Disentangled Contour Learning for Quadrilateral Text Detection

Yanguang Bi, Zhiqiang Hu✉
SenseTime Research
{biyanguang, huzhiqiang}@sensetime.com

Abstract

Precise detection of quadrilateral text is of great significance for subsequent recognition, where the main challenge comes from four distorted sides. Existing methods concentrate on learning four vertices to construct the contour. However, vertices are dummy intersections entangled by their neighbor sides. The regression of each vertex would simultaneously affect its two neighbor sides. As a result, the originally independent side would be influenced by two different vertices which further inevitably disturb other sides. The above entangled vertices learning suppresses the learning efficiency and detection performance. In this paper, we proposed disentangled contour learning network (DCLNet) to focus on clear regression of each individual side disentangled from the whole quadrilateral contour. The side is parameterized by a linear equation that disentangled in the polar coordinates for easier learning. With tailored Ray-IoU loss and sine angle loss, DCLNet could better learn the representation of each disentangled side without being disturbed by others. The final quadrilateral text contour is easily constructed by intersecting the predicted linear equations of sides. Empirically, the proposed DCLNet achieves state-of-the-art detection performances on three scene text benchmarks. Ablation study is also presented to demonstrate the effectiveness of proposed disentangled contour learning framework.

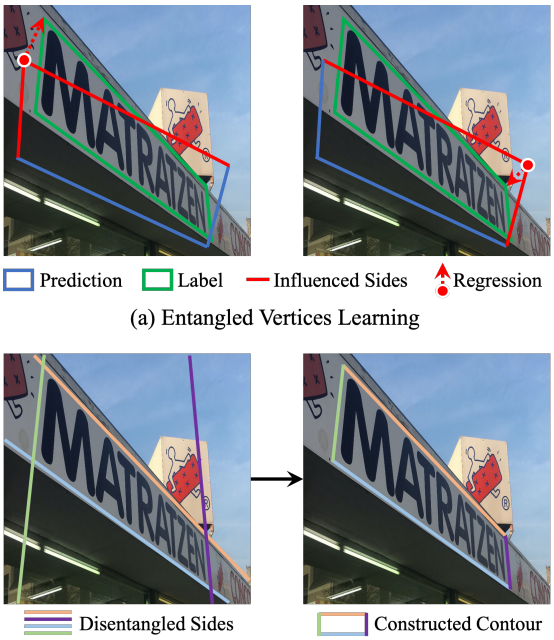

Figure 1. (a) Entangled vertices learning. The regression of each independent side is influenced by two vertices which would inevitably disturb other sides. The learning efficiency and detection performance are suppressed. (b) Disentangled contour learning in our DCLNet. The sides are disentangled from whole contour with independent line parameters in polar coordinates, which is easily learned with higher efficiency and performance.

1. Introduction

Scene text detection plays an importance role in various applications, such as authentication, product retrieve, text translation and autonomous driving. Recently, deep learning based text detection methods [32, 13, 29, 25, 3, 24] exhibit promising performances which benefits from the development of network architecture [31, 7] and detection or segmentation pipelines [12, 19, 27, 22, 28, 6]. However, many regular texts tend to appear as distorted quadrilateral shapes in image due to the changeable views of camera. In this situation, the frequently used axis-aligned rectangle and rotated rectangle are not able to precisely localize the quadrilateral shapes, which severely impairs the subsequent recognition. Therefore, more and more methods concentrate their efforts on the quadrilateral text detection [9, 20, 46, 21, 18, 16, 34].

The main challenge of quadrilateral text detection comes from the four distorted sides, which present irregular and independent arrangements. As shown in Figure 1(a), existing mainstream methods detect the quadrilateral boundary by regressing the four vertices of texts. Some of these methods [20, 34, 16, 18] learn the regression based on prior anchor boxes (two-stage) and others [21, 46, 9] di-

rectly regress the offset between current location and vertices (one-stage). However, the vertices are actually dummy intersections entangled by their two neighbor sides. In this situation, the regression of each vertex would simultaneously influence the localization of its two neighbor sides. As a result, the originally independent contour side would be influenced by two different vertices which further inevitably disturb other sides. The above entangled vertices learning suppresses the learning efficiency and detection performance as shown in following experiments.

To address the above interfering learning process caused by entangled vertices, we proposed disentangled contour learning network (DCLNet) in this paper to focus on more clear regression of each individual side disentangled from the whole quadrilateral contour shown in Figure 1(b). Especially, each side is parameterized by its own linear equation and further disentangled in polar coordinates with physical meanings in image for easier learning. With the tailored Ray-IoU loss and sine angle loss, DCLNet could better learn each disentangled side without being disturbed by others, which promotes the learning efficiency and detection performance. The quadrilateral text contour could be easily constructed by intersecting the predicted linear equations of four sides. The proposed DCLNet achieves state-of-the-art detection performances on three scene text detection benchmarks: ICDAR 2017 MLT, ICDAR 2015 and MSRA-TD500. Extensive ablation study is also performed to validate the effectiveness of proposed disentangled contour learning framework.

The contributions of this work are summarized as follows: (1) We analyzed the shortcoming of existing entangled vertices based methods and further proposed disentangled sides parameterized in polar coordinates to represent the quadrilateral contour; (2) A unified disentangled contour learning network (DCLNet) is proposed with tailored Ray-IoU loss and sine angle loss to better detect quadrilateral texts; (3) The proposed DCLNet achieves state-of-the-art performances on challenging ICDAR 2017 MLT, ICDAR 2015 and MSRA-TD500 benchmarks, which contain arbitrary quadrilateral and multi-oriented texts.

2. Related Work

Recently, more and more scene text detection methods are proposed with the development of deep learning. For instance, CTPN [32] detects the horizontal texts by grouping regressed adjacent text components based on the Faster-RCNN [27] framework. However, the scene texts tend to appear with distorted and irregular shapes due to the various viewpoints of camera. Therefore, the majority of detection methods concentrate their efforts on multi-orientated texts which are represented by rotated rectangles and more general arbitrary quadrilateral shapes.

For multi-orientated texts, R2CNN [13] and RRPN [25] constructs prior anchor boxes with different rotations as proposals which are adapted in two-stage FasterRCNN [27] framework for detection. SegLink [29] detects the text fragments and predicts the link between adjacent text fragments to complete the oriented text boundary based on SSD [19] framework. [8] also employs the SSD framework to regress the offset of rotated anchor boxes with attention mechanism and improved hierarchical inception module. CLRS [24] learns the corner points of boundary box which are sampled and grouped to construct the final boundary. EAST [46] directly regresses the bounding box with four distances and rotation angle, which is similar to the DenseBox [12]. Besides the above regression methods, PixelLink [3] segments the text region and predicts the neighborhood connections of each pixel to obtain instances which are surrounded by rotation boxes as detection results. Border [40] segments the center region of text and employ semantics-aware border detection technique to produce four types of text border to extract each scene text.

Rotation rectangles are not able to precisely localize distorted quadrilateral texts, which would damage the downstream recognition. As a result, EAST [46] also present the direct quadrilateral regression on the eight coordinates of four vertices, as well as the DeepReg [9]. Besides the above anchor-free methods, most of existing methods employ anchor based two stage framework. Textboxes++ [16] employ axis-aligned rectangles as the anchor boxes and regress the offset between anchor box and boundary label. DMPNet [20] designs quadrilateral sliding windows as anchor boxes to better match the quadrilateral texts. RRD [18] and [34] also follow the two stage framework which further propose oriented response architecture and weighted RoI (Region of Interest) pooling respectively to better extract the quadrilateral text features. Although the regression manners and feature extractions are different, it can be seen that the common operation for above methods is to regress the four vertices of quadrilateral contour. As analyzed before, the independent sides would be affected by other sides due to their entangled vertices, which suppresses the learning efficiency and detection performance.

For more complicated scene texts with arbitrary curved shapes, most detection methods are based on segmentation. CRAFT [1] predicts the Gaussian heatmap of characters and their affinity link to obtain text instances. PSENet [35] learns the text kernels on different scales and progressively integrate them into different instances. TextCohesion [37] and TextMountain [47] segment the center region of text to avoid adhesion, then assign boundary pixels to corresponding center to complete text instance. DB [17] improves the binarization in text segmentation with differentiable learning with network. Note that the above methods for arbitrary shapes also adapt quadrilateral texts and would be compared as baselines in following benchmark experiments.

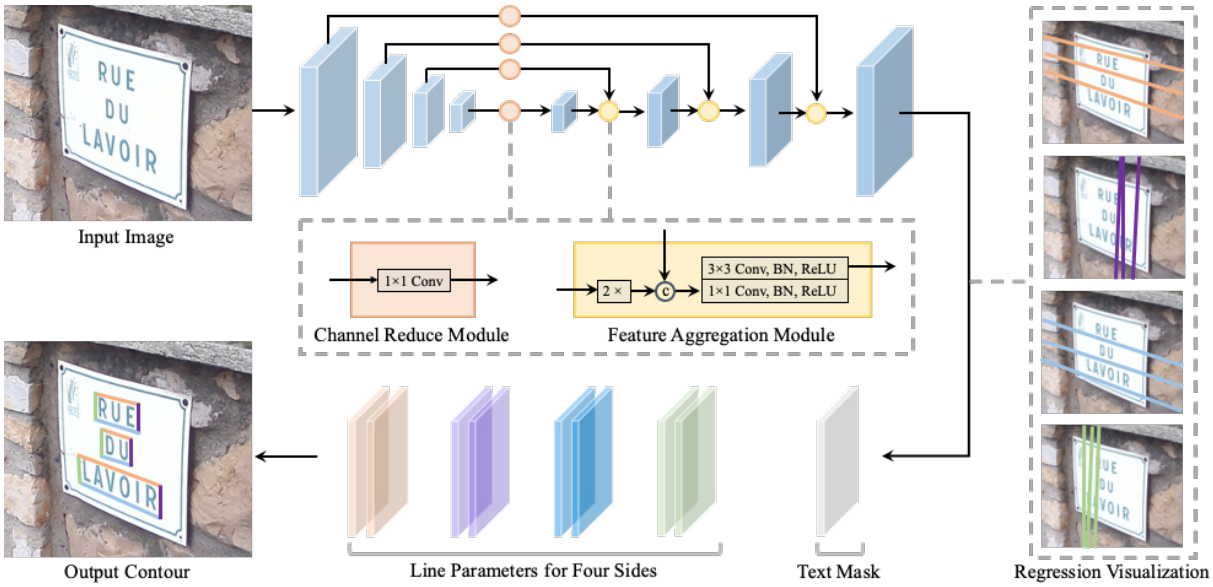

Figure 2. The overview of DCLNet framework, which employs typical encoder-decoder architecture connected by channel reduce module and feature aggregation module. The output feature maps contains nine channels denoted by different colors and further visualized using demo lines in the right. The text boundaries are constructed by intersecting the line equations of sides.

3. Method

In this section, we first present the overview of DCLNet framework. Next, the line equations of quadrilateral sides are formulated in polar coordinates for easier learning. Then, the loss functions tailored for line parameters are described. Finally, the detailed generation of line parameter labels is presented.

3.1. Overview

The overview of DCLNet framework is presented in Figure 2. The entire network follows typical encoder-decoder architecture which is connected by the channel reduce module and feature aggregation module to reduce computation and better utilize multi-scale features. The output feature maps contains nine channels denoted by different colors. The text mask is used to focus text region where the line equation parameters of disentangled sides are regressed. For easier observation, the regressed line parameters of different sides are visualized using the colors of corresponding features and shown by the demo lines in the right. The output text boundaries could be easily constructed by intersecting the line equations of sides.

It could be seen that the main pipeline differences between the proposed DCLNet and existing detection methods are the representations for quadrilateral text boundary and subsequent learning process. In the entangled vertices learning of existing baselines, the originally independent sides would be inevitably disturbed by the neighbour sides due to the regression on their entangled vertices, which suppresses the learning efficiency and detection performance. In contrast, DCLNet explicitly disentangles each side from the whole text contour which is further parameterized using independent line equation. The regression of each side would not disturb others which promotes the learning efficiency and the detection performance.

3.2. Disentangled Line Equation

The four sides of quadrilateral text boundary could be viewed as four independent parameterized line equations. For instance, Figure 3 presents the line equations of four sides with respect to current location which is considered as the origin of cartesian coordinates for easier convergence in training stage. The order of four sides is kept similar to the standard annotation which describes the text contour from the top left corner in clockwise direction. Generally, the line equation of one side could be formulated as:

$$Ax + By + C = 0, \tag{1}$$

where A, B, C are the three parameters. However, these parameters are actually redundant when $C \neq 0$. In other words, predicting implicit parameters A, B, C is ill-posed since multiple solutions exist. Their physical meanings in image are also ambiguous which is not conducive to network learning. As a result, we convert the above line equation into polar coordinates and obtain:

$$\rho = x\cos\theta + y\sin\theta, \tag{2}$$

where ρ and θ are two independent parameters and have physical meanings in image. As shown in Figure 3, ρ is

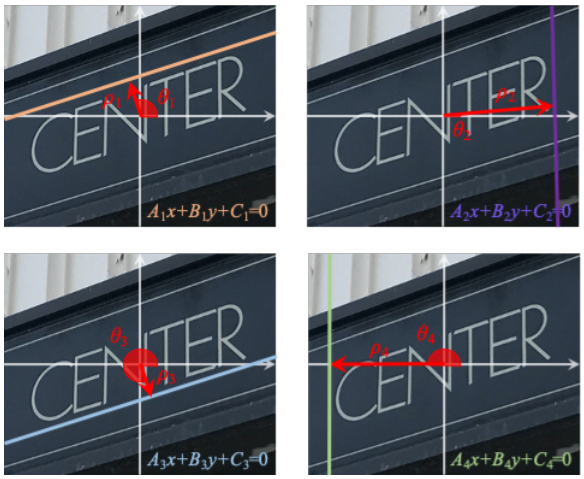

Figure 3. The visualization of four parameterized line equations with cartesian coordinates (orange, purple, blue and green) and polar coordinates (red).

the nearest distance between origin (current location) and line equation, θ is the rotation angle from x-axis positive direction to above nearest direction. Therefore, the conversion from cartesian coordinates to polar coordinates simultaneously reduces the number of parameters and the correlation between them, and brings physical meanings in images, which is beneficial for network learning.

In the inference stage, firstly, the original A, B, C parameters predicted by each location could be obtained by

$$A = \cos\theta, B = \sin\theta, C = -\rho. \tag{3}$$

Then the above line parameters are shifted to one global origin, i.e., the lower left corner of image for alignment. In this way, the intersected vertex (x_i, y_i) of two line equations (A_1, B_1, C_1), (A_2, B_2, C_2) in cartesian coordinates could be easily obtained by

$$D = A_1B_2 - A_2B_1, \tag{4}$$

$$x_i = \frac{B_1C_2 - B_2C_1}{D}, \tag{5}$$

$$y_i = \frac{A_2C_1 - A_1C_2}{D}, \tag{6}$$

where D should be checked avoid 0 to confirm meaningful intersected point.

In summary, the line equation of each side is converted to polar coordinates with respect to current location for easier training. In inference, the regressed parameters are converted back to one global cartesian coordinates to calculate the intersected points as detection results. The disentangled idea is reflected in two aspects: (1) The originally independent sides are disentangled from whole contour using line equation representation; (2) The parameters of line equation are disentangled in polar coordinates for easier learning.

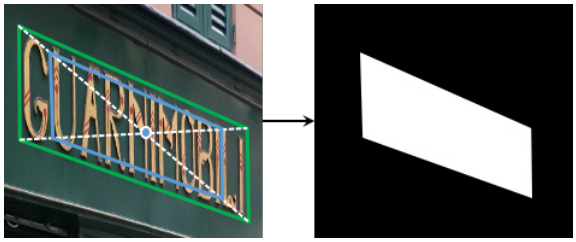

Figure 4. The shrunk version (blue box) of original text boundary (green box) based on anchor point (blue point) is viewed as classification label (right).

3.3. Label Generation

The labels contain: (1) text classification, (2) distance ρ and (3) angle θ. Here we describe the detailed label generation based on standard text annotation with four ordered vertices $(x_i, y_i), i = 0, 1, 2, 3$.

For the text classification label, we scale down the original text mask as shown in Figure 4 considering that the ρ in edge areas is hard to distinguish. Specifically, the intersection of two diagonals is viewed as anchor point. Each vertex is moved to the anchor point according to preset ratio to construct the shrunk version of mask as label.

The label maps of distance ρ and angle θ are specified based on different locations. For instance, the labels $(\widehat{\rho}_i, \widehat{\theta}_i), i = 1, 2, 3, 4$ for one location (x_0, y_0) in the text region could be calculated in Algorithm 1 given the four text vertices $(x_i, y_i), i = 1, 2, 3, 4$. Firstly, the parameters A, B, C of line equation with end points $(x_i, y_i), (x_j, y_j)$ are calculated. Then the distance label $\widehat{\rho}_i$ could be obtained with the nearest distance formula. For the angle label $\widehat{\theta}_i$, we can calculate the angle between the unit vector $\vec{e}$ in x-axis positive direction and the perpendicular line vector $\vec{l}$. Note that the positive sign of BC denotes the $\vec{l}$ is under the x-axis, which needs to be subtracted from 2π to obtain the real label $\widehat{\theta}_i$.

3.4. Loss Function

The overall loss function is formulated as

$$L = \lambda_1 L_{cls} + \lambda_2 L_\rho + \lambda_3 L_\theta, \tag{7}$$

where L_{cls}, L_ρ and L_θ denote the losses of text/non-text classification, line equation parameters ρ regression and θ regression, respectively.

L_{cls} is the commonly employed binary cross-entropy loss shown in Eq. 8 with OHEM (Online Hard Example Mining) [30] strategy to promotes the classification ability. Assuming that $\mathcal{M}$ is the training mask to ignore invalid text regions, $\widehat{y}_i$ and y_i denote the label and prediction in the ith location, respectively.

$$L_{cls} = \frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \left(-\widehat{y}_i \log y_i - (1 - \widehat{y}_i) \log (1 - y_i)\right). \tag{8}$$

Algorithm 1: Label $(\widehat{\rho}_i, \widehat{\theta}_i), i = 1, 2, 3, 4$ for the location (x_0, y_0)

Input: location (x_0, y_0), four text vertices $(x_i, y_i), i = 1, 2, 3, 4$
Output: $(\widehat{\rho}_i, \widehat{\theta}_i), i = 1, 2, 3, 4$

for $i = 1, 2, 3, 4$ **do**
  $j = mod(i,\ 4) + 1$;
  $A = y_j - y_i$;
  $B = x_i - x_j$;
  $C = x_j y_i - x_i y_j$;
  $\widehat{\boldsymbol{\rho}}_{\boldsymbol{i}} = \dfrac{|C|}{\sqrt{A^2 + B^2}}$;
  $\vec{l} = (\dfrac{-AC}{A^2 + B^2}, \dfrac{-BC}{A^2 + B^2})$;
  $\vec{e} = (1, 0)$;
  $\widehat{\boldsymbol{\theta}}_{\boldsymbol{i}} = \arccos \dfrac{\vec{l} \cdot \vec{e}}{|\vec{l}||\vec{e}|}$;
  if $BC > 0$ **then**
    $\widehat{\boldsymbol{\theta}}_{\boldsymbol{i}} = 2\pi - \widehat{\boldsymbol{\theta}}_{\boldsymbol{i}}$;
  end
end

Inspired by the IoU representation [44], we propose Ray-IoU loss as L_ρ for ρ regression:

$$L_\rho = -\frac{1}{N} \sum_i \log \frac{min(\widehat{\rho}_i, \rho_i)}{max(\widehat{\rho}_i, \rho_i)}, \tag{9}$$

where $\widehat{\rho}_i$, ρ_i denote the label and prediction of ρ in the ith of N locations, respectively. Note that N is actually the number of valid text pixels. Considering that $\widehat{\rho}_i$ and ρ_i share the same start position which looks like rays so L_ρ is named as Ray-IoU loss. Compared with the L_2, L_1 and $Smooth$ L_1 losses, Ray-IoU loss could better normalize the various distances, which contributes to the multi-scale detection.

The θ ranges in [0, 2π]. However, the 0 and 2π are actually the same representation due to the cycle characteristic in polar space. When the angle difference increases from pi, in other words, the loss should decrease accordingly. Therefore, we design the sine angle loss in Eq. 10 to alleviate the confusion in above transition areas, where $\widehat{\theta}_i$ and θ_i denote the angle label and predicted angle in the ith of N locations, respectively.

$$L_\theta = \frac{1}{N} \sum_i \sin \frac{|\widehat{\theta}_i - \theta_i|}{2}. \tag{10}$$

During the whole training stage, the weights λ_1, λ_2 and λ_3 are set to 1, 1 and 1, respectively.

4. Experiments

In this section, we evaluate the DCLNet on three challenging public benchmarks: ICDAR 2017 MLT [26], ICDAR 2015 [14] and MSRA-TD500 [43]. Ablation study is also conducted to demonstrate the effectiveness of the proposed modules in DCLNet.

4.1. Datasets

SynthText [5] is a synthetic dataset which is frequently used to pretrain the network in existing methods. It contains about 800K synthetic images by artificially blending the natural images and texts rendered with random attributes such as colors, orientations and scales.

ICDAR 2017 MLT [26] is a multi-lingual scene text dataset, which includes 9 languages representing 6 different scripts. There are totally 7,200 training images, 1,800 validation images and 9,000 testing images, which are labeled using word level quadrangle with 4 vertices. Following prior arts, the training images and validation images are both used to finetune the model.

ICDAR 2015 [14] is a dataset for incidental scene text detection. There are totally 1000 images for training and 500 images for testing, where each text instance is also labeled by word level quadrangle with 4 vertices. The texts in this scene are usually multi-orientated which suffer from motion blur and low resolution.

MSRA-TD500 [43] is a line-level dataset with 300 training images and 200 test images of multi-oriented and long texts. Although the annotations are rotated rectangles, they could be viewed as a special type of quadrangle. The training set is relatively small, so we also include 400 images from HUST-TR400 [42] as training data according to previous works [46, 24, 23].

4.2. Implementation Details

The DCLNet backbone is ResNet50 [7] which has been pretrained on ImageNet [4]. It produces 4 stages feature maps denoted by C_2, C_3, C_4 and C_5. With the channel reduce module, their channels are reduced to 64, 128, 256 and 512, respectively. The feature aggregation module fuses the feature maps on different scales to deliver the final output feature map which has stride of 4 pixels with respect to the original input image. The output feature maps for text classification, distance regression and angle regression are 1, 4 and 4 channels, respectively. Finally, the entire network contains about 35.3M parameters.

The model is optimized using Adam [15] with batch-size 64 and the learning rate decreases under cosine schedule. The training stage contains two phases: Firstly, we use SynthText [5] to pretrain the model for 5 epochs. The learning rate decreases from 1×10^{-3} to 1×10^{-4}. Then, we finetune the model on ICDAR 2017 MLT with 150 epochs

with the same learning rate setting and only select the final epoch for evaluation. The training stages on ICDAR 2015 and MSRA-TD500 are both finetuned 100 epochs from the MLT model. During the training, the blurred texts labeled as DO NOT CARE are all ignored.

For the data augmentation, we set the height and with of training image in [640, 2560] randomly without keeping the original aspect ratio in all benchmarks. It is because the texts usually have various viewpoints with distorted shapes. Color jitters on brightness, contrast and saturation are also employed to improve the generalization ability. Finally, 640×640 patches are randomly cropped from the transformed images as training data.

In inference stage, the images are resized by setting the short side to 1280 and keep the original aspect ratio as single scale test. The constructed bounding boxes are filtered using the Locality-Aware NMS (Non Maximum Suppression) [46] with 0.2 threshold in all experiments.

4.3. Benchmark Comparisons

ICDAR 2017 MLT. We evaluate the proposed DCLNet and other state-of-the-art baselines on ICDAR 2017 MLT, which contains multiple languages texts with challenging distorted quadrilateral boundaries. Table 1 presents the performance comparisons where the best scores are highlighted in bold. DCLNet achieves 73.3% F-measure in the single scale test, which has outperformed many detection methods. Furthermore, the short side of image is resized to [640, 1920] with the same aspect ratio to perform multi-scale test. In this way, DCLNet finally achieves state-of-the-art 76.3% F-measure, with a 1.8% improvement with respect to prior arts. The comparison clearly demonstrates the effectiveness of DCLNet to handle arbitrary quadrilateral texts, with the tailored Ray-IoU loss and sine angle loss.

ICDAR 2015. We also evaluate the proposed DCLNet and other state-of-the-art baselines on ICDAR 2015. The viewpoints of texts in ICDAR 2015 are relatively stable and not distorted like ICDAR 2017 MLT. However, the main challenge comes from the motion blur and low resolution of texts in this incidental scene. Table 2 presents the performance comparisons where the best scores are highlighted in bold. In the single scale test, DCLNet achieves 88.7% F-measure and already outperforms all other detection methods even under multi-scale test. It indicates that DCLNet could effectively recognize the texts even with low quality imaging. With the proposed Ray-IoU loss and sine angle loss, the boundary of these challenging texts could also be accurately restored.

MSRA-TD500. We further evaluate DCLNet and other state-of-the-art baselines on MSRA-TD500. The main challenge is from the texts with multiple orientations and extreme long lengths. Considering that the texts are usually clear with relatively large scale, the short side of image in set to 880 in single scale test. Table 3 presents the performance comparisons where the best scores are highlighted in bold. DCLNet achieves the state-of-the-art 85.1 F-measure in the single scale test. It demonstrates the effectiveness of DCLNet with tailored Ray-IoU loss to address extreme long texts with large scale in some orientations.

Figure 5 presents some detected text boundaries in above three benchmarks for better visualization. The 1-3 rows

Method	Precision	Recall	F-measure
DR [10]	76.7	57.9	66.0
Border [40]	77.7	62.1	69.0
PSENet [35]	73.8	68.2	70.9
CLRS [24]	83.8	55.6	66.8
CLRS* [24]	74.3	70.6	72.4
LOMO [45]	78.8	60.6	68.5
LOMO* [45]	80.2	67.2	73.1
CRAFT [1]	80.6	68.2	73.9
SPCNet [38]	73.4	66.9	70.0
SPCNet* [38]	80.6	68.6	74.1
Two-stage [34]	78.3	63.4	70.1
Two-stage* [34]	81.8	68.5	74.5
GNNet [39]	79.6	70.1	74.5
DCLNet	81.0	66.9	73.3
DCLNet*	**81.9**	**71.4**	**76.3**

Table 1. The quantitative results of DCLNet and other baselines on the ICDAR 2017 MLT dataset. The best scores are highlighted in bold. "*" denotes multi-scale test.

Method	Precision	Recall	F-measure
SegLink [29]	73.1	76.8	75.0
WordSup [11]	79.3	77.0	78.2
EAST* [46]	83.3	78.3	80.7
DeepReg [9]	82.0	80.0	81.0
R2CNN [13]	85.6	79.7	82.5
TextSnake [23]	84.9	80.4	82.6
TextBoxes++* [16]	87.8	78.5	82.9
PixelLink [3]	85.5	82.0	83.7
RRD* [18]	88.0	80.0	83.8
FTSN [2]	88.6	80.0	84.1
CLRS* [24]	89.5	79.7	84.3
Two-stage [34]	89.2	82.3	85.6
SAE [33]	88.3	85.0	86.6
IncepText* [41]	89.4	84.3	86.8
CRAFT [1]	89.8	84.3	86.9
SPCNet [38]	88.7	85.8	87.2
PSENet [35]	89.3	85.2	87.2
LOMO* [45]	87.8	**87.6**	87.7
DCLNet	**90.3**	87.1	**88.7**

Table 2. The quantitative results of DCLNet and other baselines on the ICDAR 2015 dataset. The best scores are highlighted in bold. "*" denotes multi-scale test.

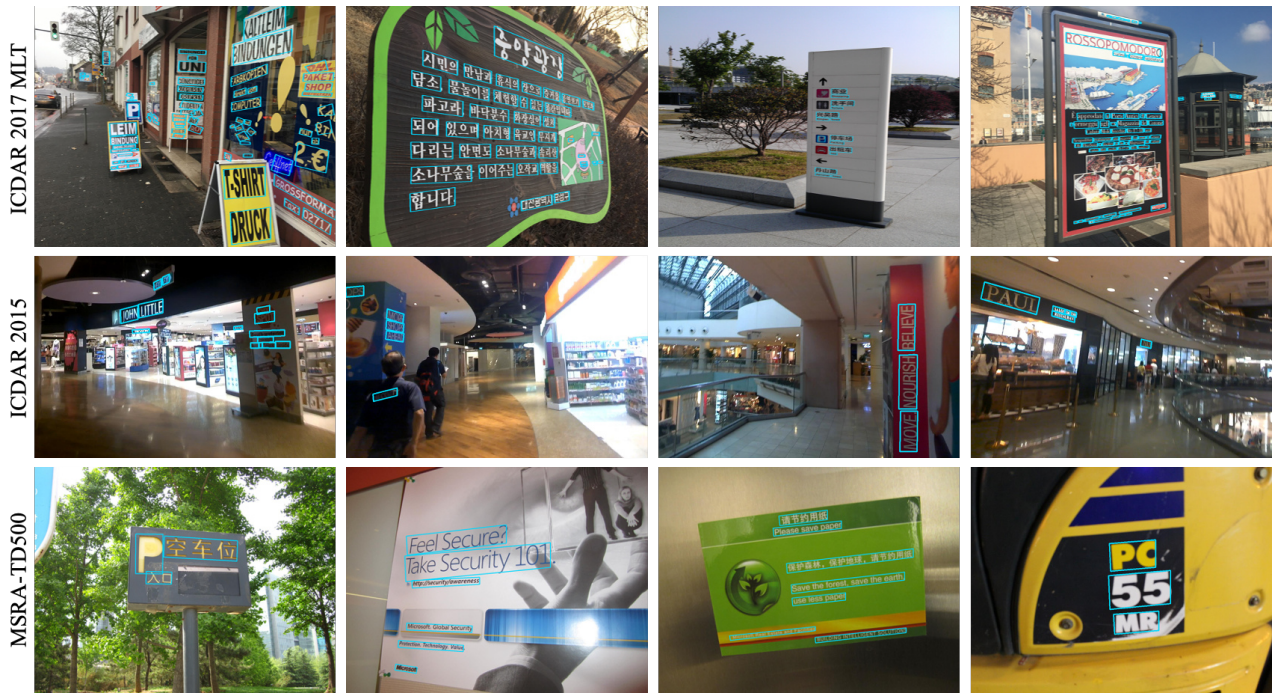

Figure 5. Visualization of detected texts in ICDAR 2017 MLT, ICDAR 2015 and MSRA-TD500.

Method	Precision	Recall	F-measure
DeepReg [9]	77.0	70.0	74.0
RRPN* [25]	82.0	69.0	75.0
EAST* [46]	87.3	67.4	76.1
SegLink [29]	86.0	70.0	77.0
PixelLink [3]	83.0	73.2	77.8
TextSnake [23]	83.2	73.9	78.3
CLRS [24]	87.6	76.2	81.5
FTSN [2]	87.6	77.1	82.0
CRAFT [1]	88.2	78.2	82.9
SAE [33]	84.2	81.7	82.9
IncepText [41]	87.5	79.0	83.0
RRD* [18]	88.0	80.0	83.8
PAN [36]	84.4	**83.8**	84.1
DCLNet	**90.2**	80.5	**85.1**

Table 3. The quantitative results of DCLNet and other baselines on the MSRA-TD500 dataset. The best scores are highlighted in bold. "*" denotes multi-scale test.

denote the detection results of ICDAR 2017 MLT, ICDAR 2015 and MSRA-TD500, respectively. It can be seen that the texts with different languages, orientations and scales are all precisely located. Moreover, the distorted quadrilateral texts, blurred texts with low resolution and long texts are also detected with high quality boundaries. It solidly demonstrates that the disentangled contour learning could better regress the independent sides without distributions from each other. In the polar coordinates, the proposed Ray-IoU loss and sine angle loss further improve the regression quality to produce more accurate detection results.

4.4. Ablation Study

Disentangled v.s. Entangled. Although the benchmark comparisons have demonstrated the effectiveness of disentangled learning, there are many other variables affect the fair comparison such as backbone architecture, data augmentation, training strategy. Therefore, we design ablation study to clearly compare the disentangled DCLNet and entangled baseline. Specifically, baseline regresses the offsets of eight vertices coordinates with the L_1 loss like prior arts. Figure 6 presents the regression loss curves of DCLNet and baseline on ICDAR 2017 MLT dataset without pretrain. Note that the regression of DCLNet contains ρ and θ two parts, which are added with 1:1 ratio as in training. It can be seen that the loss curve of DCLNet is smoother and converges faster compared with the baseline, which brings about 3.5% F-measure improvement. The above comparison proves that the learning process in DCLNet is much easier than that in entangled baseline, which benefits from the disentangled side representation of quadrilateral contour and meaningful parameterization in polar coordinates. Moreover, Figure 7 presents some detected examples of distorted quadrilateral texts, where the irregular boundaries are all precisely localized. It clearly shows the effectiveness of DCLNet to detect quadrilateral texts for the subsequent recognition applications.

Ray-IoU and sine angle losses. DCLNet clearly sepa-

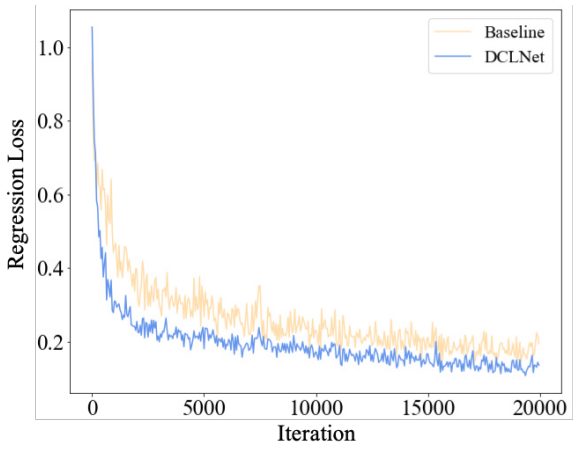

Figure 6. Regression losses comparison of entangled baseline and our DCLNet. The training of disentangled DCLNet is more stable and converges faster than baseline.

Figure 7. Detection results of distorted quadrilateral texts with DCLNet, where irregular boundaries are all precisely localized.

Ray-IoU Loss	Sine Angle Loss	F-measure
-	-	86.9
✓	-	88.3
-	✓	87.2
✓	✓	**88.7**

Table 4. The performances of DCLNet with different losses on ICDAR 2015 dataset. The best score is highlighted in bold.

rates the independent parameters of each disentangled side. Ray-IoU and sine angle losses are designed to further improve the learning ability for these line parameters. Table 4 presents the single scale F-measures on ICDAR 2015 with different losses. The replacements for these two losses are commonly used L_1 loss and cosine angle loss [46]. It can be seen that Ray-IoU loss significantly improves the performance. For texts with extreme short or long lengths in some orientations, Ray-IoU loss could better supervise the learning under various scales compared with other distance losses. Sine loss alleviates the angle confusion in junction

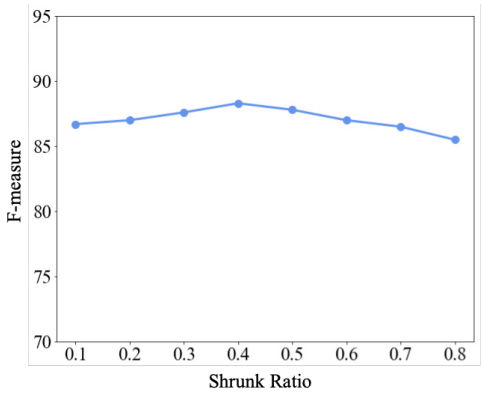

Figure 8. The F-measures with different shrink ratios on ICDAR 2015 dataset.

areas, which further promotes the performance to 88.7% F-measure. The above improvement clearly demonstrates the effectiveness of proposed Ray-IoU loss and sine angle loss for line equation parameters learning.

Text classification label. The regression is based on text region which is supervised by shrunk text mask with preset ratio. Shrunk mask could help avoid confusing learning in the edge areas. However, too large shrink ratio would also cause classification confusion on surrounding text region. Therefore, we perform the single scale experiments of shrink ratio on ICDAR 2015 dataset as shown in Figure 8. It can be seen that appropriate ratio could avoid confusion and improve the performance. However, large ratio would also introduce extra classification confusion thus the performance severely drops. Overall, the ratio in [0.3, 0.4] is beneficial for improving performance and 0.35 is selected as the default value in our experiments.

5. Conclusion

The main challenge of quadrilateral text detection comes from four distorted sides. Existing methods detect the contour by learning four vertices, which are actually dummy intersections entangled by neighbor sides. Each originally independent side would be influenced by two different vertices which inevitably disturb other sides. The above entangled vertices learning suppresses learning efficiency and detection performance. In this paper, we proposed DCLNet to clearly regress each individual side disentangled from whole quadrilateral contour. The side is parameterized by a linear equation disentangled in polar coordinates. With the tailored Ray-IoU loss and sine angle loss, DCLNet could better learn the representation of each disentangled side without being disturbed by others. DCLNet achieves state-of-the-art detection performances on three scene text benchmarks. Ablation study also proves the effectiveness of proposed disentangled learning framework.

References

[1] Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. Character region awareness for text detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9365–9374, 2019.

[2] Yuchen Dai, Zheng Huang, Yuting Gao, Youxuan Xu, Kai Chen, Jie Guo, and Weidong Qiu. Fused text segmentation networks for multi-oriented scene text detection. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3604–3609. IEEE, 2018.

[3] Dan Deng, Haifeng Liu, Xuelong Li, and Deng Cai. Pixellink: Detecting scene text via instance segmentation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[5] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2315–2324, 2016.

[6] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[8] Pan He, Weilin Huang, Tong He, Qile Zhu, Yu Qiao, and Xiaolin Li. Single shot text detector with regional attention. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3047–3055, 2017.

[9] Wenhao He, Xu-Yao Zhang, Fei Yin, and Cheng-Lin Liu. Deep direct regression for multi-oriented scene text detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 745–753, 2017.

[10] Wenhao He, Xu-Yao Zhang, Fei Yin, and Cheng-Lin Liu. Multi-oriented and multi-lingual scene text detection with direct regression. *IEEE Transactions on Image Processing*, 27(11):5406–5419, 2018.

[11] Han Hu, Chengquan Zhang, Yuxuan Luo, Yuzhuo Wang, Junyu Han, and Errui Ding. Wordsup: Exploiting word annotations for character based text detection. In *Proceedings of the IEEE international conference on computer vision*, pages 4940–4949, 2017.

[12] Lichao Huang, Yi Yang, Yafeng Deng, and Yinan Yu. Densebox: Unifying landmark localization with end to end object detection. *arXiv preprint arXiv:1509.04874*, 2015.

[13] Yingying Jiang, Xiangyu Zhu, Xiaobing Wang, Shuli Yang, Wei Li, Hua Wang, Pei Fu, and Zhenbo Luo. R2cnn: rotational region cnn for orientation robust scene text detection. *arXiv preprint arXiv:1706.09579*, 2017.

[14] Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, et al. Icdar 2015 competition on robust reading. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 1156–1160. IEEE, 2015.

[15] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[16] Minghui Liao, Baoguang Shi, and Xiang Bai. Textboxes++: A single-shot oriented scene text detector. *IEEE transactions on image processing*, 27(8):3676–3690, 2018.

[17] Minghui Liao, Zhaoyi Wan, Cong Yao, Kai Chen, and Xiang Bai. Real-time scene text detection with differentiable binarization. In *AAAI*, pages 11474–11481, 2020.

[18] Minghui Liao, Zhen Zhu, Baoguang Shi, Gui-song Xia, and Xiang Bai. Rotation-sensitive regression for oriented scene text detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5909–5918, 2018.

[19] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[20] Yuliang Liu and Lianwen Jin. Deep matching prior network: Toward tighter multi-oriented text detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1962–1969, 2017.

[21] Yuliang Liu, Sheng Zhang, Lianwen Jin, Lele Xie, Yaqiang Wu, and Zhepeng Wang. Omnidirectional scene text detection with sequential-free box discretization. *arXiv preprint arXiv:1906.02371*, 2019.

[22] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[23] Shangbang Long, Jiaqiang Ruan, Wenjie Zhang, Xin He, Wenhao Wu, and Cong Yao. Textsnake: A flexible representation for detecting text of arbitrary shapes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 20–36, 2018.

[24] Pengyuan Lyu, Cong Yao, Wenhao Wu, Shuicheng Yan, and Xiang Bai. Multi-oriented scene text detection via corner localization and region segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7553–7563, 2018.

[25] Jianqi Ma, Weiyuan Shao, Hao Ye, Li Wang, Hong Wang, Yingbin Zheng, and Xiangyang Xue. Arbitrary-oriented scene text detection via rotation proposals. *IEEE Transactions on Multimedia*, 20(11):3111–3122, 2018.

[26] Nibal Nayef, Fei Yin, Imen Bizid, Hyunsoo Choi, Yuan Feng, Dimosthenis Karatzas, Zhenbo Luo, Umapada Pal, Christophe Rigaud, Joseph Chazalon, et al. Icdar2017 robust reading challenge on multi-lingual scene text detection and script identification-rrc-mlt. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 1, pages 1454–1459. IEEE, 2017.

[27] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[29] Baoguang Shi, Xiang Bai, and Serge Belongie. Detecting oriented text in natural images by linking segments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2550–2558, 2017.

[30] Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 761–769, 2016.

[31] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[32] Zhi Tian, Weilin Huang, Tong He, Pan He, and Yu Qiao. Detecting text in natural image with connectionist text proposal network. In *European conference on computer vision*, pages 56–72. Springer, 2016.

[33] Zhuotao Tian, Michelle Shu, Pengyuan Lyu, Ruiyu Li, Chao Zhou, Xiaoyong Shen, and Jiaya Jia. Learning shape-aware embedding for scene text detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4234–4243, 2019.

[34] Siwei Wang, Yudong Liu, Zheqi He, Yongtao Wang, and Zhi Tang. A quadrilateral scene text detector with two-stage network architecture. *Pattern Recognition*, 102:107230, 2020.

[35] Wenhai Wang, Enze Xie, Xiang Li, Wenbo Hou, Tong Lu, Gang Yu, and Shuai Shao. Shape robust text detection with progressive scale expansion network. *arXiv preprint arXiv:1903.12473*, 2019.

[36] Wenhai Wang, Enze Xie, Xiaoge Song, Yuhang Zang, Wenjia Wang, Tong Lu, Gang Yu, and Chunhua Shen. Efficient and accurate arbitrary-shaped text detection with pixel aggregation network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8440–8449, 2019.

[37] Weijia Wu, Jici Xing, and Hong Zhou. Textcohesion: Detecting text for arbitrary shapes. *arXiv preprint arXiv:1904.12640*, 2019.

[38] Enze Xie, Yuhang Zang, Shuai Shao, Gang Yu, Cong Yao, and Guangyao Li. Scene text detection with supervised pyramid context network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9038–9045, 2019.

[39] Youjiang Xu, Jiaqi Duan, Zhanghui Kuang, Xiaoyu Yue, Hongbin Sun, Yue Guan, and Wayne Zhang. Geometry normalization networks for accurate scene text detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9137–9146, 2019.

[40] Chuhui Xue, Shijian Lu, and Fangneng Zhan. Accurate scene text detection through border semantics awareness and bootstrapping. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 355–372, 2018.

[41] Qiangpeng Yang, Mengli Cheng, Wenmeng Zhou, Yan Chen, Minghui Qiu, Wei Lin, and Wei Chu. Inceptext: A new inception-text module with deformable psroi pooling for multi-oriented scene text detection. *arXiv preprint arXiv:1805.01167*, 2018.

[42] Cong Yao, Xiang Bai, and Wenyu Liu. A unified framework for multioriented text detection and recognition. *IEEE Transactions on Image Processing*, 23(11):4737–4749, 2014.

[43] Cong Yao, Xiang Bai, Wenyu Liu, Yi Ma, and Zhuowen Tu. Detecting texts of arbitrary orientations in natural images. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1083–1090. IEEE, 2012.

[44] Jiahui Yu, Yuning Jiang, Zhangyang Wang, Zhimin Cao, and Thomas Huang. Unitbox: An advanced object detection network. In *Proceedings of the 24th ACM international conference on Multimedia*, pages 516–520. ACM, 2016.

[45] Chengquan Zhang, Borong Liang, Zuming Huang, Mengyi En, Junyu Han, Errui Ding, and Xinghao Ding. Look more than once: An accurate detector for text of arbitrary shapes. *arXiv preprint arXiv:1904.06535*, 2019.

[46] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. East: an efficient and accurate scene text detector. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 5551–5560, 2017.

[47] Yixing Zhu and Jun Du. Textmountain: Accurate scene text detection via instance segmentation. *arXiv preprint arXiv:1811.12786*, 2018.