTAL-TEXT BENCHMARKS USING TIOU METRIC [76]. THE BEST RESULT OF EACH COLUMN IS HIGHLIGHTED IN **BOLD**

Method	SCUT-CTW1500			Total-Text		
	TIoU-Recall	TIoU-Precision	TIoU-H-mean	TIoU-Recall	TIoU-Precision	TIoU-H-mean
DRRG [12]	60.9	67.0	63.8	63.5	66.5	65.0
ABC-Net [26]	62.0	69.2	65.4	62.6	68.6	65.5
FCE-Net [28]	60.1	68.8	64.2	60.6	67.8	64.0
HGR-Net	**64.6**	**73.1**	**68.6**	**64.2**	**72.5**	**68.1**

1) Arbitrarily-Shaped Scene Text Detection: As mentioned above, SCUT-CTW1500 and Total-Text datasets focus on text instances with arbitrary shapes. From Table I, HGR-Net obtains 86.0% and 87.5% in terms of H-mean on SCUT-CTW1500 and Total-Text, respectively, which achieves superior detection performances. Compared with bottom-up methods, such as TextSnake [2], TextDragon [5] and CentripetalText [13], HGR-Net benefits from directly adopting contour regression to detect curved text instances, instead of relying on complicated post-processing. Mask TextSpotter [21] simultaneously predicts both text instances and characters, but it needs character-level annotations for supervision. Compared to Mask TextSpotter, HGR-Net applies two parallel branches to regress text instances and components without time-consuming character-level annotations. Moreover, we take a next important step beyond two-level textual information extraction, in which HGR-Net bridges information interaction across levels to learn hierarchically complementary features. Thus, HGR-Net has achieved 2.3% higher performance than Mask TextSpotter on Total-Text dataset.

Despite recent advancement of top-down methods, such as ABC-Net [26], Boundary [75], PCR [29], FCE-Net [28], and TextBPN [27], HGR-Net achieves more competitive performance in terms of H-mean. Particularly, HGR-Net is 0.5% superior to FCENet on SCUT-CTW1500 benchmark (86.0% versus 85.5%) and performs 0.5% better than ABC-Net on Total-Text benchmark (87.5% versus 87.0%).

Moreover, compared with GCN-based method DRRG [12] presented for exploring linkage likelihood between detected components, HGR-Net not only learns to establish component-level linkages, but also captures contextual information with position-awareness among text instances. After that, GCN

Fig. 6. Qualitative detection results of HGR-Net, DRRG [12], ABC-Net [26], and FCE-Net [28] on selected samples, which are depicted with blue polygons.

aggregates graph node linkages and propagates relation information to its neighbors, which guides HGR-Net to learn complementary graph embeddings. It is noted that HGR-Net has achieved 1.5% and 1.8% higher on H-mean than that of DRRG on SCUT-CTW1500 and Total-Text datasets, respectively. Some qualitative results on arbitrarily-shaped benchmarks are depicted in Figure 5(a) and (b).

2) Arbitrarily-Oriented Scene Text Detection: As for arbitrarily-oriented benchmark ICDAR15, HGR-Net also outperforms previous methods. Compared with top-down methods proposed to learn rotated text representation, such as Textboxes++ [20], RRPN [19], TBTD [30] and ACE [32], HGR-Net adapts anchor generation to obtain text location roughly, and then we progressively regress text contours and components from two parallel branches. Some qualitative results on arbitrarily-oriented benchmark ICDAR15 are depicted in Figure 5(c).

D. Comprehensive Comparison With DRRG, ABC-Net, and FCE-Net

Besides standard H-mean metric, we have applied Tightness-aware Intersect-over-Union (TIoU) metric [76] to evaluate detection performance. The indicators TIoU-Recall, TIoU-Precision, and TIoU-Hmean are used to assess completeness, compactness, and tightness of detection results, respectively. Table II provides detection results of HGR-Net, DRRG [12], ABC-Net [26], and FCE-Net [28] in terms of TIoU metric. It is noted that HGR-Net achieves the best results in terms of TIoU-Recall, TIoU-Precision, and TIoU-Hmean indicators on both SCUT-CTW1500 and Total-Text datasets. Comparing the results on TIoU metric and previous H-mean metric, we find that FCE-Net shows better H-mean performance than ABC-Net in terms of H-mean on SCUT-CTW1500 dataset (85.5% versus 84.7%). On the contrary, ABC-Net has surprisingly achieved better performance in terms of TIoU-Hmean metric (64.2% versus 65.4%). This could be attributed to different text contour parameterization between Bezier curves and Fourier curves, which results in different compactness and completeness. As shown in the fourth and fifth columns of Figure 6, detection results predicted by parameterized Bezier curves are more compact and sharper. Compared with ABC-Net, the substantial superiority of HGR-Net could be attributed to complementary multi-grained textual information extraction and comprehensive representation learning. For Total-Text dataset, bottom-up method DRRG outperforms ABC-Net, and FCE-Net in terms of TIoU-Recall, but it shows limited TIoU-Precision performance. It could be inferred that the higher TIoU-Recall of DRRG is benefited from text component extraction and aggregation. And top-down methods ABC-Net and FCE-Net tend to make text detection compact by applying concise and efficient text curve parametrization. Moreover, as for hybrid HGR-Net, it ranks the first place in terms of TIoU-Recall (64.2% versus the second place DRRG 63.5%) and TIoU-Precision (72.5% versus the second place ABC-Net 68.6%) on Total-Text dataset, where the evolved graph embeddings could join forces for detection refining.

Figure 6 depicts an intuitive visualization comparison of detection results predicted by different methods in multiple situations, including occluded text, low-contrast background, and text in high diversity. At the bottom of Figure 6(a-3), for

Fig. 7. Qualitative detection results of baseline, Intra-Text SLM, Inter-Text GCM, and HGR-Net. The centroids of detected text instances and components are connected by learned graph edges. Edge thickness corresponds to scaled edge weights.

the last line with characters 'II P M M SAT', DRRG detects each individual character separately instead of a holistic text instance. However, it is rarely encountered in top-down methods. HGR-Net, ABC-Net and FCE-Net directly learn to predict holistic text instances. Figure 6(a-4) has overlapping issues and redundant detection. However, the redundant detection is generally low in HGR-Net due to partial spatial relationship captured by component-level Intra-Text SLM. In the second row of Figure 6, for the long text instance 'SCAPEGOAT' in the style of WordArt, DRRG, ABC-Net, and FCE-Net suffer from missing characters. But, HGR-Net is able to handle curved text instances with diverse text fonts and styles, thereby confirming the values of exploiting global context for detection completeness. In the third row of Figure 6, there exists low-contrast background noise similar to foreground text, which might potentially lead to unexpected detection. HGR-Net is capable of enhancing the robustness of textual representation learning by instance-level and component-level cooperation and information interaction. In Figure 6(c-2), HGR-Net obtains accurate detection results and matches the ground truth closely. The above observation is consistent with performance comparison shown in Table II in terms of TIoU metric [76].

E. Ablation Study

To verify the effectiveness of each module, several ablation experiments are conducted on arbitrarily-shaped Total-Text and SCUT-CTW1500 datasets, and quantitative results are shown in Table III.

1) With or Without Component-Level Branch (CLB): The baseline is implemented by Faster R-CNN with the adapted instance-level contour regression branch. As shown in Table III, detection result of baseline equipped with component-level branch (CLB) exceeds that of baseline alone by 0.5% (83.7% → 84.2%) and 0.6% (83.5% → 84.1%).

2) With or Without Intra-Text Spatial Layout Module (Intra-Text SLM): Using Intra-Text SLM, the results can be improved by 1.3% (83.7% → 85.0%) and 1.7% (83.5% → 85.2%), respectively. Moreover, compared to the baseline trained with CLB, it obtains higher results in terms of H-mean (84.2% → 85.0% on Total-Text and 84.1% → 85.2% on SCUT-CTW1500).

3) With or Without Inter-Text Global Contextual Module (Inter-Text GCM): We conduct experiments to study the influences of the baseline with/without Inter-Text GCM. Compared with Intra-Text SLM, linkage establishing in instance-level is equally vital for scene text detection, which guides detectors to explore global contextual information from pixel-wise visual connection. From Table III, the performance of the baseline with Inter-Text GCM is improved by 1.5% (83.7% → 85.2%) and 1.3% (83.5% → 84.8%).

4) With or Without Hierarchical Relational Reasoning Module (HRRM): Integrating two-level graph embeddings, the performance has been improved by 2.8% and 2.0%.

TABLE III

ABLATION STUDY ON SCUT-CTW1500 AND TOTAL-TEXT BENCHMARKS WITH DIFFERENT MODULES. △ IMPR. REPRESENTS THE IMPROVEMENT OF THE H-MEAN COMPARED WITH THE BASELINE. THE MODULES ARE DENOTED AS FOLLOWING: CLB COMPONENT-LEVEL BRANCH; INTRA-TEXT SLM INTRA-TEXT SPATIAL LAYOUT MODULE; INTER-TEXT GCM INTER-TEXT GLOBAL CONTEXTUAL MODULE; AND HRRM HIERARCHICAL RELATIONAL REASONING MODULE. THE BEST RESULT OF EACH COLUMN IS HIGHLIGHTED IN **BOLD**

	Modules				Total-Text		SCUT-CTW1500		
	CLB	Intra-Text SLM	Inter-Text GCM	HRRM	H-mean	△Impr.	H-mean	△Impr.	Speed (ms / img)
baseline	-	-	-	-	83.7	-	83.5	-	60
baseline+	√	-	-	-	84.2	↑ 0.5%	84.1	↑ 0.6%	62
baseline+	√	√	-	-	85.0	↑ 1.3%	85.2	↑ 1.7%	69
baseline+	√	-	√	-	85.2	↑ 1.5%	84.8	↑ 1.3%	66
baseline+	√	√	√	-	86.5	↑ 2.8%	85.5	↑ 2.0%	71
HGR-Net	√	√	√	√	**87.5**	↑ 3.8%	**86.0**	↑ 2.5%	73

TABLE IV

INFERENCE SPEED AND ACCURACY COMPARISON OF HGR-NET, ABC-NET AND DRRG ON SCUT-CTW1500 BENCHMARK

	DRRG	ABC-Net	HGR-Net(ours)
H-mean (%)	84.5	84.7	**86.0**
Speed (ms / img)	82	**59**	73

In detail, it achieves higher improvement than the baseline with Intra-Text SLM (1.3% → 2.8% and 1.7% → 2.0%) or Inter-Text GCM (1.5% → 2.8% and 1.3% → 2.0%). Ideally, we want to inject intra-text graph embeddings into feature matrix, and thus enrich feature matrix for inter-text graph learning. The injected textual attributes are expected to provide auxiliary inner-text information. However, an implicit limitation of the learned two-level graph embeddings is that they are unable to perform information interaction and relation deduction across hierarchical levels. HRRM bridges cross-feed interaction between the constructed intra-text graph and inter-text graph to achieve information aggregation and propagation. By learning complementary graph embeddings, the baseline equipped with CLB, Intra-Text SLM, Inter-Text GCM and HRRM achieves optimal performance. It obtains performance improvement by 3.8% (83.7% → 87.5% on Total-Text) and 2.5% (83.5% → 86.0% on SCUT-CTW1500).

5) Inference Speed: To evaluate the real-time detection performance, we investigate computation efficiency on different settings of HGR-Net. Hierarchical graph reasoning costs most of computation time, which includes Intra-Text SLM and Inter-Text GCM. It takes approximately 9 ms per image, accounting for 70 % the computation time. Due to dynamic graph learning for each input image, it thus inevitably consumes additional computational costs.

In Table IV, we further compare HGR-Net with ABC-Net and DRRG under the same device and experimental settings. Especially, we have removed the recognition branch of ABC-Net for a fair comparison. It is shown that HGR-Net outperforms ABC-Net and DRRG by 1.3 (84.7% versus 86.0%) and 1.5 (84.5% versus 86.0%) on SCUT-CTW1500 benchmark. However, ABC-Net shows faster inference speed than HGR-Net and DRRG. It could be attributed to that ABC-Net adopts anchor-free baseline network with simplified pipelines.

6) More Results: Figure 7 shows the qualitative comparison between HGR-Net, baseline, Intra-Text SLM, and Inter-Text GCM. Moreover, Figure 7(a-2), (b-2), and (c-2) depict the learned graph structures from Intra-Text SLM, and Figure 7(a-3), (b-3), and (c-3) depict the learned graph structures from Inter-Text GCM. In Figure 7(a-1), the long yellow text instance 'Captain Kibbs Inn' is falsely split into separate fragments. And the false detection issues are also found in Figure 7(c-1), where the bottom long text instance 'D.A.V.Sektion Leutkirch' and the bottom-right equation '1984+85' are not correctly detected by the baseline. HGR-Net could capture global contextual information from the constructed inter-text graphs, and instance-level relation aggregation might enable us to properly detect the text instances with a variety of aspect ratios, with larger receptive field. Thus, the text contours could be accurately located. In Figure 7(b-1), the baseline suffers from occlusion problems, with the background ropes occluding most text instances vertically. Due to the exploited sub-text components, HGR-Net is sensitive to text locality in spatial layout. Therefore, these clues with geometric information would help stack partial fragments into the holistic text instances based on component-level intra-text information propagation. Hence, it is obvious to see that the detected text components are relatedly connected by learned graph edges, as shown in Figure 7(b-2). At the bottom of Figure 7(a-1), two redundant candidates are overlapped with the blue text instance 'grul spirits'. After hierarchical relational reasoning, the overlap issue is alleviated by learning advanced graph embeddings and the performance of HGR-Net is thus improved by linkage likelihood deduction.

F. Limitation and Discussion

Our method is one of the very first attempts to explore multi-grained text representation learning and hierarchical linkage likelihood deduction across levels. It still has several

limitations. First of all, due to the notable characteristics of text in the wild, such as inevitable variations in shape, font, orientation, and aspect ratio, a fixed number of text components within text instances is usually not practical. In cases where a text instance consists of small and short words, setting fixed number of components is likely to result in reduced homogeneity, which could deteriorate detection performance. Second, due to the vanishing gradient problem, GCN-based methods are usually limited to shallow models [78]. Although graph learning modules show promising results, as shown in Table III, the combination of deep graph architecture is an interesting avenue for future work.

V. Conclusion

We have presented HGR-Net, a novel end-to-end trainable framework for arbitrary shape scene text detection. Different from previous scene text detectors, which focus on top-down or bottom-up methods, HGR-Net exploits instance-level as well as component-level textual attributes from a hybrid perspective. To exploit multi-grained textual information, we first generate instance-level and component-level text candidates. Meanwhile, inter-text and intra-text graph construction is utilized to establish two-level linkages, and graph reasoning achieves cross-feed interaction with the aid of hierarchical design. The overall detection performance has been promoted, and the effects of occluded, redundant and ambiguous issues can be restrained by learning multi-grained textual attributes and complementary graph embeddings. Extensive experimental results demonstrate the superiority and efficiency of HGR-Net on both arbitrary shape and arbitrary orientation benchmarks.

References

[1] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao, "Detecting text in natural image with connectionist text proposal network," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 56–72.

[2] S. Long, J. Ruan, W. Zhang, X. He, W. Wu, and C. Yao, "TextSnake: A flexible representation for detecting text of arbitrary shapes," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 20–36.

[3] D. Deng, H. Liu, X. Li, and D. Cai, "PixelLink: Detecting scene text via instance segmentation," in *Proc. AAAI Conf. Artif. Intell.*, 2018, vol. 32, no. 1, pp. 1–10.

[4] 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 Trans. Image Process.*, vol. 28, no. 11, pp. 5566–5579, Nov. 2019.

[5] W. Feng, W. He, F. Yin, X. Zhang, and C. Liu, "TextDragon: An end-to-end framework for arbitrary shaped text spotting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9075–9084.

[6] 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 Recognit.*, vol. 96, Dec. 2019, Art. no. 106954.

[7] Y. Baek, B. Lee, D. Han, S. Yun, and H. Lee, "Character region awareness for text detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9357–9366.

[8] W. Wang et al., "Efficient and accurate arbitrary-shaped text detection with pixel aggregation network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8439–8448.

[9] Z. Tian et al., "Learning shape-aware embedding for scene text detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4229–4238.

[10] M. Liao, Z. Wan, C. Yao, K. Chen, and X. Bai, "Real-time scene text detection with differentiable binarization," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 7, pp. 11474–11481.

[11] Y. Zhou, H. Xie, S. Fang, Y. Li, and Y. Zhang, "CRNet: A center-aware representation for detecting text of arbitrary shapes," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 2571–2580.

[12] S. Zhang et al., "Deep relational reasoning graph network for arbitrary shape text detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9696–9705.

[13] T. Sheng, J. Chen, and Z. Lian, "CentripetalText: An efficient text instance representation for scene text detection," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 335–346.

[14] Y. Wang, H. Xie, M. Xing, J. Wang, S. Zhu, and Y. Zhang, "Detecting tampered scene text in the wild," in *Proc. 17th Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2022, pp. 215–232.

[15] C. Xue, J. Huang, W. Zhang, S. Lu, C. Wang, and S. Bai, "Contextual text block detection towards scene text understanding," in *Proc. 17th Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2022, pp. 374–391.

[16] X. Zhou et al., "EAST: An efficient and accurate scene text detector," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2642–2651.

[17] W. He, X. Zhang, F. Yin, and C. Liu, "Multi-oriented and multi-lingual scene text detection with direct regression," *IEEE Trans. Image Process.*, vol. 27, no. 11, pp. 5406–5419, Nov. 2018.

[18] Y. Liu, L. Jin, and C. Fang, "Arbitrarily shaped scene text detection with a mask tightness text detector," *IEEE Trans. Image Process.*, vol. 29, pp. 2918–2930, 2020.

[19] J. Ma et al., "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Trans. Multimedia*, vol. 20, no. 11, pp. 3111–3122, Nov. 2018.

[20] M. Liao, B. Shi, and X. Bai, "TextBoxes++: A single-shot oriented scene text detector," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3676–3690, Aug. 2018.

[21] 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 *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 67–83.

[22] M. Liao, Z. Zhu, B. Shi, G. Xia, and X. Bai, "Rotation-sensitive regression for oriented scene text detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5909–5918.

[23] P. Dai, H. Zhang, and X. Cao, "Deep multi-scale context aware feature aggregation for curved scene text detection," *IEEE Trans. Multimedia*, vol. 22, no. 8, pp. 1969–1984, Aug. 2020.

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

[25] J. Hou et al., "HAM: Hidden anchor mechanism for scene text detection," *IEEE Trans. Image Process.*, vol. 29, pp. 7904–7916, 2020.

[26] Y. Liu et al., "ABCNet v2: Adaptive Bezier-curve network for real-time end-to-end text spotting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 8048–8064, Nov. 2022.

[27] S. Zhang, X. Zhu, C. Yang, H. Wang, and X. Yin, "Adaptive boundary proposal network for arbitrary shape text detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 1285–1294.

[28] Y. Zhu, J. Chen, L. Liang, Z. Kuang, L. Jin, and W. Zhang, "Fourier contour embedding for arbitrary-shaped text detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 3122–3130.

[29] P. Dai, S. Zhang, H. Zhang, and X. Cao, "Progressive contour regression for arbitrary-shape scene text detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 7389–7398.

[30] Z. Raisi, M. A. Naiel, G. Younes, S. Wardell, and J. S. Zelek, "Transformer-based text detection in the wild," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 3156–3165.

[31] 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 Trans. Multimedia*, vol. 24, pp. 1883–1895, 2022.

[32] P. Dai, S. Yao, Z. Li, S. Zhang, and X. Cao, "ACE: Anchor-free corner evolution for real-time arbitrarily-oriented object detection," *IEEE Trans. Image Process.*, vol. 31, pp. 4076–4089, 2022.

[33] X. Zhang, Y. Su, S. Tripathi, and Z. Tu, "Text spotting transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 9509–9518.

[34] M. Huang et al., "SwinTextSpotter: Scene text spotting via better synergy between text detection and text recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 4583–4593.

[35] Z. Zhou, X. Du, Y. Zheng, and C. Jin, "Aggregated text transformer for scene text detection," 2022, *arXiv:2211.13984*.

[36] S. Long, S. Qin, D. Panteleev, A. Bissacco, Y. Fujii, and M. Raptis, "Towards end-to-end unified scene text detection and layout analysis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 1039–1049.

[37] J. Tang et al., "Few could be better than all: Feature sampling and grouping for scene text detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 4553–4562.

[38] S. Long, X. He, and C. Yao, "Scene text detection and recognition: The deep learning era," *Int. J. Comput. Vis.*, vol. 129, no. 1, pp. 161–184, Jan. 2021.

[39] Q. Ye and D. Doermann, "Text detection and recognition in imagery: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 7, pp. 1480–1500, Jul. 2015.

[40] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2963–2970.

[41] C. Yi and Y. Tian, "Text string detection from natural scenes by structure-based partition and grouping," *IEEE Trans. Image Process.*, vol. 20, no. 9, pp. 2594–2605, Sep. 2011.

[42] X. Yin, X. Yin, K. Huang, and H. Hao, "Robust text detection in natural scene images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 970–983, May 2014.

[43] W. Huang, Z. Lin, J. Yang, and J. Wang, "Text localization in natural images using stroke feature transform and text covariance descriptors," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 1241–1248.

[44] K. Wang and S. Belongie, "Word spotting in the wild," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2010, pp. 591–604.

[45] J. Lee, P. Lee, S. Lee, A. Yuille, and C. Koch, "AdaBoost for text detection in natural scene," in *Proc. Int. Conf. Document Anal. Recognit.*, Sep. 2011, pp. 429–434.

[46] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1457–1464.

[47] M. Jaderberg, A. Vedaldi, and A. Zisserman, "Deep features for text spotting," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 512–528.

[48] S. M. Lucas et al., "ICDAR 2003 robust reading competitions: Entries, results, and future directions," *Int. J. Document Anal. Recognit.*, vol. 7, nos. 2–3, pp. 105–122, Jul. 2005.

[49] A. Shahab, F. Shafait, and A. Dengel, "ICDAR 2011 robust reading competition challenge 2: Reading text in scene images," in *Proc. Int. Conf. Document Anal. Recognit.*, Sep. 2011, pp. 1491–1496.

[50] D. Karatzas et al., "ICDAR 2013 robust reading competition," in *Proc. 12th Int. Conf. Document Anal. Recognit.*, Aug. 2013, pp. 1484–1493.

[51] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, "Detecting texts of arbitrary orientations in natural images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 1083–1090.

[52] X. Yin, W. Pei, J. Zhang, and H. Hao, "Multi-orientation scene text detection with adaptive clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1930–1937, Sep. 2015.

[53] C.-K. Ch'ng, C. S. Chan, and C.-L. Liu, "Total-text: Toward orientation robustness in scene text detection," *Int. J. Document Anal. Recognit.*, vol. 23, no. 1, pp. 31–52, Mar. 2020.

[54] W. Liu et al., "SSD: Single shot MultiBox detector," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 21–37.

[55] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–6.

[56] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[57] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. 16th Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 213–229.

[58] H. Wang, Y. Zhu, H. Adam, A. Yuille, and L. Chen, "MaX-DeepLab: End-to-end panoptic segmentation with mask transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 5459–5470.

[59] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[60] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.

[61] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 1–12.

[62] H. Bi et al., "SRRV: A novel document object detector based on spatial-related relation and vision," *IEEE Trans. Multimedia*, early access, Apr. 7, 2022, doi: 10.1109/TMM.2022.3165717.

[63] X. Chen, L. Li, L. Fei-Fei, and A. Gupta, "Iterative visual reasoning beyond convolutions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7239–7248.

[64] Z. Ying, J. You, C. Morris, X. Ren, W. Hamilton, and J. Leskovec, "Hierarchical graph representation learning with differentiable pooling," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 1–15.

[65] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[66] D. Karatzas et al., "ICDAR 2015 competition on robust reading," in *Proc. 13th Int. Conf. Document Anal. Recognit. (ICDAR)*, Aug. 2015, pp. 1156–1160.

[67] B. Shi, X. Bai, and S. Belongie, "Detecting oriented text in natural images by linking segments," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3482–3490.

[68] Y. Liu and L. Jin, "Deep matching prior network: Toward tighter multi-oriented text detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3454–3461.

[69] X. Wang, Y. Jiang, Z. Luo, C. Liu, H. Choi, and S. Kim, "Arbitrary shape scene text detection with adaptive text region representation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6442–6451.

[70] C. Xue, S. Lu, and W. Zhang, "MSR: Multi-scale shape regression for scene text detection," 2019, *arXiv:1901.02596*.

[71] W. Wang et al., "Shape robust text detection with progressive scale expansion network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9328–9337.

[72] C. Zhang et al., "Look more than once: An accurate detector for text of arbitrary shapes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10544–10553.

[73] Y. Wang, H. Xie, Z. Zha, M. Xing, Z. Fu, and Y. Zhang, "ContourNet: Taking a further step toward accurate arbitrary-shaped scene text detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11750–11759.

[74] M. Liao, Z. Zou, Z. Wan, C. Yao, and X. Bai, "Real-time scene text detection with differentiable binarization and adaptive scale fusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 919–931, Jan. 2023.

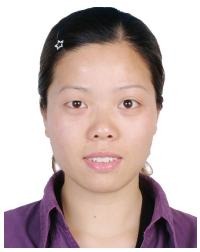
[75] H. Wang et al., "All you need is boundary: Toward arbitrary-shaped text spotting," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 7, 2020, pp. 12160–12167.

[76] Y. Liu, L. Jin, Z. Xie, C. Luo, S. Zhang, and L. Xie, "Tightness-aware evaluation protocol for scene text detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9604–9612.

[77] N. Nayef et al., "ICDAR2017 robust reading challenge on multi-lingual scene text detection and script identification—RRC-MLT," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, vol. 1, Nov. 2017, pp. 1454–1459.

[78] G. Li, M. Müller, A. Thabet, and B. Ghanem, "DeepGCNs: Can GCNs go as deep as CNNs?" in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9266–9275.

Hengyue Bi is currently pursuing the M.S. degree with the School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao, China. His research interests include computer vision and machine learning.



Canhui Xu received the Ph.D. degree from Central South University, China, in 2011. She was a Postdoctoral Research Fellow at Peking University from 2012 to 2014. She was a Visiting Scholar at Arizona State University, USA, from 2019 to 2020, and a Visiting Ph.D. Student at Imperial Collage London, U.K., from 2009 to 2010. She is currently with the School of Information Science and Technology, Qingdao University of Science and Technology, China. Her research interests include deep learning, document layout analysis, and image understanding.

Cao Shi received Ph.D. degree from Central South University in 2011. He was a Postdoctoral Research Fellow at Peking University from 2011 to 2013. He is currently with the School of Information Science and Technology, Qingdao University of Science and Technology, China. His research interests include image, video processing, and artificial intelligence.

Guozhu Liu received the Ph.D. degree in computer science and technology from the Qingdao University of Science and Technology, Qingdao, China. He is currently a Professor of computer science with the School of Information Science and Technology, Qingdao University of Science and Technology. His current research interests include deep learning, recommender systems, and intelligent software analysis.

Honghong Zhang is currently pursuing the M.S. degree with the School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao, China. Her research interests include artificial intelligence, computer vision, and image processing.

Yuteng Li is currently pursuing the M.S. degree with the School of Information Science and Technology, Qingdao University of Science and Technology, Qingdao, China. His research interests include deep learning, computer vision, and image processing.

Junyu Dong (Member, IEEE) received the B.Sc. and M.Sc. degrees from the Department of Applied Mathematics, Ocean University of China, Qingdao, China, in 1993 and 1999, respectively, and the Ph.D. degree in image processing from the Department of Computer Science, Heriot-Watt University, U.K., in 2003. In 2004, he joined the Ocean University of China, where he is currently a Professor and the Dean of Faculty of Information Science and Engineering. His research interests include machine learning, big data, computer vision, and underwater vision.