IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received February 5, 2021, accepted February 20, 2021, date of publication March 1, 2021, date of current version March 9, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3063030

Quadbox: Quadrilateral Bounding Box Based Scene Text Detection Using Vector Regression

PRATEEK KESERWANI[1], ANKIT DHANKHAR[1], RAJKUMAR SAINI[2], AND PARTHA PRATIM ROY[1], (Member, IEEE)

[1]Department of Computer Science and Engineering, IIT Roorkee, Roorkee 247667, India
[2]Department of Computer Science, Electrical and Space Engineering, Luleå tekniska universitet, 971 87 Luleå, Sweden

Corresponding authors: Prateek Keserwani (pkeserwani@cs.iitr.ac.in) and Rajkumar Saini (rajkumar.saini@ltu.se)

This work was supported by the Vinnova Nationellt Rymddatalabb (Swedish Space Data Lab) Project. The work of Prateek Keserwani was supported by the Visvesvaraya Ph.D. Fellowship.

ABSTRACT Scene text appears with a wide range of sizes and arbitrary orientations. For detecting such text in the scene image, the quadrilateral bounding boxes provide a much tight bounding box compared to the rotated rectangle. In this work, a vector regression method has been proposed for text detection in the wild to generate a quadrilateral bounding box. The bounding box prediction using direct regression requires predicting the vectors from each position inside the quadrilateral. It needs to predict four-vectors, and each varies drastically in its length and orientation. It makes the vector prediction a difficult problem. To overcome this, we have proposed a centroid-centric vector regression by utilizing the geometry of quadrilateral. In this work, we have added the philosophy of indirect regression to direct regression by shifting all points within the quadrilateral to the centroid and afterward performed vector regression from shifted points. The experimental results show the improvement of the quadrilateral approach over the existing direct regression approach. The proposed method shows good performance on many existing public datasets. The proposed method also demonstrates good results on the unseen dataset without getting trained on it, which validates the approach's generalization ability.

INDEX TERMS Scene text detection, direct regression, indirect regression, quadrilateral bounding boxes, centroid of the quadrilateral.

I. INTRODUCTION

Among all the remarkable inventions of the human being, the text is the most influential. It helps to preserve, spread, and communicate information, idea, and fact across time. The documentation of the fact and the information helps to transfer knowledge from generation to generation reliably. However, the influence of the text is not only confined to the books and the documentation. In the modern world, we are surrounded by text in the form of a vehicle's number plate, house number, location information written on signboards. The written text in the surrounding communicates the high-level semantics, which becomes helpful in understanding the world. The text which surrounds us in the real world is known as scene text. Scene text appears on the captured images/videos. The automatic detection [1], [2] and recognition [3], [4] of the appeared text leads to the automation of a wide range of applications such as image search [5], [6], instant translation [7], [8], robot navigation [9]–[12], and industrial automation [13], [14]. Hence, it is an important computer vision problem [1], [15]. Although several solutions have been proposed for this important problem in the last decade, it is still not fully solved and encounters many challenges.

At first glance, the text detection seems to be similar to object detection. However, in many aspects, text detection appears to be different from object detection. First, a significant difference is that an object is an entity with a set of attributes, such as a human is an entity with hands, legs, face, and chest. In contrast, text could be either a single character or a group of characters. In the case of text detection, a single word is assumed to be a valid detection. Even detecting a substring, which could be a valid lexical word, may be wrong as per the evaluation criterion. To illustrate this, consider a word, 'airspace' to be detected. Here, the sub-string 'space' is also a valid word in the dictionary. During the experiment, if the

The associate editor coordinating the review of this manuscript and approving it for publication was Li Minn Ang.

text detector only identifies 'space,' it will be considered a false positive, which reduces the performance. These primary differences make text detection a challenging problem.

Some significant challenges of scene text detection are the vast range of aspect ratio of the text in an image. The word can be of various lengths ranging from a single character to tens of characters. The arbitrary orientations of the text make this problem more challenging. The text can appear with complex backgrounds. It occurs due to text present on a glass window, a pole, number plate of vehicles, signboard, banner, poster, and air-balloon. The uneven illumination, a reflection of text from the floor, blur effect due to the camera's movement while capturing an image distorts the text's appearance on an image. The presence of text in an image in various scales makes the problem more difficult.

Other than the challenges mentioned above, one important issue is the bounding box representation. For object detection, a rectangular bounding box is suitable. However, in the case of text detection, the rectangular bounding box may contain a considerable amount of background pixels in addition to the text pixels. Suppose a word consist of a significant size of the first character and other smaller size characters, in some other cases few characters have a long tail. Then a rectangle is not a good representation [16]. Moreover, a rotated text may make this situation more complicated. A rectangular representation in such cases may contain the parts of characters from the nearby word. Hence, a quadrilateral bounding box [17] is a more appropriate representation for text detection. Fig. 1 demonstrates all aforementioned scenarios.

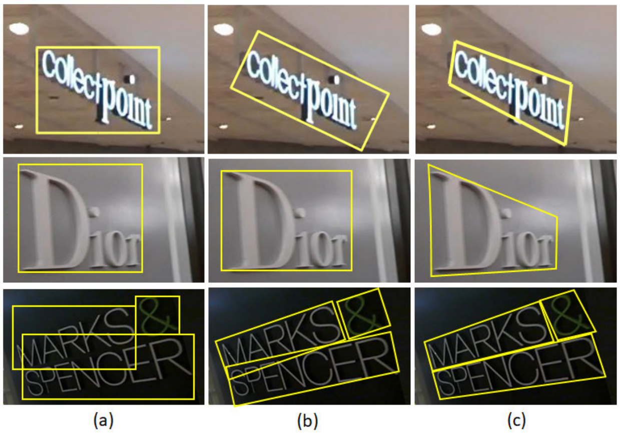


FIGURE 1. Various bounding box representations for text detection. (a) Rectangle (b) Rotated Rectangle (c) Quadrilateral. First row: The quadrilateral bounding box covers the least number of background pixels compared to the rectangular and rotated rectangular bounding box. Second row: The first character is bigger than the rest of the characters. Third row: Less overlap of the bounding box on the nearby text in quadrilateral bounding box representation.

Most of the existing text detection methods [18]–[21] produce a word-level rectangular/rotated rectangular bounding box. Only a few works had attempted and proposed text detection methods using regression-based quadrilateral bounding boxes [16], [22]. There are mainly two approaches for quadrilateral bounding box regression: (i) indirect regression (ii) direct regression. In indirect regression, the bounding box is predicted from the prior bounding boxes (anchor boxes) by learning the offset, i.e., the method estimates the distance of the desired quadrilateral bounding box from the proposal. Whereas, in the direct regression, a quadrilateral is directly predicted from a given point. The first approach has a drawback, that is, rectangular anchor boxes. In multi-oriented bounding boxes, most anchor boxes are not sufficient to capture the complete ground truth. The methods based on indirect regression and anchor boxes use a huge range of anchor boxes with various aspect-ratios and sizes to achieve good performance. In the second approach, a quadrilateral has been predicted from each point directly for the bounding boxes. Due to this, direct regression predicts irregular quadrilateral for multi-oriented scene text. However, from each point, the distance of four vertices is non-uniform and skewed.

In this work, we combined the advantages of the direct and indirect regression for the quadrilateral bounding box regression. For this purpose, we have utilized the fundamental geometrical property of the quadrilateral and proposed centroid-centric vector regression approach. Since each quadrilateral has a unique centroid, we propose that the quadrilateral bounding box be estimated from the centroid. From the indirect regression, the two-level estimation (i.e., estimation of centroid and then the prediction of vertices of the quadrilateral) has been derived, whereas, from the direct regression concept, the direct computation of the vertices from a point has been adopted. Thus, for the quadrilateral bounding boxes, the four vectors are to be learned from the quadrilateral centroid for the text bounding box. Hence, the learned bounding boxes always follow the geometry of the quadrilateral. Fig. 2 shows the difference between indirect, direct, and proposed (centroid-centric vector regression) regression for bounding box prediction.

The significant contribution of this work is four-folded:

1) We propose a centroid-centric vector regression approach that combines the philosophy of direct and indirect regression for the quadrilateral bounding box prediction.
2) It utilizes the geometry of the quadrilateral for the estimation of the quadrilateral bounding boxes.
3) We propose a region removal multi-scale testing method to improve the detection performance.
4) We propose a Street View Text Detection (SVTD) dataset to benchmark the generalization test for quadrilateral scene text detection.

The rest of the paper is organized into five sections. Section II discusses the work related to the proposed method. In Section III, the proposed method has been described. The experimental setup and results are discussed in Section IV and V, respectively. Finally, the conclusion is drawn in Section VI.

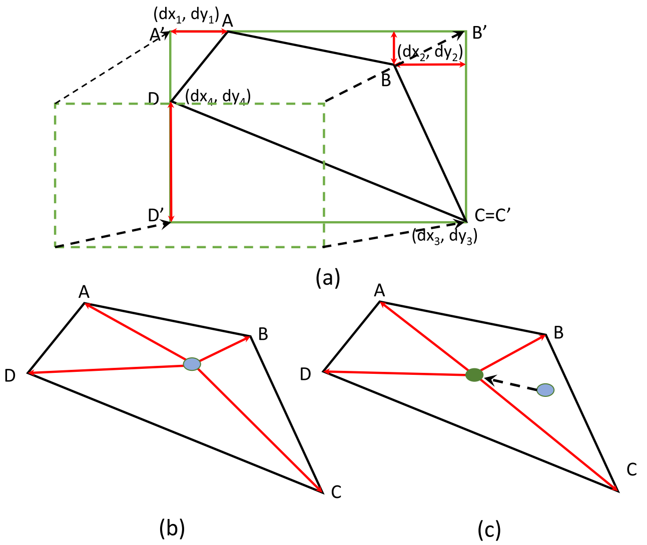


FIGURE 2. (a) Indirect Regression: The traditional approach used by various existing methods for quadrilateral bounding box prediction. First shift the anchor boxes (shown by dotted green color) and then learn the offset from vertices of the updated anchor box by (dx_1, dy_1, dx_2, dy_2, dx_3, dy_3, dx_4, dy_4). (b) Direct Regression: Prediction of the bounding box from any point within the bounding box. (c) Proposed approach: centroid-centric vector regression. The representation used in our method considers predicting the quadrilateral from the centroid (green color point) of the quadrilateral through vector regression. The other points (one such point is shown by blue color point) inside the quadrilateral are shifted towards the centroid of the quadrilateral.

II. RELATED WORK

Recently, text detection has gained massive attention from the deep learning community, and several methods have been proposed for text detection. Broadly, the text detection methods can be categorized into two approaches: (i) regression-based approach, (ii) segmentation-based approach.

A. REGRESSION-BASED APPROACH

The regression-based text detection approach are mostly inspired from two kinds of network in the object detection domain. These are two-stage and single-stage network. Two-stage network is based on the Faster-RCNN [23], whereas the single-stage network is based on YOLO [24] and SSD [25]. In a two-stage network, a region proposal network generates the candidate boxes in the first stage, and the candidate box is further refined and classified. In two-stage network, a region proposal network generate the candidate boxes in first stage and the candidate box is further refine as well as assigned a score. However, a single-stage network directly predicts boxes without the help of a region proposal network. However, on the basis of the representation a regression-based approach can be divided into the rotated rectangle representation, and quadrilateral bounding box representation.

Most of the existing regression-based approaches come under the umbrella of the rotated rectangle regression. In [26], a single-stage text detector network with text region attention has been proposed. It directly predicts word-level bounding boxes. A self-learned attention mechanism identifies the textual regions. The text region attention suppresses non-text regions in the convolution feature maps for accurate text detection. They also developed a hierarchical inception module to efficiently aggregate multi-scale inception features. In [19], the authors proposed a single-stage network based method for the horizontal rectangular bounding boxes. They have used the concept of anchor boxes with a prior aspect ratio, and they set this aspect ratio biased for text detection. In [27], the authors proposed a method that detects text through the text bounding box's corner point localization and segmenting text region in relative position. During inference, rotated bounding boxes are generated by sampling and grouping corner points. The boxes are scored with the help of segmentation. The non-maximal suppression is performed as post-processing. In [20], the authors proposed a two-stage network based method to handle text detection present with arbitrary orientation based on the region proposal networks. They introduced the rotated region-of-interest (ROI) pooling layer to pool out the rotated regions from convolution's feature maps.

For quadrilateral bounding box predictions, there are few existing approaches [16], [22], [28]–[32]. In [28], the authors proposed a single-stage network based method which directly predicts the text instance with both rotated rectangle and quadrilateral shaped bounding box. In [29], a direct regression-based text detection has been introduced, relying on single-stage architecture. It advocates the prediction of quadrilateral vertices directly from each point in the image. In [16], Liu *et al.* proposed DMPNet. It has used the quadrilateral sliding window over the convolutional features to improve the recall of text detection. The Monte-Carlo method has been used to compute the intersection over union (IoU) area of the predicted quadrilateral with ground truth in GPU during training. In [22], Liao *et al.* extends the previous work on rectangular text detection [19] for quadrilateral bounding box prediction. In contrast to the previous work, the square kernel has been used instead of a rectangular kernel. In [30], a rotation-sensitive regression-based text detector has been proposed. It performs segmentation and regression by two different branches of a different design. In [31], the authors proposed a two-stage detection pipeline for quadrilateral text detection. The quadrilateral region proposal is the first stage, followed by ROI pooling for further refinement of proposals. Similarly, in [32], the authors introduced a text-specific inception module. It is a two-stage network based method. They decompose $n \times n$ block in google architecture with $1 \times n$ and $n \times 1$ convolution. They also added deformable convolution at the end of each block. They have introduced a deformable position-sensitive ROI pooling layer to handle arbitrary orientated scene text.

B. SEGMENTATION-BASED APPROACH

Some of the current methods are based on the semantic segmentation of the image between text and non-text regions. In [33], the authors proposed a text detection method that produces pixel-wise prediction maps, followed by bounding box generation. Three types of information have been estimated: text region, individual characters, and their relationship. With proposed properties, the model can detect horizontal, arbitrary oriented, and arbitrary shaped text. In [34], the authors proposed an end-to-end learnable method to perform text detection and recognition using semantic segmentation. In [35], author consider the problem as instance segmentation approach. The text is first segmented by pixel connectivity within a text instance. The boundary is further decided with the help of segmentation. In [36] they consider the text detection problem as an instance segmentation approach. To differentiate between different text instances, they mapped the pixels to an embedding space such that pixels belonging to the same instance are clustered together and vice-versa. To make their model shape agnostic, they introduced shape aware loss for training their model. In [37] predict text by character region score and affinity score. In [38], the authors proposed a scale-robust arbitrary oriented text-detection deep learning architecture. They introduced a feature refining block for multi-scale context features fusion. It is useful for text detection in a feature map of higher resolution. In [39] a differentiable binarization technique has been proposed for arbitrary shaped text detection.

From the literature review, we have found a few works under the regression category for quadrilateral bounding box prediction. However, our proposed approach has some similarity with existing approaches, but it also differs in many points. As similar to [16], [22], [28], [29], our approach is also a single-stage text detection approach. In contrast, the [28], [29] are direct regression-based approaches, and [16], [22], [30] are indirect regression-based approaches. Whereas our proposed method has combined the philosophy of both direct and indirect regression-based approaches. The [31], [32] has employed the ROI pooling layer to pool out the feature and further adjust the quadrilateral. In contrast, our approach is fully convolutional and does not need the expensive ROI pooling layer. From this analysis, we have found that the proposed method possesses the following superiority with other similar approaches (i) anchor box free approach (ii) no need of expensive ROI pooling (iii) no complex inference heuristic for box prediction (iv) required only a vanilla non-maximal suppression. All these points make this approach simple yet effective for scene text detection.

III. PROPOSED METHOD

The proposed method is based on utilizing the geometry of quadrilateral for bounding box regression. This method is based on a deep architecture having three building blocks: feature up-sampling, pyramid feature fusion, and bounding box prediction. The bounding box prediction is made by proposed centroid-centric vector regression approach which combines the philosophy of direct and indirect regression. The proposed method only relies on standard non-maximal suppression as a post-processing step. For boosting the performance, a region removal multi-testing approach has also been proposed. The used approach and the method are explained in detail in the following subsections.

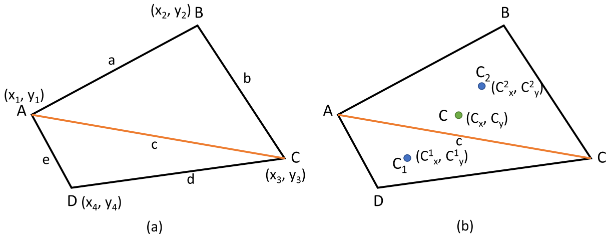


FIGURE 3. **Steps to compute the centroid (a) The quadrilateral *ABCD* is divided into two halves i.e. two triangles (b) The centroid of two triangles (blue color circle) C_1 and C_2 are computed and then the centroid of quadrilateral (green color circle) is computed from these triangle's centroid.**

A. OUR APPROACH

Suppose the input image is of dimension $M \times N$. The proposed architecture first divides the input image into the $P \times Q$ grids, where each grid is of dimension $\Delta w \times \Delta h$. From each grid location, the four vertices of the quadrilateral are predicted. Suppose, the quadrilateral consists of four vertices $A(x_1, y_1)$, $B(x_2, y_2)$, $C(x_3, y_3)$ and $D(x_4, y_4)$. The length of the four sides of the quadrilateral are a, b, d, and e. The quadrilateral is split into two triangles. The centroid of the triangles are computed separately, which further computed the centroid of the quadrilateral [40] (illustrated in Fig.3). The centroid of the quadrilateral is computed as:

$$(C_x, C_y) = \left(\frac{\sum_{i=1}^{2} C_x^i A_i}{\sum_{i=1}^{2} A_i}, \frac{\sum_{i=1}^{2} C_y^i A_i}{\sum_{i=1}^{2} A_i} \right) \tag{1}$$

where

$$(C_x^1, C_y^1) = \left(\frac{x_1 + x_2 + x_3}{3}, \frac{y_1 + y_2 + y_3}{3} \right) \tag{2}$$

$$(C_x^2, C_y^2) = \left(\frac{x_1 + x_3 + x_4}{3}, \frac{y_1 + y_3 + y_4}{3} \right) \tag{3}$$

$$A_1 = \sqrt{S_1(S_1 - a)(S_1 - b)(S_1 - c)} \tag{4}$$

$$A_2 = \sqrt{S_2(S_2 - c)(S_2 - d)(S_2 - e)} \tag{5}$$

$$S_1 = \frac{a + b + c}{2} \tag{6}$$

$$S_2 = \frac{c + d + e}{2} \tag{7}$$

The (C_x^1, C_y^1) and (C_x^2, C_y^2) are centroid of the triangle $\triangle ABC$ and $\triangle ACD$ respectively. A_1 and A_2 are area of triangle $\triangle ABC$ and $\triangle ACD$ respectively. The above-computed centroid is selected as a reference point to predict the four quadrilateral vertices. Since each quadrilateral has a well-defined centroid, this geometrical property of a quadrilateral is used in our proposed work for predicting the quadrilateral bounding box. The boxes are to be predicted from the centroid of the quadrilateral by using direct regression. There are n points in a quadrilateral, and offset from all n points to the centroid of the ground truth bounding box is learned.

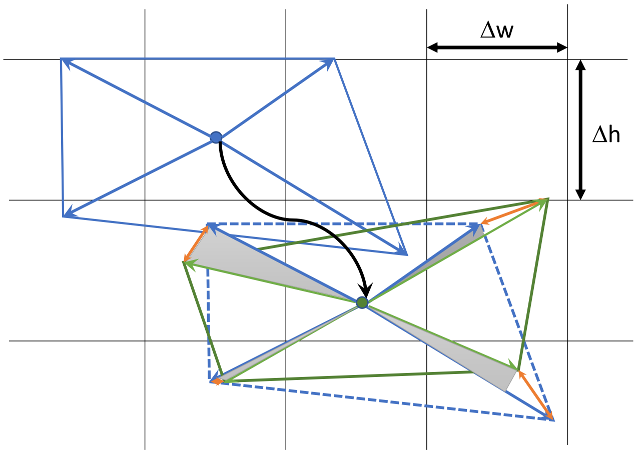


FIGURE 4. The approach for centroid-centric vector regression: The green and blue colors show the ground truth and the predicted bounding boxes, respectively. The learned offset is to move the predicted bounding box to the centroid of the ground truth bounding box is shown by the black-colored arrow. Orange-colored arrows indicate the difference between the vertices of the predicted and ground truth bounding box. Gray areas show the angle difference between the predicted vector and the ground truth vectors. These differences can be learned with the help of loss functions proposed in Section III-C.

So, each predicted bounding box always follows a balanced quadrilateral structure, and the approach is the combination of direct and indirect regression. The same is illustrated in Fig. 4.

B. NETWORK ARCHITECTURE

For the regression of the quadrilateral bounding box, a fully convolutional architecture has been used. The proposed architecture is an encoder-decoder based model and fully convolutional. The proposed architecture consists of three blocks, namely: (a) Feature Up-sampling, (b) Feature pyramid block, and (c) Bounding box prediction. The proposed architecture is built over the backbone of pre-trained VGGNet. The kernel configurations are shown in Fig. 5.

Feature Up-sampling: In a standard detection pipeline such as [23], [24], the feature maps are reduced by applying successive convolutions and pooling operations. Due to this, the feature maps spatial size reduces and divides the image into grids. If the grid location contains more than one text instance, multiple regression bounding boxes are computed for each grid location. However, the regression of multiple bounding boxes makes the model complex. It is mentioned in [24] that such a method does not behave well for the cluttered objects. Although, in the case of text detection, the cluttered text is more likely to be present. Hence, to address this issue, our architecture is designed similar to U-Net to increase feature maps' size before prediction. It is achieved by the successive fusion of downstream features with the up-sampled features. However, the direct up-sampling without feature adaptation introduces uncertainty and ambiguity during inference [41]. To get rid of it, we do three things: (i) only considered the shrunk region of the polygon for box regression (to avoid ambiguity of text border region). (ii) the features are up-sampled and then merged with the backbone feature again. (iii) used bi-linear up-sampling in place of the nearest neighbor.

The architecture is designed specially to handle multi-oriented and skewed text detection. The text may appear with different angles, sizes, and perspective distortions in the multi-orientation scene text. Predicting a bounding box requires that the receptive field of the neural network covers the whole text instance. The simplest solution is to use deeper architecture to cover it. However, in images, some of the text instances are very small, whereas others are huge. It requires a flexible receptive field to cover text instances of various size ranges. Due to this analysis, the deformable convolutions [42] is used in our model. The conventional convolution is applied on a predefined rectangular grid. Whereas, in the case of deformable convolution, the grid location is computed by learning the offset. Thus, the deformable convolutions are capable of achieving scaling and rotation via learned grid offset. This property of deformable convolution is used in our architecture.

The VGGNet consists of five downsampling operations. Among them, four down-sampling feature maps, d_1, d_2, d_3, and d_4 have been extracted with spatial resolution of $\frac{M}{32} \times \frac{N}{32}$, $\frac{M}{16} \times \frac{N}{16}$, $\frac{M}{8} \times \frac{N}{8}$, and $\frac{M}{4} \times \frac{N}{4}$ respectively. The progressive up-sampling of feature map's consists of two entities, namely, merging (m_i) and feature up-sampling (s_i) defined by the following equations:

$$s_i = \begin{cases} U(D_{3\times3}(C_{1\times1}(m_{i-1})); 2), & i > 1 \\ U(d_i; 2), & i = 1 \end{cases} \tag{8}$$

$$m_i = \begin{cases} [d_{i+1}|s_i], & i < 3 \\ D_{3\times3}(C_{1\times1}([d_{i+1}|s_i])), & i = 3 \end{cases} \tag{9}$$

where $U(.; x)$ is the feature bi-linear up-sampling by a factor x. $D_{p\times q}$ and $C_{p\times q}$ represent the deformable and vanilla convolution operation with kernel size $p \times q$ respectively. The [xly] represent the concatenation of two features along channel dimension.

Pyramid Feature Fusion : The receptive field plays an important role in object detection. The receptive field must be big enough such that no important information is left out for the desired prediction [43]. The text appearing in the image are of different sizes. The text covers a wide range of aspect ratios and sizes. A small text instance requires a smaller receptive field, whereas the bigger text instance requires a bigger receptive field. In [44], the property of receptive field has been used to introduce a light-weight architecture for text region proposal. In [44], the architecture is trained for a fixed receptive field, and bigger text is detected by considering the down-sampled image. Also, [45]–[47] proves the impact of pyramid feature for effective object detection. Hence, inspired by this, a pyramid feature fusion block has been used to capture the text with various sizes. The feature maps of various levels have different receptive fields. Hence, the level which covers the whole text instance is better to predict the bounding box at a specific scale.

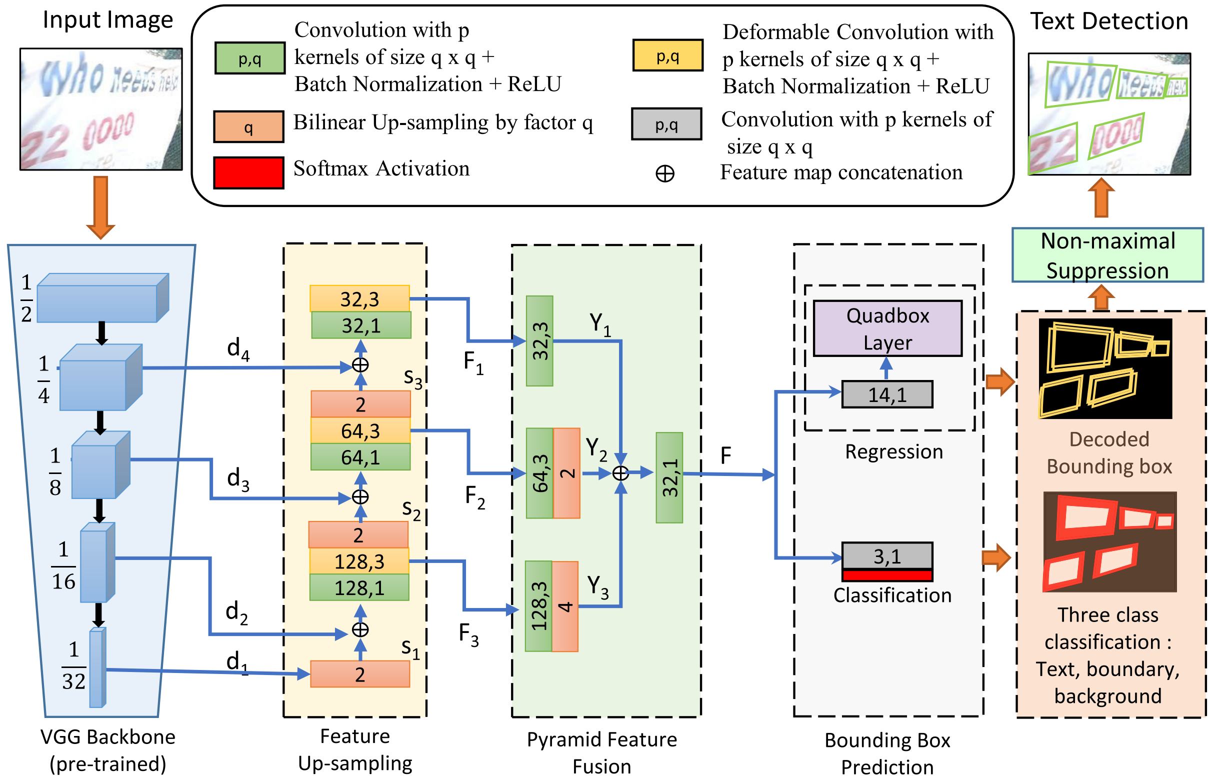

Input Image
p,q Convolution with p kernels of size q x q + Batch Normalization + ReLU
q Bilinear Up-sampling by factor q
Softmax Activation
p,q Deformable Convolution with p kernels of size q x q + Batch Normalization + ReLU
p,q Convolution with p kernels of size q x q
⊕ Feature map concatenation
Text Detection
1/2
1/4
1/8
1/16
1/32
d4
d3
d2
d1
32,3
32,1
s3
2
64,3
64,1
s2
2
128,3
128,1
s1
2
F1
F2
F3
32,3
Y1
64,3
2
Y2
128,3
4
Y3
32,1
F
Quadbox Layer
14,1
Regression
3,1
Classification
Non-maximal Suppression
Decoded Bounding box
Three class classification : Text, boundary, background
VGG Backbone (pre-trained)
Feature Up-sampling
Pyramid Feature Fusion
Bounding Box Prediction

FIGURE 5. The proposed architecture consists of three blocks: (a) Feature Up-sampling, (b) Pyramid feature fusion, and (c) Bounding box prediction. The prediction block consists of learning two tasks: (i) classification, (ii) quadrilateral bounding box regression. Finally, the non-maximal suppression is used as a post-processing.

The PFF block obtains feature from different levels of feature up-sampling block.The features are denoted by F_i. The spatial resolution of each feature map is $\frac{M}{2^{i+1}} \times \frac{N}{2^{i+1}}$. The features F_i are computed at different level of up-sampling block and represent features at different receptive field. For F_i the features are mapped up-sampled and the computed up-sampled feature map is:

$$Y_i = U(\phi(F_i); 2^{i-1}), i = \{1, 2, 3\} \tag{10}$$

where $U(.; x)$ represents the bi-linear up-sampling of the feature map with the scaling factor x, ϕ denotes the combination of convolution-normalization-ReLU operations. Finally, the features Y_i which have same spatial resolution are fused and defined by:

$$F = \varphi(W_{1\times 1} * [Y_1|Y_2|Y_3] + b) \tag{11}$$

where φ is the combination of the batch-normalization and ReLU operation, $[x_1|x_2]$ denotes the concatenation of the vectors x_1 and x_2, $*$ represent the convolution operation, b represent the bias. The $W_{1\times 1}$ represents a 1×1 kernel which reduces the channel dimension of concatenated feature from 224 to 32.

Bounding Box Prediction: The bounding box prediction has two tasks to perform, namely, regression and classification. For regression, the quadbox layer has been used for the quadrilateral bounding box. Whereas for classification, semantic segmentation has been done. The following subsections cover the details of regression and classification.

1) QUADBOX LAYER (CENTROID-CENTRIC VECTOR REGRESSION)

The objective is to predict the quadrilateral for text detection. The geometry of the quadrilateral has been utilized to directly predict the quadrilateral. Each quadrilateral has a unique centroid, and the quadbox layer shifts all candidate points (identified by segmentation) to the centroid of the quadrilateral. The vertices are directly predicted using the vector regression from the centroid of the quadrilateral. The quadrilateral can be represented by a tuple of 14 dimension $R = (a, b, \boldsymbol{V_i}|i \in \{1, 2, 3, 4\})$ and $\boldsymbol{V_i} = (r_i, \alpha_i, \beta_i)$. Here, $\boldsymbol{V_1}$, $\boldsymbol{V_2}$, $\boldsymbol{V_3}$, and $\boldsymbol{V_4}$ are four vectors from the centroid (C_x, C_y). Symbol r, α, and β are vector length, sine, and cosine angle between the vectors and the x-axis respectively. The label generation for this representation is done by Algorithm 1. For ground truth generation of quadbox, a square image is cropped with a window size of W to make batches of the same size in the training phase. The bounding box ground truth generation requires a bounding box and cropped window size. The 14-dimension encoded vector representation is generated for a bounding box.

The quadrilateral bounding box has been predicted as 14-dimension vector $(\hat{R})$ from the quadbox layer by the

Algorithm 1 Box Regression Ground Truth Generation

Input: A Bounding box *BB* consists of 4 vertices, and image crop size *W*.

Output: Vector regression map *Map*.

```
procedure Regression($BB$, $W$)
    $Map \leftarrow$ zeros($\frac{W}{4}$,$\frac{W}{4}$,14)
    $P \leftarrow$ point inside the provided bounding box
    $R \leftarrow$ zeros(14)
    $center \leftarrow CENTROID(BB)$
    $R[0] \leftarrow \frac{center[0]}{4}$
    $R[1] \leftarrow \frac{center[1]}{4}$
    for $i \leftarrow 1$ to 4 do
        $vector \leftarrow BB[i]$- $center$
        $r \leftarrow \sqrt{vector[0]^2 + vector]1]^2}$
        $R[3i] \leftarrow r$
        $R[3i+1] \leftarrow \frac{vector[1]}{r}$
        $R[3i+2] \leftarrow \frac{vector[0]}{r}$
    end for
    for $i \leftarrow |P|$ do
        $Map[P[i][0],P[i][1]] \leftarrow R$
    end for
end procedure
```

following equations:

$$\overline{C_x} = \gamma \frac{e^{2\hat{R}_1} - 1}{e^{2\hat{R}_1} + 1} + x \tag{12}$$

$$\overline{C_y} = \eta \frac{e^{2\hat{R}_2} - 1}{e^{2\hat{R}_2} + 1} + y \tag{13}$$

$$\overline{r_i} = \zeta \frac{e^{\hat{R}_{3i}}}{1 + e^{\hat{R}_{3i}}} \tag{14}$$

$$\overline{\alpha_i} = \frac{e^{2\hat{R}_{3i+1}} - 1}{e^{2\hat{R}_{3i+1}} - 1} \tag{15}$$

$$\overline{\beta_i} = \frac{e^{2\hat{R}_{3i+2}} - 1}{e^{2\hat{R}_{3i+2}} - 1} \tag{16}$$

where $\hat{R}_i$ represent the i^{th} dimensional value from prediction vector. γ, η, and ζ are the scaling factors and fixed by the value $\frac{W}{8}$, $\frac{W}{8}$, and W respectively. The scaling factor γ and η scales the predicted values in the range of $[-\frac{W}{8}, \frac{W}{8}]$ and $[-\frac{W}{8}, \frac{W}{8}]$ with respect to the point (x, y) respectively. Whereas, the scaling factor ζ scales the predicted vector to the same scale as used for a normalization during ground truth generation.

The quadrilateral could be decoded with respect to a point (x, y) by using the following equations:

$$\overline{V_i^x} = \overline{C_x} + \overline{r_i}\,\overline{\beta_i} \tag{17}$$

$$\overline{V_i^y} = \overline{C_y} + \overline{r_i}\,\overline{\alpha_i} \tag{18}$$

2) CLASSIFICATION

The second task is the classification, which has been used to score the bounding boxes. Since our primary goal is to make the bounding box at the word level, the boxes must be predicted near the center of the quadrilateral and avoid the boundary pixel. Due to this fact, the classification is posed as a three-class problem, namely, text, non-text, and boundary pixels. For generating three-class ground truth, each bounding box is shrunk such that the length of the vector from the centroid to the vertices of the quadrilateral is reduced to 0.8 times the original. Then, *CONTOUR*(*BB*, *value*) is called to draw contour with values and produce a mask. The values for shrank and original bounding box contour are set to 1 and 2, respectively. Then the shank and original contour mask are added together. The pixel with value 3, 1, and 0 are considered text, boundary, and non-text classes. Algorithm 2 gives the procedure to generate the ground truth for the classification. Sample of the text, non-text and boundary pixels for classification are shown in Fig. 6.

Algorithm 2 Classification Ground Truth Generation

Input: A Bounding box *BB* consist of 4 vertices.
Output: Segmentation map for text (*Text_class*),
boundary (*Boundary_class*),
and background (*NonText_class*).

```
procedure Classification($BB$)
    center $\leftarrow$ $CENTROID(BB)$
    shrink_coordinate $\leftarrow$ [ ]
    per $\leftarrow$ 0.8                          ▷ percentage of length
    for $i \leftarrow 1$ to 4 do
        end_point $\leftarrow$ center + per * (BB[i]- center)
        shrink_coordinate.append(end_point)
    end for
    shrink_quad $\leftarrow$ $CONTOUR$(shrink_coordinate, 1)
    original_quad $\leftarrow$ $CONTOUR$(coordinate, 2)
    sum $\leftarrow$ shrink_quad + original_quad
    Text_class $\leftarrow$ (sum = 3)
    Boundary_class $\leftarrow$ (sum = 1)
    NonText_class $\leftarrow$ (sum = 0)
end procedure
```

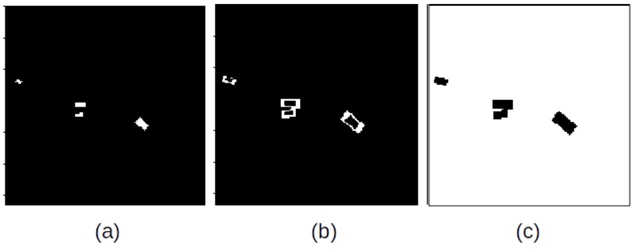


FIGURE 6. Sample for classification ground truth: (a) Text region (b) Boundary region (c) Non-text region.

C. LOSS FUNCTION

The proposed method consists of the learning of two tasks: vector regression and classification. The loss computed on four vectors from the centroid to the quadrilateral vertex is the combination of three losses, namely, centroid loss, angle loss, and length loss. The centroid loss ($\mathcal{L}_{centroid}$) is computed to shift the default position of prediction to the centroid of the ground truth quadrilateral. The angle loss ($\mathcal{L}_{angle}$) is used to compute the angle difference between the predicted vector and the ground truth vector. The length loss ($\mathcal{L}_{length}$) is used to find the length difference between the predicted vector and the ground truth vector. The losses are computed by the following equations:

$$\mathcal{L}_{centroid} = \frac{1}{|\mathbb{I}^{text}|}\sum_{i=1}^{s^2}\mathbb{I}_i^{text}[|C_{x_i}-\overline{C_{x_i}}|+|C_{y_i}-\overline{C_{y_i}}|] \tag{19}$$

$$\mathcal{L}_{angle} = \frac{1}{|\mathbb{I}^{text}|}\sum_{i=1}^{s^2}\mathbb{I}_i^{text}[(\alpha_i-\hat{\alpha_i})^2+(\beta_i-\hat{\beta_i})^2] \tag{20}$$

$$\mathcal{L}_{length} = -\frac{1}{|\mathbb{I}^{text}|}\sum_{i=1}^{s^2}\mathbb{I}_i^{text} log\left(1-\left[\frac{|r_i-\hat{r}_i|}{max(r_i,\hat{r}_i)}\right]\right) \tag{21}$$

where s represents the grids, and $\mathbb{I}_i^{text}$ is the position where the ground truth has text annotation. Length loss $\mathcal{L}_{length}$ calculated by measuring the error in absolute length is not a correct measure as that loss is averaged while updating the network's weight. This way of calculating loss suppresses the loss from small text instances in images. It deteriorates the performance as the model suffers in the detection of small text instances. The equation for regression loss ($\mathcal{L}_{reg}$) is given as follows:

$$\mathcal{L}_{reg} = \lambda_{centroid}\mathcal{L}_{centroid} + \lambda_{angle}\mathcal{L}_{angle} + \lambda_{length}\mathcal{L}_{length} \tag{22}$$

where $\lambda_{centroid}$, λ_{angle}, and λ_{length} are used as 0.2, 5.0, and 1.0 in this work. The other task of the network is to classify between text and non-text. This classification is done at the pixel level and avoids the expensive ROI pooling operation. The number of text pixels is comparatively less as compared to the background pixel. Since the method is fully convolutional, it increases the false-positive case if a wrong classification occurs. A false positive classification increases the recall but decreases the precision. Hence, a loss function is needed, which maximizes precision while maintaining the recall. Suppose p_i is the predicted probability of pixel to be text, and g_i is the probability of ground truth to be text. Then, the Tversky loss [48] for the classification between text and non-text is defined as:

$$\mathcal{L}_{class} = \frac{\sum_{i=1}^{s^2} p_i g_i}{\sum_{i=1}^{s^2} p_i g_i + \lambda \sum_{i=1}^{s^2} p_i g_i{'} + (1-\lambda)\sum_{i=1}^{s^2} g_i p_i{'}} \tag{23}$$

$$p_i{'} = 1 - p_i \tag{24}$$

$$g_i{'} = 1 - g_i \tag{25}$$

where λ parameter is used to control the trade off between false-positive and false-negative rate. In this experiment, the value of λ is taken as 0.8. The classification loss ($\mathcal{L}_{class}$) is applied on three planes separately by considering it into three different binary problems. The sum of these three dice losses has been used as a classification loss.

The total loss, $\mathcal{L}$ is the linear combination of the regression loss $\mathcal{L}_{reg}$ and classification loss $\mathcal{L}_{class}$ given by the following equation:

$$\mathcal{L} = \mathcal{L}_{reg} + \mathcal{L}_{class} \tag{26}$$

D. REGION REMOVAL MULTI-SCALE TESTING

The text appearing in the image has various aspect-ratios and sizes. From the literature, it is found that multi-scale testing is beneficial to handle this challenge. In usual multi-scale testing, the $B = \{B_1, B_2, \ldots B_n\}$ bounding boxes are predicted from n input scales $I = \{I_1, I_2, \ldots I_n\}$. A bounding box appearing at scale I_x may have overlapping bounding boxes at scale I_y. This overlapping causes an extra burden on the NMS algorithm. As a solution to the above-stated dependency, a region removal multi-scale testing (RRMT) method is introduced. The central idea is that the text region predicted on one scale with reasonable confidence will be removed from the next scale's image. The NMS is applied over the gathered bounding boxes B from n scales. The procedure of RRMT is summarized in Algorithm 3.

Algorithm 3 Region Removal Multi-Scale Testing

```
Input: $I = \{I_1, I_2, \ldots, I_n\}$ is input image at n scales.
$M$ is the Quadbox trained model.
$T$ is the segmentation threshold.
Output: Bounding Boxes ($BB$).
procedure RRMT($I, M, T$)
    $B \leftarrow \phi, S \leftarrow \phi$
    for $i \leftarrow 1$ to $n$ do
        if i $> 1$ then
            $I_i \leftarrow I_i \times CS$
        end if
        $B_i, S_i \leftarrow M(I_i)$
        $B \leftarrow B \cup B_i$
        $S \leftarrow S \cup S_i$
        $S_i^T \leftarrow S_i > T$
        if i $= 1$ then
            $CS \leftarrow (1 - S_i^T)$        ▷ cumulative seg. mask
        else
            $CS \leftarrow CS \times (1 - S_i^T)$
        end if
    end for
    $BB \leftarrow NMS(B, S)$
    return $BB$
end procedure
```

IV. EXPERIMENTS

The proposed method is evaluated over many publicly available datasets, namely, ICDAR2015, COCO-Text, ICDAR2013, and MSRA-TD500. We also introduced a challenging dataset SVTD for the generalization test of the trained model for quadrilateral bounding box predictions. Description of datasets, implementation details, and evaluation protocols are provided in the following subsections.

A. DATASETS DESCRIPTION

The following benchmark datasets have been used in this work:

ICDAR2015: This dataset [17] covers the incidental scene text and proposed in Robust Reading Competition as challenge 4. It consists of 1000 training set images for word-level text detection and 500 test set images with text localization bounding boxes. Each image may contain multiple multi-oriented text instances. This dataset is provided with the quadrilateral bounding boxes.

COCO-Text: This dataset [49] contains 63,686 images and 173,589 labelled text regions. It consists of horizontal, curved, and multi-oriented text. The label provided for this dataset is the axis-aligned rectangle. It consists of text covering a wide range of variations, and it is the biggest scene text dataset for the detection task, which covers most of the possibility of the existence of text in the real world.

ICDAR2013: ICDAR2013 dataset consists of a total of 462 images for horizontal text detection and recognition. The training set and the testing set contain 229 and 233 images, respectively. For the text localization task, the ground truth is provided in the term of word bounding boxes.

MSRA-TD500: MSRA-TD500 contains text with multi-oriented rotated rectangle annotations. It consists of English and Chinese text images as well as indoor and outdoor images. It consists of 300 training images and 200 testing images.

SVTD[1]Dataset donwload link : Street View Text Detection dataset consists of 250 images borrowed from the SVT dataset [50]. This dataset contains the images harvested from Google Street View. Image text in this dataset exhibits high variability, and some of the images are severely corrupted by noise and blur. As SVT is a word spotting dataset, only testing images were considered, and the dataset for the text detection was annotated on the same hypothesis as ICDAR2015 with quadrilateral bounding boxes.

B. IMPLEMENTATION DETAILS

Training The proposed model is trained with a batch size of 10 for 1800 epochs. The model is trained on the ICDAR2015 dataset for quadrilateral bounding box predictions. The proposed model is trained by using Adam optimizer with the initial learning rate, β_1, and β_2 are 10^{-4}, 0.9, and 0.999, respectively. The decay in the learning rate is performed by a step decay policy, i.e., decay by a fraction of 0.1 after 1000 epochs. Each training iteration takes 0.45 seconds. For other datasets, the training details are mentioned in Table 1.

TABLE 1. Fine-tuning details for various datasets. [ILR: Initial Learning Rate].

Dataset	ILR	#Epoch	Decay Interval
ICDAR2015	10^{-4}	1800	1000
ICDAR2013	10^{-4}	1700	855
MSRA-TD500	10^{-4}	1400	750

Data Augmentation In this work, the random scaling and random rotation have been done within the range of [0.8, 1.2] and [−10°, 10°], respectively. Next, a random crop of size 512 × 512 has been done. On these cropped images, the gamma correction, Gaussian blur, median blur, mean blur, light reflection effect, channel swap, Gaussian noise, and Poisson noise has been performed.

Post-processing During inference, the segmentation threshold and NMS are the only post-processing methods used in our method.

Hardware and Software The proposed method is implemented on PyTorch.[2] All the experiments have been conducted on the DGX server with Xeon E7 processor and 32GB NVIDIA Tesla V100 GPU.All the experiment has been conducted on DGX server having Xeon E7 processor, 32GB NVIDIA Tesla V100 GPU. The evaluation has been done on the batch size of 1.

C. EVALUATION PROTOCOL

For evaluating the performance of the text detection on various datasets, we have taken the precision (P), recall (R), and f-measure (FM). These are as follows:

$$P = \frac{TP}{TP + FP} \tag{27}$$

$$R = \frac{TP}{TP + FN} \tag{28}$$

$$FM = 2 \times \frac{P \times R}{P + R} \tag{29}$$

where TP, FP, and FN are the true positive, false positive, and false negative, respectively. In the case of text detection, if the detected bounding box has an intersection over union with respect to the ground truth bounding box more than the threshold, then it is considered as a true positive. Incorrect bounding box predictions are considered as a false positive, and missed bounding boxes as a false negative. The F-measure is used to measure the trade-off between precision and recall.

V. RESULTS AND ANALYSIS

For the detailed analysis and evaluation of the proposed method, the qualitative and quantitative analysis of the proposed method is performed. The ablation study is also conducted. The proposed method is also compared with the

[1] https://drive.google.com/drive/folders/1DGW-k8LxATXu06hGze1t0E3Cfhbv1kKg?usp=sharing

[2] https://pytorch.org/

TABLE 2. The performance of detection results in various settings. The best results regarding each setting is highlighted by boldfaced text.

Dataset	Channel Order	Single Scale						Multi Scale					
		NMS			LANMS			NMS			LANMS		
		P	R	FM	P	R	FM	P	R	FM	P	R	FM
ICDAR2015	RGB	0.906	0.788	0.843	0.901	0.788	0.841	0.887	0.818	**0.851**	0.881	0.819	**0.849**
	GBR	0.903	0.785	0.840	0.897	0.785	0.837	0.880	0.809	0.843	0.875	0.810	0.841
	GRB	0.905	0.790	**0.844**	0.903	0.792	**0.844**	0.884	0.817	0.849	0.880	0.820	**0.849**
	BRG	0.903	0.786	0.840	0.904	0.791	0.843	0.878	0.813	0.845	0.876	0.816	0.845
	BGR	0.901	0.785	0.839	0.897	0.784	0.837	0.881	0.814	0.846	0.870	0.809	0.839
ICDAR2013	RGB	0.898	0.714	0.795	0.904	0.717	0.800	0.844	0.780	0.811	0.833	0.784	**0.808**
	GBR	0.902	0.712	0.796	0.908	0.718	**0.802**	0.842	0.765	0.801	0.832	0.773	0.802
	GRB	0.901	0.712	0.795	0.901	0.711	0.795	0.840	0.776	0.807	0.828	0.779	0.803
	BRG	0.901	0.715	**0.797**	0.903	0.715	0.798	0.847	0.783	**0.814**	0.824	0.778	0.800
	BGR	0.902	0.715	**0.797**	0.905	0.720	**0.802**	0.842	0.778	0.809	0.830	0.774	0.801
MSRA-TD500	RGB	0.846	0.617	0.714	0.839	0.619	0.712	0.790	0.714	**0.750**	0.743	0.708	0.725
	GBR	0.842	0.609	0.707	0.837	0.611	0.706	0.787	0.698	0.740	0.752	0.705	**0.727**
	GRB	0.848	0.619	0.715	0.845	0.615	0.712	0.790	0.714	**0.750**	0.743	0.708	0.725
	BRG	0.847	0.615	0.713	0.838	0.612	0.708	0.780	0.709	0.743	0.724	0.698	0.711
	BGR	0.853	0.619	**0.717**	0.847	0.615	**0.713**	0.790	0.711	0.748	0.746	0.709	**0.727**

existing state-of-the-art (SOTA) methods in the following subsections.

A. QUANTITATIVE RESULTS

1) DETECTION PERFORMANCE

The robustness of the model is an important aspect. For ensuring the robustness of the model, it is evaluated in different settings. We have checked the performance by changing the input image's channel order using two types of post-processing techniques, i.e., NMS and locality aware non-maximal suppression (LANMS). For measuring the text detection, the metrics, namely, precision, recall, and F-measure, are used. The obtained values are summarized in Table 2.

Quadrilateral Text Detection Since the major contribution of the proposed method is quadrilateral text detection, the proposed method's performance with the ICDAR2015 is evaluated, which comes with the quadrilateral bounding box annotation. For testing on a single scale, the image is resized such that the larger size becomes 1504. The post-processing threshold for segmentation and NMS is taken as 0,99 and 0.1, respectively. Multi-scale testing is also conducted, and the chosen size for multi-scale testing is 1504 and 512. Multi-scale testing helps to improve both precision and recall. The results are summarized on Table 2 with various testing settings.

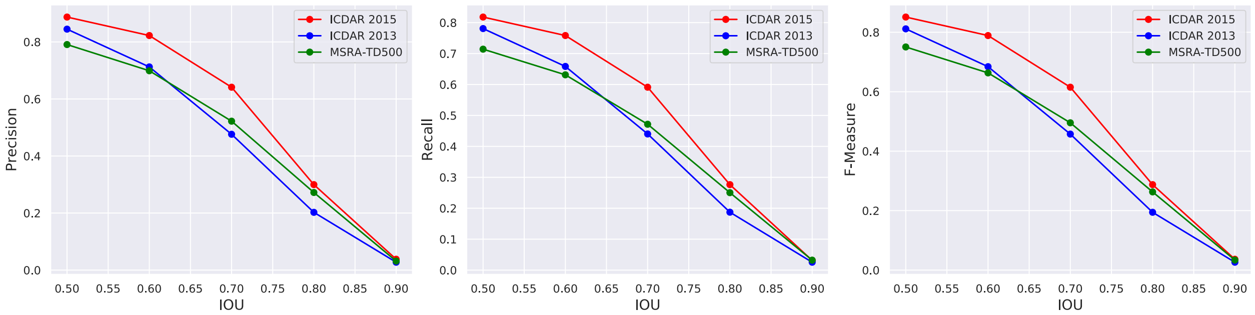
Detecting Horizontal Text For the analysis of the proposed method on the rectangular annotation text instances, the ICDAR2013 dataset is chosen. In ICDAR2013, some of the images contain wrong annotations in the training dataset. Such images with the wrong annotations were dropped during training. The model was trained by combining the images of ICDAR2013 and ICDAR2015. The trained model has been tested by resizing the image such that the largest size of the image is 640 for single scale testing. However, for the multi-scale testing, sizes of 256, 768, and 1504 are considered. The results are mentioned in Table 8. The results on ICDAR2013 are mentioned in Table 2.

Line level text detection Most of the existing datasets are for the word-level bounding box prediction. Whereas line-level text detection is also essential because it is a prime concern for some languages (such as Chinese). Hence, the performance of the proposed method is also evaluated on the text line detection. For this study, the MSRA-TD500 dataset has been chosen. The combined image of MSRA-TD500 and HUST-TR400 [51] has been trained for 1400 epochs. For single scale testing, the images are resized to 640 while preserving the aspect ratio. Similarly, for multi-scale testing, the sizes used are 256, 640, and 1024. The performance of the proposed method on the MSRA-TD500 dataset is mentioned in Table 2.

From Table 2, it was noted that the proposed method achieves similar performance irrespective of the color channel order. The NMS shows better results than the LANMS for both the single-scale testing and the multi-scale testing. The comparison of the NMS with LANMS shows that the predicted boxes are well clustered. Thus, the use of LANMS is to combine the bounding box for creating the bigger bounding box. The recall improves in most of the case due to this property of the LANMS. However, the precision drops because some boxes get merged, and the IoU threshold penalizes the prediction performance.

2) IoU RELIANCE DETECTION PERFORMANCE

The standard metric for text detection is based on the IoU threshold of 0.5. A higher threshold requires more efficient and robust methods. Hence, this analysis is performed to explain the effectiveness of the proposed method better. For this analysis, the datasets, i.e., ICDAR2015, ICDAR2013, and MSRA-TD500 with publicly available testing annotation, are selected. The IoU threshold of 0.5, 0.6, 0.7, 0.8, and 0.9 are considered to analyze this trade-off. It includes IoU vs. Precision, IoU vs. Recall, and IoU vs. F-measure analysis. The computed values at the thresholds mentioned above are shown in Fig. 7. From Fig. 7, it has been observed that the proposed method is better performing for the dataset



(a) IoU vs. Precision (b) IoU vs. Recall (c) IoU vs. F-measure

FIGURE 7. Trade-off (a) IoU vs. Precision, (b) IoU vs. Recall, (c) IoU vs. F-Measure.

which has images that were captured irrespective of the presence of text in it (i.e., incidental scene text) as compared to the images which have focused text (i.e., focused scene text). In the case of focused scene text, i.e., ICDAR2013 and MSRA-TD500, the proposed method sustains the performance for MSRA-TD500 as the IoU threshold increases compared to the ICDAR2013 dataset. In MSRA-TD500, at IoU of 0.5, some word-level detection appears in place of line-level detection. It penalizes both precision and recall. However, as the IoU threshold increases, the candidate bounding boxes have covered more ground truth areas in MSRA-TD500 than ICDAR2013. From Fig. 7, it has been observed that the prediction performance deteriorates as the bounding box area increases. It is best for the incidental scene text dataset (ICDAR2015), which further drops for the focused word-level bounding box dataset (ICDAR2013), and it further drops for the line-level bounding box dataset (MSRA-TD500).

3) GENERALIZATION ABILITY

In the current trend for scene text detection, the inclination is to train and test on the same dataset. However, the machine learning-based method's final objective is to perform good in unseen real-world data without training. For accomplishing the final objective of machine learning-based methods, two datasets close to the real-world environment are chosen, namely, COCO-Text, and SVTD. In COCO-Text, the images are taken irrespective of the compulsion of the presence of text in the image, making this dataset very challenging. The evaluation of the COCO-Text required the rectangular bounding box, whereas our method predicts the quadrilateral bounding box. Hence, the fittest rectangle enclosing the quadrilateral bounding box is taken for evaluation purposes. The second dataset is SVTD, which is based on the Google Street View Images. Our ICDAR2015 trained model is directly applied to COCO-Text and SVTD datasets to check the trained model's generalization ability. The results are mentioned on the Table 3. From Table 3, it has been concluded that the proposed method, trained on the ICDAR2015 dataset, has shown satisfactory performance on the unseen dataset.

TABLE 3. Detection performance of methods on untrained dataset.

Dataset	Precision	Recall	F-Measure
COCO-Text	0.642	0.586	0.613
SVTD	0.857	0.459	0.598

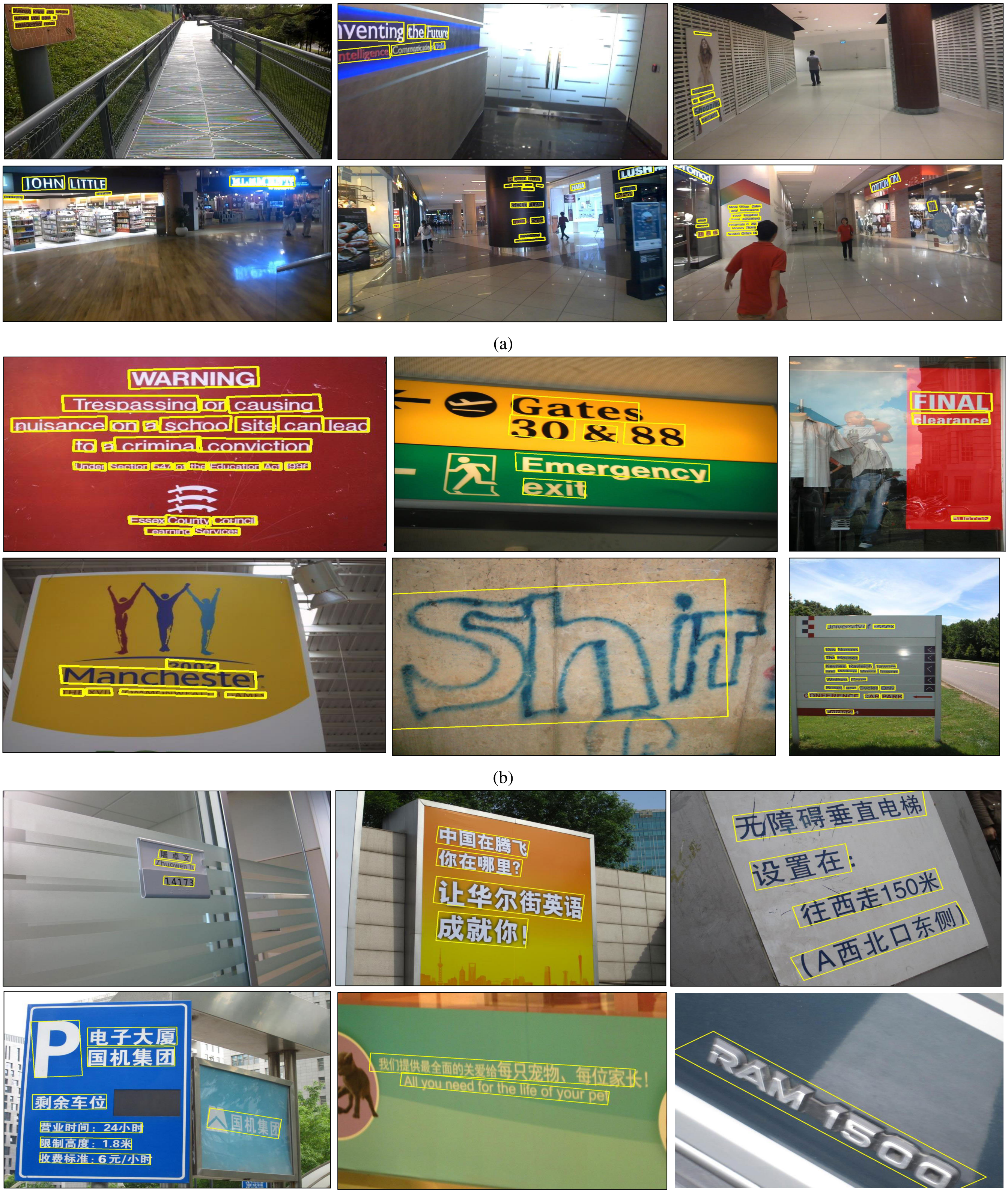
B. QUALITATIVE RESULTS

The text detection metric is borrowed from object detection and is based on the predicted box's IoU over ground truth. Nevertheless, if we understand this metric, then it has a significant drawback for text detection. If a bounding box does not cover the text's full height and only covers the top half part and still achieves the IoU of 0.5, it is considered a true positive case. It may provide good quantitative results, but qualitatively the results are not good enough. Hence, we also presented the qualitative results of the proposed method. The qualitative results of the proposed method on various used datasets have been shown in Fig. 8. From Fig. 8, it is clear that the proposed method can detect the skewed, multi-oriented, and handwritten text.

The qualitative results obtained from the generalization test on COCO-Text and SVTD are shown in Fig. 9. From Fig. 9 it is also clear that the trained model even works satisfactorily on untrained datasets. It is analyzed from Fig. 9 that the proposed method can detect the text written on clothes (some part occluded by the gloves), at unusual pose (written on the bus surface), text written on building with strokes likes structures. The used model is trained on the ICDAR2015 dataset, which contains shopping malls, metro stations, and few outdoor scene images. In contrast, the COCO-Text contains images from a vast spectrum covering both indoor and outdoor images. The SVTD dataset comprises street view images, including deformation such as blur and noise. Qualitatively, it has been observed that the trained model is not explicitly working only for a trained dataset but also producing reasonable results on untrained datasets.

C. ABLATION STUDY

An ablation study has been conducted to analyze the impact of pyramid feature fusion (PFF), hyperparameter λ, and the

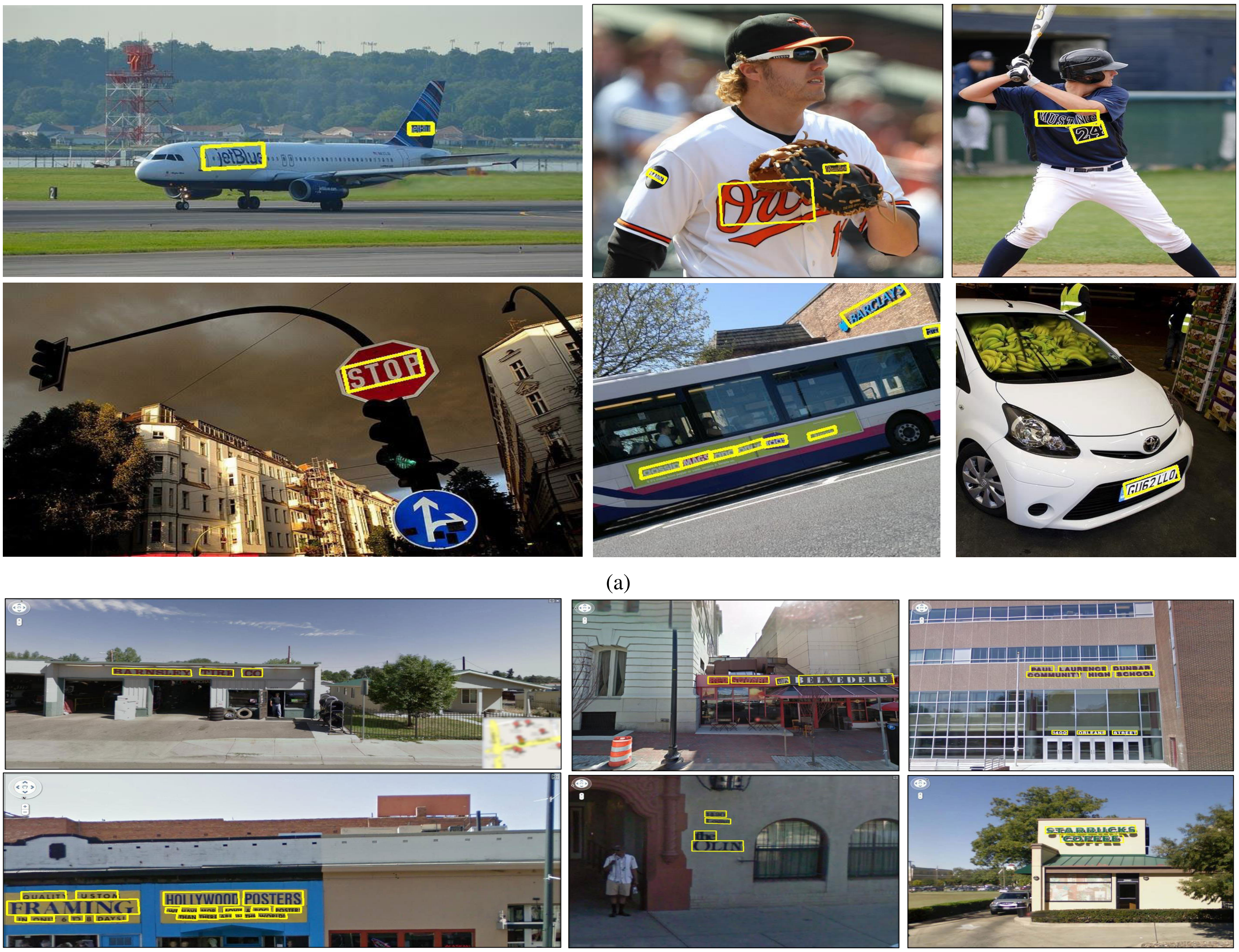


(a)

(b)

(c)

FIGURE 8. Qualitative results for (a) ICDAR2015 dataset, (b) ICDAR2013 dataset, (c) MSRA-TD500 dataset.



(a)

(b)

FIGURE 9. Qualitative results for untrained dataset used to check the generalization of the trained model on (a) COCO-Text dataset, (b) SVTD dataset.

significance of quadrilateral over rotated rectangle representation on the proposed approach.

Impact of pyramid feature fusion: In the proposed method, the PFF block has been included to combine the multi-scale features. The impact of the PFF block has been investigated in Table 4. In Table 4, the impact of the PFF block is clearly visible. The use of the PFF block improves the detection performance of the bounding box prediction.

TABLE 4. Ablation study to show the impact of the pyramid feature fusion.

Dataset	PFF	Precision	Recall	F-measure
ICDAR2015	✓	0.905	0.790	0.844
	✗	0.935	0.721	0.814
ICDAR2013	✓	0.901	0.715	0.797
	✗	0.871	0.692	0.771
MSRA-TD500	✓	0.853	0.619	0.717
	✗	0.837	0.602	0.698

Impact of λ: In this work, the λ hyperparameter is used to control the trade-off between the false-positive and false-negative rates. To show this ablation, we have chosen various values of λ and train model for ICDAR2015 dataset and summarized in Table 5. The choice of λ helps to improve the precision at the cost of a small drop in the recall. But overall, the f-measure increases. Hence, we choose the λ value as 0.8 in our proposed method.

Quadrilateral vs. rotated rectangle boxes: This ablation helps to inspect the capability of our proposed method for predicting the quadrilateral as well as a rotated bounding box. It is verified by converting the quadrilateral bounding box annotation to the rotated rectangle one, using a minimum area rectangle to fit the quadrilateral. The training is conducted on both quadrilateral and rotated rectangle annotation by using the proposed method. For this ablation, we have considered the ICDAR2015 dataset. The results are summarized in Table 6. The results show that the proposed representation performance is good for the quadrilateral bounding box as compared to the rotated rectangle bounding box.

TABLE 5. Impact of the λ hyperparameter in the model for ICDAR2015 dataset.

λ	Precision	Recall	F-measure
0.2	0.830	0.816	0.823
0.4	0.886	0.792	0.836
0.6	0.888	0.796	0.839
0.8	0.905	0.790	0.844

TABLE 6. Detection performance to analyze the significance of centroid-centric vector regression approach for Quadrilateral annotation and rotated rectangle annotation.

Annotation (used in training)	Precision	Recall	F-Measure
Quadrilateral	0.905	0.790	0.844
Rotated Rectangle	0.910	0.755	0.826

TABLE 7. Comparison of the proposed method with state-of-the-art approaches for ICDAR2015 dataset. The best and second best results are boldfaced. [∗: Multi scale testing].

Method	Recall	Precision	F-Measure
EAST [28]	0.783	0.832	0.807
TextBoxes++ [22]	0.785	0.878	0.829
DR [29]	0.800	0.820	0.810
DMPNet [16]	0.682	0.732	0.706
FTSN [54]	0.800	0.886	0.841
RRD [30]	0.800	0.880	0.838
IncepText [32]	0.806	0.905	**0.853**
Wang etal [31]	0.867	0.821	0.843
QuadBox	0.790	0.905	0.844
QuadBox∗	0.818	0.887	**0.851**

TABLE 8. Comparison of the proposed method with state-of-the-art approaches for ICDAR2013 dataset. [∗: Multi scale testing].

Method	IC13 Eval			Deteval		
	R	P	FM	R	P	FM
TextFlow [55]	0.76	0.85	0.80	-	-	-
He et al. [56]	0.76	0.85	0.80	-	-	-
He et al. [57]	0.73	0.93	0.82	-	-	-
Tian et al [58]	0.84	0.84	0.84	–	–	–
Qin et al. [59]	0.79	0.89	0.83	–	–	–
TextBoxes [19]	0.74	0.86	0.80	0.74	0.88	0.81
TextBoxes∗ [19]	0.83	0.88	0.85	0.83	0.89	0.86
TextBoxes++ [22]	0.74	0.86	0.80	0.74	0.88	0.81
TextBoxes++∗ [22]	0.84	0.91	0.88	0.86	0.92	0.89
QuadBox	0.70	0.90	0.79	0.71	0.90	0.79
QuadBox∗	0.81	0.88	0.84	0.81	0.89	0.85

TABLE 9. Comparison of the proposed method with state-of-the-art approaches for COCO-Text dataset. The best and second best results are boldfaced. [∗: Multi scale testing].

Method	Precision	Recall	FM
RRD [30]	0.640	0.570	0.610
Lyu et al. [27]	0.725	0.529	0.611
Lyu et al.* [27]	0.629	0.622	**0.626**
Textboxes ++ [22]	0.558	0.560	0.559
Textboxes ++ * [22]	0.567	0.608	0.587
SR-DeepText [38]	0.478	0.593	0.529
QuadBox	0.642	0.586	**0.613**

D. COMPARISON WITH SOTA

Quadrilateral Text Detection For the quadrilateral text detection, the ICDAR2015 dataset is used, and the proposed method is compared with the other state-of-the-art methods and reported in Table 7. For a fair comparison, the quadrilateral bounding box prediction methods are only considered in the comparison. Our method outperforms most of the existing state-of-the-art quadrilateral bounding box prediction methods. Only the IncepText [32] is marginal better performing than the proposed method with the difference in F-measure of .002.

Detecting Horizontal Text: Our work is mainly focused on text detection of the quadrilateral bounding box annotated dataset. However, for the broad applicability of the proposed approach, the horizontal text annotated dataset is also used. Two horizontal annotated datasets, namely, ICDAR2013 and COCO-Text, are taken. For a fair comparison of the ICDAR2013 dataset, the IC13 Eval [52] and DetEval [53] metrics are considered. SOTA methods have used these metrics for the ICDAR2013 dataset evaluation. The proposed method's result is compared with other state-of-the-art approaches for ICDAR2013 and summarized in Table 8.

Since the COCO-Text dataset has many versions of the annotation, the V1.4 annotation of the COCO-Text dataset is used for a fair comparison. The same has been used on the ICDAR2017 robust reading competition. The comparison has been shown in Table 9. We achieved the second-best results on the COCO-Text dataset without having the model trained on the COCO-Text dataset. However, from Table 9, it is clear that our method achieves state-of-the-art performance if a single scale testing is taken into consideration. Even the single scale testing of our model on the COCO-Text dataset has achieved similar performance to the other multi-scale testing SOTA methods.

Line level text detection The performance of the proposed method is validated on the line-level bounding box predictions. The comparison has been made with other state-of-the-art approaches in Table 10. The MSRA-TD500 is both line-level text detection and a multi-lingual dataset. Our method is also robust to generate good results on multi-lingual and line-level text detection dataset from the comparison.

From the comparison of the proposed method with other state-of-the-art approaches, we have found that the proposed method's performance is one of the best in quadrilateral bounding box representation, i.e., for the ICDAR2015 dataset. The competitive methods for ICDAR2015 dataset are IncepText [32] and [31]. The advantage of the proposed approach over IncepText [32] is that the proposed system removes the need for the ROI pooling operation and still achieves similar results. The proposed method has a drop of 0.002 in F-measure in comparison with [32]. This ROI pool removal makes the proposed system simple at the cost of the marginal drop in performance. On the other hand, the proposed method is a single-stage network, whereas [31] is a two-stage network. Usually, a two-stage

TABLE 10. Comparison of the proposed method with state-of-the-art approaches for MSRA-TD500 dataset [∗: Multi scale testing, †: Segmentation-based approach, ‡ regression-based approach].

Method	Precision	Recall	H-Mean
Yin et al [60]	0.71	0.61	0.66
Kang et al [61]	0.71	0.62	0.66
Yin et al [62]	0.81	0.63	0.71
Zhang et al † [63]	0.83	0.67	0.74
Yao et al † [33]	0.76	0.75	0.75
EAST + VGG ‡ [28]	0.81	0.61	0.70
He et al. [57]	0.71	0.61	0.69
DeepReg ‡ [29]	0.77	0.70	0.74
RRPN ‡ [20]	0.82	0.68	0.74
RRD † [30]	0.87	0.73	0.79
PixelLink † [35]	0.83	0.73	0.77
Corner ‡ [27]	0.87	0.76	0.81
TextSnake ‡ [64]	83.2	0.73	0.78
CRAFT † [37]	0.88	0.78	0.82
SAE † [36]	0.84	0.81	0.82
DB † [39]	0.91	0.79	0.84
Quadbox ‡	0.80	0.59	0.71
Quadbox ∗ ‡	0.84	0.68	0.75

network performs better than a single-stage, but our method outperforms [31] by the margin of 0.008 in F-measure.

The proposed system also achieves similar performance to the state-of-the-art method for the COCO-Text dataset, even with the cross-modal validation. The competitor method is corner [27] for the COCO-Text dataset. The [27] method is similar to ours, but they detected several corners and performed segmentation. However, inference steps are complex. It includes the grouping and sampling of corner points for probable text regions and then uses Rotated Position-Sensitive ROI Average Pooling for box scoring. In contrast, our approach does not rely on expensive ROI pooling operation. Our method utilizes the geometric property of the quadrilateral bounding box, which makes the proposed system simple yet effective. In Table 7 and Table 9, the proposed method shows its efficient performance for incidental scene text.

Also, for the horizontal scene text dataset, the proposed system shows good results. However, the method such as TextBoxes [19] and TextBoxes++ [22] are based on the prior anchor boxes. Whereas our proposed approach removes the requirement of prior boxes for text detection. Again, this makes the proposed system simple and effective. The geometrical property helps to get better results for quadrilateral representation (see Table 7) and competitive in horizontal box representation. However, the proposed method's limitation appears due to the longer length text instance for the line-level annotated dataset. For line-level annotation, methods such as PixelLink [35], SAE [36], and DB [39] shows better results than our method. These are the segmentation-based method, and learning segmentation for the character and space between characters as a text region seems easier for line-level text detection than the regression-based approach.

VI. CONCLUSION

This paper proposed a quadrilateral bounding box prediction that combines the philosophy of direct and indirect regression for text detection in the wild. Here, the geometry of quadrilateral for text detection is exploited to improve performance. The simplistic approach is based on single-stage detection with standard non-maximal suppression and region removal multi-scale testing. The qualitative and quantitative results show the effectiveness of the proposed method on publicly available datasets. The proposed method also performs well in the generalization test, conducted on the proposed SVTD dataset and publicly available COCO-Text dataset. In the future, the semantic information of object could be integrated into the network for further improvement in the accuracy.

ACKNOWLEDGMENT

The authors would like to thank Dr. Marcus Liwicki, Chaired Professor, EISLAB ML, Luleå University of Technology, Sweden, for providing the computing resources to conduct extensive experiments.

REFERENCES

[1] 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.

[2] A. Sain, A. K. Bhunia, P. P. Roy, and U. Pal, "Multi-oriented text detection and verification in video frames and scene images," *Neurocomputing*, vol. 275, pp. 1531–1549, Jan. 2018.

[3] A. K. Bhunia, G. Kumar, P. P. Roy, R. Balasubramanian, and U. Pal, "Text recognition in scene image and video frame using color channel selection," *Multimedia Tools Appl.*, vol. 77, no. 7, pp. 8551–8578, Apr. 2018.

[4] P. Keserwani, K. De, P. P. Roy, and U. Pal, "Zero shot learning based script identification in the wild," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Sep. 2019, pp. 987–992.

[5] G. Schroth, S. Hilsenbeck, R. Huitl, F. Schweiger, and E. Steinbach, "Exploiting text-related features for content-based image retrieval," in *Proc. IEEE Int. Symp. Multimedia*, Dec. 2011, pp. 77–84.

[6] S. S. Tsai, H. Chen, D. Chen, G. Schroth, R. Grzeszczuk, and B. Girod, "Mobile visual search on printed documents using text and low bit-rate features," in *Proc. 18th IEEE Int. Conf. Image Process.*, Sep. 2011, pp. 2601–2604.

[7] Y. Dvorin and U. E. Havosha, "Method and device for instant translation," U.S. Patent 11 998 931, Jun. 6, 2009.

[8] C. Parkinson, J. J. Jacobsen, D. B. Ferguson, and S. A. Pombo, "Instant translation system," U.S. Patent 9 507 772, Nov. 29, 2016.

[9] G. N. Desouza and A. C. Kak, "Vision for mobile robot navigation: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 2, pp. 237–267, Feb. 2002.

[10] X. Liu and J. Samarabandu, "An edge-based text region extraction algorithm for indoor mobile robot navigation," in *Proc. IEEE Int. Conf. Mechatronics Autom.*, Jul. 2005, pp. 701–706.

[11] X. Liu and J. K. Samarabandu, "A simple and fast text localization algorithm for indoor mobile robot navigation," *Proc. SPIE*, vol. 5672, pp. 139–150, Mar. 2005.

[12] R. Schulz, B. Talbot, O. Lam, F. Dayoub, P. Corke, B. Upcroft, and G. Wyeth, "Robot navigation using human cues: A robot navigation system for symbolic goal-directed exploration," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 1100–1105.

[13] Y. K. Ham, M. S. Kang, H. K. Chung, R.-H. Park, and G. T. Park, "Recognition of raised characters for automatic classification of rubber tires," *Opt. Eng.*, vol. 34, no. 1, pp. 102–110, 1995.

[14] Z. He, J. Liu, H. Ma, and P. Li, "A new automatic extraction method of container identity codes," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 1, pp. 72–78, Mar. 2005.

[15] Y. Zhu, C. Yao, and X. Bai, "Scene text detection and recognition: Recent advances and future trends," *Frontiers Comput. Sci.*, vol. 10, no. 1, pp. 19–36, Feb. 2016.

[16] 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. 1962–1969.

[17] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu, F. Shafait, S. Uchida, and E. Valveny, "ICDAR 2015 competition on robust reading," in *Proc. 13th Int. Conf. Document Anal. Recognit. (ICDAR)*, Aug. 2015, pp. 1156–1160.

[18] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2315–2324.

[19] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "TextBoxes: A fast text detector with a single deep neural network," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 4161–4167.

[20] J. Ma, W. Shao, H. Ye, L. Wang, H. Wang, Y. Zheng, and X. Xue, "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Trans. Multimedia*, vol. 20, no. 11, pp. 3111–3122, Nov. 2018.

[21] X. Liu, D. Liang, S. Yan, D. Chen, Y. Qiao, and J. Yan, "FOTS: Fast oriented text spotting with a unified network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5676–5685.

[22] 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.

[23] 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.*, 2015, pp. 91–99.

[24] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[25] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, Oct. 2016, pp. 21–37.

[26] P. He, W. Huang, T. He, Q. Zhu, Y. Qiao, and X. Li, "Single shot text detector with regional attention," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3047–3055.

[27] P. Lyu, C. Yao, W. Wu, S. Yan, and X. Bai, "Multi-oriented scene text detection via corner localization and region segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7553–7563.

[28] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, and J. Liang, "EAST: An efficient and accurate scene text detector," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5551–5560.

[29] W. He, X.-Y. Zhang, F. Yin, and C.-L. Liu, "Deep direct regression for multi-oriented scene text detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 745–753.

[30] M. Liao, Z. Zhu, B. Shi, G.-S. 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.

[31] S. Wang, Y. Liu, Z. He, Y. Wang, and Z. Tang, "A quadrilateral scene text detector with two-stage network architecture," *Pattern Recognit.*, vol. 102, Jun. 2020, Art. no. 107230.

[32] Q. Yang, M. Cheng, W. Zhou, Y. Chen, M. Qiu, W. Lin, and W. Chu, "IncepText: A new inception-text module with deformable PSROI pooling for multi-oriented scene text detection," 2018, *arXiv:1805.01167*. [Online]. Available: http://arxiv.org/abs/1805.01167

[33] C. Yao, X. Bai, N. Sang, X. Zhou, S. Zhou, and Z. Cao, "Scene text detection via holistic, multi-channel prediction," 2016, *arXiv:1606.09002*. [Online]. Available: http://arxiv.org/abs/1606.09002

[34] M. Liao, P. Lyu, M. He, C. Yao, W. Wu, and X. Bai, "Mask TextSpotter: An end-to-end trainable neural network for spotting text with arbitrary shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 2, pp. 532–548, Feb. 2021.

[35] D. Deng, H. Liu, X. Li, and D. Cai, "PixelLink: Detecting scene text via instance segmentation," 2018, *arXiv:1801.01315*. [Online]. Available: http://arxiv.org/abs/1801.01315

[36] Z. Tian, M. Shu, P. Lyu, R. Li, C. Zhou, X. Shen, and J. Jia, "Learning shape-aware embedding for scene text detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4234–4243.

[37] 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. 9365–9374.

[38] Y. Zheng, Y. Xie, Y. Qu, X. Yang, C. Li, and Y. Zhang, "Scale robust deep oriented-text detection network," *Pattern Recognit.*, vol. 102, Jun. 2020, Art. no. 107180.

[39] M. Liao, Z. Wan, C. Yao, K. Chen, and X. Bai, "Real-time scene text detection with differentiable binarization," in *Proc. AAAI*, 2020, pp. 11474–11481.

[40] R. E. Deakin, S. C. Bird, and R. I. Grenfell, "The centroid? Where would you like it to be be?" *Cartography*, vol. 31, no. 2, pp. 153–167, Dec. 2002.

[41] W. Yang, X. Zhang, Y. Tian, W. Wang, J.-H. Xue, and Q. Liao, "Deep learning for single image super-resolution: A brief review," *IEEE Trans. Multimedia*, vol. 21, no. 12, pp. 3106–3121, Dec. 2019.

[42] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 764–773.

[43] W. Luo, Y. Li, R. Urtasun, and R. Zemel, "Understanding the effective receptive field in deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 4898–4906.

[44] P. Keserwani, T. Ali, and P. P. Roy, "TRPN: A text region proposal network in the wild under the constraint of low memory GPU," in *Proc. 4th IAPR Asian Conf. Pattern Recognit. (ACPR)*, Nov. 2017, pp. 286–291.

[45] P. Dollar, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1532–1545, Aug. 2014.

[46] W. Wang, S. Zhao, J. Shen, S. C. H. Hoi, and A. Borji, "Salient object detection with pyramid attention and salient edges," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1448–1457.

[47] C. Ma, X. Mu, and D. Sha, "Multi-layers feature fusion of convolutional neural network for scene classification of remote sensing," *IEEE Access*, vol. 7, pp. 121685–121694, 2019.

[48] S. S. M. Salehi, D. Erdogmus, and A. Gholipour, "Tversky loss function for image segmentation using 3D fully convolutional deep networks," in *Proc. Int. Workshop Mach. Learn. Med. Imag.*, 2017, pp. 379–387.

[49] A. Veit, T. Matera, L. Neumann, J. Matas, and S. Belongie, "COCO-text: Dataset and benchmark for text detection and recognition in natural images," 2016, *arXiv:1601.07140*. [Online]. Available: http://arxiv.org/abs/1601.07140

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

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

[52] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. I. Bigorda, S. R. Mestre, J. Mas, D. F. Mota, J. A. Almazan, and L. P. de las Heras, "ICDAR 2013 robust reading competition," in *Proc. 12th Int. Conf. Document Anal. Recognit.*, Aug. 2013, pp. 1484–1493.

[53] C. Wolf and J.-M. Jolion, "Object count/area graphs for the evaluation of object detection and segmentation algorithms," *Int. J. Document Anal. Recognit. (IJDAR)*, vol. 8, no. 4, pp. 280–296, Sep. 2006.

[54] Y. Dai, Z. Huang, Y. Gao, Y. Xu, K. Chen, J. Guo, and W. Qiu, "Fused text segmentation networks for multi-oriented scene text detection," in *Proc. 24th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2018, pp. 3604–3609.

[55] S. Tian, Y. Pan, C. Huang, S. Lu, K. Yu, and C. L. Tan, "Text flow: A unified text detection system in natural scene images," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4651–4659.

[56] D. He, X. Yang, W. Huang, Z. Zhou, D. Kifer, and C. L. Giles, "Aggregating local context for accurate scene text detection," in *Proc. Asian Conf. Comput. Vis.*, 2016, pp. 280–296.

[57] T. He, W. Huang, Y. Qiao, and J. Yao, "Text-attentional convolutional neural network for scene text detection," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2529–2541, Jun. 2016.

[58] C. Tian, Y. Xia, X. Zhang, and X. Gao, "Natural scene text detection with MC–MR candidate extraction and coarse-to-fine filtering," *Neurocomputing*, vol. 260, pp. 112–122, Oct. 2017.

[59] S. Qin and R. Manduchi, "A fast and robust text spotter," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2016, pp. 1–8.

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

[61] L. Kang, Y. Li, and D. Doermann, "Orientation robust text line detection in natural images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 4034–4041.

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

[63] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai, "Multi-oriented text detection with fully convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4159–4167.

[64] 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.*, Sep. 2018, pp. 20–36.

PRATEEK KESERWANI received the B.Sc. and M.Sc. (CS) degrees from the University of Allahabad, Allahabad, India, in 2008 and 2010, respectively, and the M.Tech. degree from the University of Allahabad in 2015. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, IIT Roorkee, Roorkee, India. His current research interests include computer vision and deep learning.

ANKIT DHANKHAR received the B.Tech. degree in computer science and engineering from IIT Roorkee, Roorkee, India, in 2020. He was a Research Intern with Adobe Big Data Laboratory, Bangalore, in 2019, and a Research Assistant with the Multimedia and Interactive Computing Laboratory, NTU Singapore, in 2019. His current interest includes computer vision and natural language processing.

RAJKUMAR SAINI received the Ph.D. degree from the Department of Computer Science and Engineering, IIT Roorkee, Roorkee, India. He is currently a Postdoctoral Researcher with the EISLAB Machine Learning, Luleå University of Technology, Sweden. His research interests include computer vision, machine learning, pattern recognition, human–computer interface, brain signal analysis, and digital image processing.

PARTHA PRATIM ROY (Member, IEEE) was a Postdoctoral Research Fellow with the RFAI Laboratory, France, in 2012, and the Synchromedia Laboratory, Canada, in 2013. He was with the Advanced Technology Group, Samsung Research Institute, Noida, India, from 2013 to 2014. He is currently an Associate Professor with the Department of Computer Science and Engineering, IIT Roorkee, India. He has authored or coauthored more than 200 articles in international journals and conferences. His research interests are pattern recognition, bio-signal analysis, EEG-based pattern analysis, and multilingual text recognition. He is an Associate Editor of the *IET Image Processing*, *IET Biometrics*, *IEICE Transactions on Information and Systems* and *Springer Nature Computer Science*.