Deep Matching Prior Network: Toward Tighter Multi-oriented Text Detection

Yuliang Liu, Lianwen Jin*
College of Electronic Information Engineering
South China University of Technology
*lianwen.jin@gmail.com

Abstract

Detecting incidental scene text is a challenging task because of multi-orientation, perspective distortion, and variation of text size, color and scale. Retrospective research has only focused on using rectangular bounding box or horizontal sliding window to localize text, which may result in redundant background noise, unnecessary overlap or even information loss. To address these issues, we propose a new Convolutional Neural Networks (CNNs) based method, named Deep Matching Prior Network (DMPNet), to detect text with tighter quadrangle. First, we use quadrilateral sliding windows in several specific intermediate convolutional layers to roughly recall the text with higher overlapping area and then a shared Monte-Carlo method is proposed for fast and accurate computing of the polygonal areas. After that, we designed a sequential protocol for relative regression which can exactly predict text with compact quadrangle. Moreover, a auxiliary smooth Ln loss is also proposed for further regressing the position of text, which has better overall performance than L2 loss and smooth L1 loss in terms of robustness and stability. The effectiveness of our approach is evaluated on a public word-level, multi-oriented scene text database, ICDAR 2015 Robust Reading Competition Challenge 4 "Incidental scene text localization". The performance of our method is evaluated by using F-measure and found to be 70.64%, outperforming the existing state-of-the-art method with F-measure 63.76%.

1. Introduction

Scene text detection is an important prerequisite [32, 31, 37, 1, 34] for many content-based applications, e.g., multilingual translation, blind navigation and automotive assistance. Especially, the recognition stage always stands in need of localizing scene text in advance, thus it is a significant requirement for detecting methods that can tightly and robustly localize scene text.

Camera captured scene text are often found with low-quality; these texts may have multiple orientations, perspective distortions, and variation of text size, color or scale [40], which makes it a very challenging task [39]. In the past few years, various existing methods have successfully been used for detecting horizontal or near-horizontal texts [2, 4, 23, 11, 10]. However, due to the horizontal rectangular constraints, multi-oriented text are restrictive to be recalled in practice, *e.g.* low accuracies reported in ICDAR 2015 Competition Challenge 4 "Incidental scene text localization" [14].

Recently, numerous techniques [35, 36, 13, 39] have been devised for multi-oriented text detection; these methods used rotated rectangle to localize oriented text. However, Ye and Doermann [34] indicated that because of characters distortion, the boundary of text may lose rectangular shape, and the rectangular constraints may result in redun-

(a) Rectangular bounding box cause unnecessary overlap.

(c) Marginal text can not be exactly localized with rectangle.

(d) Rectangular bounding box brings redundant noise.

Figure 1. Comparison of quadrilateral bounding box and rectangular bounding box for localizing texts.

dant background noise, unnecessary overlap or even information loss when detecting distorted incidental scene text as shown in Figure 1. It can be visualized from the Figure that the rectangle based methods must face three kinds of circumstances: i) redundant information may reduce the reliability of detected confidence [18] and make subsequent recognition harder [40]; ii) marginal text may not be localized completely; iii) when using non-maximum suppression [21], unnecessary overlap may eliminate true prediction.

To address these issues, in this paper, we proposed a new Convolutional Neural Networks (CNNs) based method, named Deep Matching Prior Network (DMPNet), toward tighter text detection. To the best of our knowledge, this is the first attempt to detect text with quadrangle. Basically, our method consists of two steps: roughly recall text and finely adjust the predicted bounding box. First, based on the priori knowledge of textual intrinsic shape, we design different kinds of quadrilateral sliding windows in specific intermediate convolutional layers to roughly recall text by comparing the overlapping area with a predefined threshold. During this rough procedure, because numerous polygonal overlapping areas between the sliding window (SW) and ground truth (GT) need to be computed, we design a shared Monte-Carlo method to solve this issue, which is qualitatively proved more accurate than the previous computational method [30]. After roughly recalling text, those SWs with higher overlapping area would be finely adjusted for better localizing; different from existing methods [2, 4, 23, 11, 10, 35, 36, 39] that predict text with rectangle, our method can use quadrangle for tighter localizing scene text, which owe to the sequential protocol we purposed and the relative regression we used. Moreover, a new smooth Ln loss is also proposed for further regressing the position of text, which has better overall performance than $L2$ loss and smooth $L1$ loss in terms of robustness and stability. Experiments on the public word-level and multi-oriented dataset, ICDAR 2015 Robust Reading Competition Challenge 4 "Incidental scene text localization", demonstrate that our method outperforms previous state-of-the-art methods [33] in terms of F-measure.

We summarize our contributions as follow:

- We are the first to put forward prior quadrilateral sliding window, which significantly improve the recall rate.
- We proposed sequential protocol for uniquely determining the order of 4 points in arbitrary plane convex quadrangle, which enable our method for using relative regression to predict quadrilateral bounding box.
- The proposed shared Monte-Carlo computational method can fast and accurately compute the polygonal overlapping area.
- The proposed smooth Ln loss has better overall performance than $L2$ loss and smooth $L1$ loss in terms of robustness and stability.
- Our approach shows state-of-the-art performance in detecting incidental scene text.

2. Related work

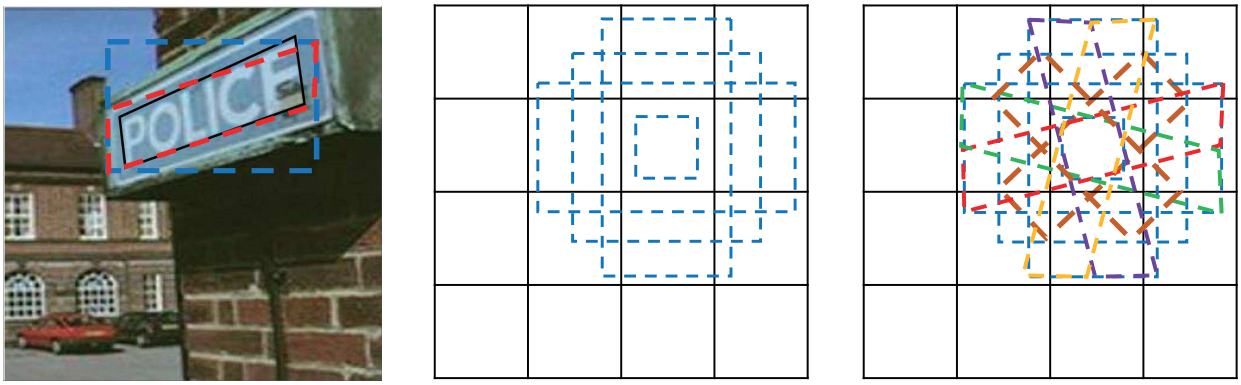
Reading text in the wild have been extensively studied in recent years because scene text conveys numerous valuable information that can be used on many intelligent applications, *e.g*. autonomous vehicles and blind navigation. Unlike generic objects, scene text has unconstrained lengths, shape and especially perspective distortions, which make text detection hard to simply adopt techniques from other domains. Therefore, the mainstream of text detection methods always focused on the structure of individual characters and the relationships between characters [40], *e.g*. connected component based methods [38, 27, 22]. These methods often use stroke width transform (SWT) [9] or maximally stable extremal region (MSER) [20, 24] to first extract character candidates, and using a series of subsequence steps to eliminate non-text noise for exactly connecting the candidates. Although accurate, such methods are somewhat limited to preserve various true characters in practice [3].

Another mainstream method is based on sliding window [2, 15, 8, 17], which shifts a window in each position with multiple scales from an image to detect text. Although this method can effectively recall text, the classification of the locations can be sensitive to false positives because the sliding windows often carry various background noise.

Recently, Convolutional Neural Networks [28, 6, 26, 19, 25] have been proved powerful enough to suppress false positives, which enlightened researchers in the area of scene text detection; in [10], Huang *et al*. integrated MSER and CNN to significantly enhance performance over conventional methods; Zhang *et al*. utilized Fully Convolutional Network [39] to efficiently generate a pixel-wise text/non-text salient map, which achieve state-of-the-art performance on public datasets. It is worth mentioning that the common ground of these successful methods is to utilized textual intrinsic information for training the CNN. Inspired by this promising idea, instead of using constrained rectangle, we designed numerous quadrilateral sliding windows based on the textual intrinsic shape, which significantly improves recall rate in practice.

3. Proposed methodology

This section presents details of the Deep Matching Prior Network (DMPNet). It includes the key contributions that make our method reliable and accurate for text localization: firstly, roughly recalling text with quadrilateral sliding window; then, using a shared Monte-Carlo method for fast



(a) Comparison of recalling scene text. (b) Horizontal sliding windows. (c) Proposed quadrilateral sliding windows.

Figure 2. Comparison between horizontal sliding window and quadrilateral sliding window. (a): Black bounding box represents ground truth; red represents our method. Blue represents horizontal sliding window. It can be visualized that quadrilateral window can easier recall text than rectangular window with higher intersection over union (IoU). (b): Horizontal sliding windows used in [19]. (c): Proposed quadrilateral sliding windows. Different quadrilateral sliding window can be distinguished with different color.

and accurate computing of polygonal areas; finely localizing text with quadrangle and design a Smooth Ln loss for moderately adjusting the predicted bounding box.

3.1. Roughly recall text with quadrilateral sliding window

Previous approaches [19, 26] have successfully adopted sliding windows in the intermediate convolutional layers to roughly recall text. Although the methods [26] can accurately learn region proposal based on the sliding windows, these approaches have been too slow for real-time or near real-time applications. To raise the speed, Liu [19] simply evaluate a small set of prior windows of different aspect ratios at each location in several feature maps with different scales, which can successfully detect both small and big objects. However, the horizontal sliding windows are often hard to recall multi-oriented scene text in our practice. Inspired by the recent successful methods [10, 39] that integrated the textual feature and CNN, we put forward numerous quadrilateral sliding windows based on the textual intrinsic shape to roughly recall text.

During this rough procedure, an overlapping threshold was used to judge whether the sliding window is positive or negative. If a sliding window is positive, it would be used to finely localize the text. Basically, a small threshold may bring a lot of background noise, reducing the precision, while a large threshold may make text harder to be recalled. But if we use quadrilateral sliding window, the overlapping area between sliding window and ground truth can be larger enough to reach a higher threshold, which are beneficial to improve both the recall rate and the precision as shown in Figure 2. As the figure presents, we reserve the horizontal sliding windows, simultaneously designing several quadrangles inside them based on the prior knowledge of textual intrinsic shape: a) two rectangles with 45 degrees are added inside the square; b) two long parallelograms are added inside the long rectangle. c) two tall parallelograms are added inside the tall rectangle.

With these flexible sliding windows, the rough bounding boxes become more accurate and thus the subsequence finely procedure can be easier to localize text tightly. In addition, because of less background noise, the confidence of these quadrilateral sliding windows can be more reliable in practice, which can be used to eliminate false positives.

3.1.1 Shared Monte-Carlo method

As mentioned earlier, for each ground truth, we need to compute its overlapping area with every quadrilateral sliding window. However, the previous method [30] can only compute rectangular area with unsatisfactory computational accuracy, thus we proposed a shared Monte-Carlo method that has both high speed and accuracy properties when computing the polygonal area. Our method consists of two steps.

a) First, we uniformly sample 10,000 points in circumscribed rectangle of the ground truth. The area of ground truth (S_{GT}) can be computed by calculating the ratio of overlapping points in total points multiplied by the area of circumscribed rectangle. In this step, all points inside the ground truth would be reserved for sharing computation.

b) Second, if the circumscribed rectangle of each sliding window and the circumscribed rectangle of each ground truth do not have a intersection, the overlapping area is considered zero and we do not need to further compute. If the overlapping area is not zero, we use the same sampling strategy to compute the area of sliding window (S_{SW}) and then calculating how many the reserved points from the first step inside the sliding window. The ratio of inside points multiplies the area of the circumscribed rectangle is

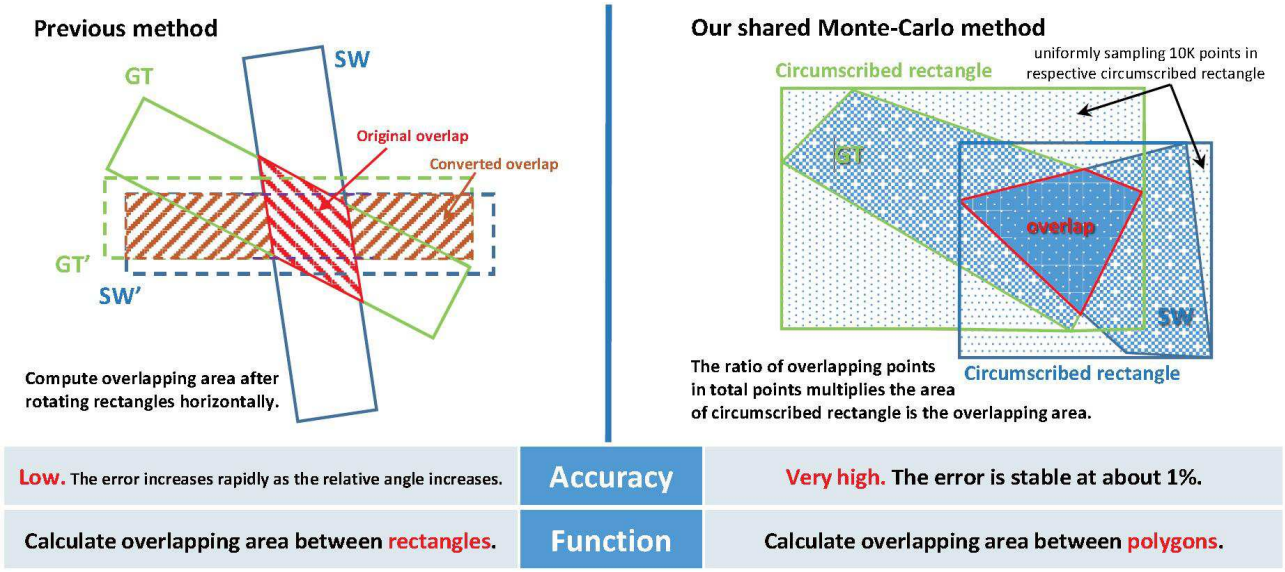


Figure 3. Comparison between previous method and our method in computing overlapping area.

the overlapping area. Specially, this step is suitable for using GPU parallelization, because we can use each thread to be responsible for calculating each sliding window with the specified ground truth, and thus we can handle thousands of sliding windows in a short time.

Note that we use a method proposed in [12] to judge whether a point is inside a polygon, and this method is also known as the crossing number algorithm or the even-odd rule algorithm [5]. The comparison between previous method and our algorithm is shown in Figure 3, our method shows satisfactory performance for computing polygonal area in practice.

3.2. Finely localize text with quadrangle

The fine procedure focuses on using those sliding windows with higher overlapping area to tightly localize text. Unlike horizontal rectangle that can be determined by two diagonal points, we need to predict the coordinates of four points to localize a quadrangle. However, simply using the 4 points to shape a quadrangle is prone to be self-contradictory, because the subjective annotation may make the network ambiguous to decide which is the first point. Therefore, before training, it is essential to order 4 points in advance.

Sequential protocol of coordinates. The propose protocol can be used to determine the sequence of four points in the plane convex quadrangle, which contains four steps as shown in Figure 4. First, we determine the first point with minimum value x. If two points simultaneously have the minimum x, then we choose the point with smaller value y as the first point. Second, we connect the first point to the other three points, and the third point can be found

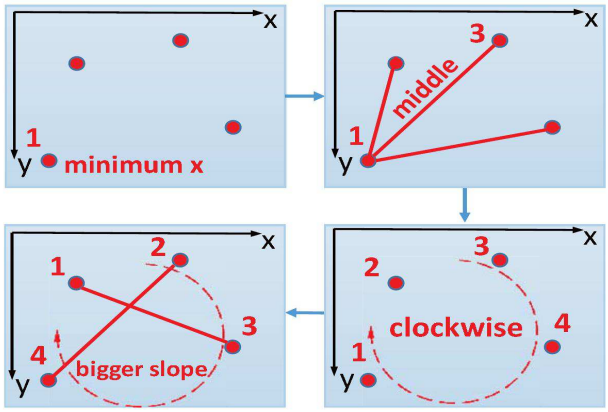


Figure 4. Procedure of uniquely determining the sequence of four points from a plane convex quadrangle.

from the line with middle slope. The second and the fourth points are in the opposite side (defined "bigger" side and "smaller" side) of the middle line. Here, we assume middle line $L_m : ax + by + c = 0$, and we define an undetermined point $P(x_p, y_p)$. If $L_m(P) > 0$, we assume P is in the "bigger" side. If $L_m(P) < 0$, P is assumed in the "smaller" side. Based on this assumption, the point in the "bigger" side would be assigned as second point, and the last point would be regarded as the fourth point. The last step is to compare the slopes between two diagonals ($line_{13}$ and $line_{24}$). From the line with bigger slope, we choose the point with smaller x as the new first point. Specially, if the bigger slope is infinite, the point that has smaller y would be chosen as the first point. Similarly, we find out the third point, and then the second and fourth point can be determined again. After finishing these four steps, the final

sequence of the four points from a given convex quadrangle can be uniquely determined.

Based on the sequential protocol, DMPNet can clearly learn and regress the coordinate of each point by computing the relative position to the central point. Different from [26] which regress two coordinates and two lengths for a rectangular prediction, our regressive method predicts two coordinates and eight lengths for a quadrilateral detection. For each ground truth, the coordinates of four points would be reformatted to $(x, y, w_1, h_1, w_2, h_2, w_3, h_3, w_4, h_4)$, where x, y are the central coordinate of the minimum circumscribed horizontal rectangle, and w_i, h_i are the relative position of the i-th point ($i = \{1, 2, 3, 4\}$) to the central point. As Figure 5 shows, the coordinates of four points $(x_1,y_1,x_2,y_2,x_3,y_3,x_4,y_4)$= $(x+w_1,y+h_1,x+w_2,y+h_2,x+w_3,y+h_3,x+w_4,y+h_4)$. Note that w_i and h_i can be negative. Actually, eight coordinates are enough to determine the position of a quadrangle, and the reason why we use ten coordinates is because we can avoid regressing 8 coordinates, which do not contain relative information and it is more difficult to learn in practice [6]. Inspired by [26], we also use Lreg(p_i;p_i^*) = R(p_i-p_i^*) for multi-task loss, where R is our proposed loss function (smooth Ln) that would be described in section 3.4. $p^* = (p_x^*, p_y^*, p_{w1}^*, p_{h1}^*, p_{w2}^*, p_{h2}^*, p_{w3}^*, p_{h3}^*, p_{w4}^*, p_{h4}^*)$ represents the ten parameterized coordinates of the predicted bounding box (sliding window), and $p = (p_x, p_y, p_{w1}, p_{h1}, p_{w2}, p_{h2}, p_{w3}, p_{h3}, p_{w4}, p_{h4})$ represents the ground truth.

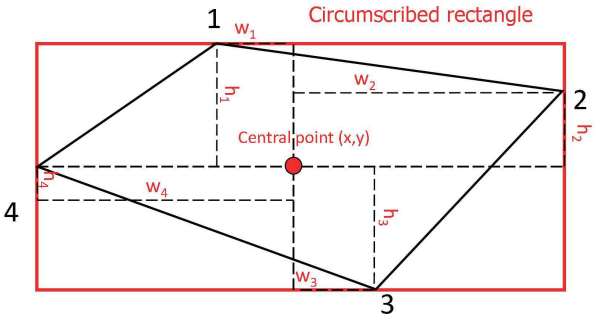


Figure 5. The position of each point of quadrangle can be calculated by central point and the relative lengths.

From the given coordinates, we can calculate the minimum x (x_{min}) and maximum x (x_{max}) of the circumscribed rectangle, and the width of circumscribed horizontal rectangle $w_{chr} = x_{max} - x_{min}$. Similarly, we can get the height $h_{chr} = y_{max} - y_{min}$.

We adopt the parameterizations of the 10 coordinates as following: $d_x = \frac{p_x^* - p_x}{w_{chr}}$, $d_y = \frac{p_y^* - p_y}{h_{chr}}$, $d_{w_1} = \frac{p_{w_1}^* - p_{w_1}}{w_{chr}}$, $d_{h_1} = \frac{p_{h_1}^* - p_{h_1}}{h_{chr}}$, $d_{w_2} = \frac{p_{w_2}^* - p_{w_2}}{w_{chr}}$, $d_{h_2} = \frac{p_{h_2}^* - p_{h_2}}{h_{chr}}$, $d_{w_3} = \frac{p_{w_3}^* - p_{w_3}}{w_{chr}}$, $d_{h_3} = \frac{p_{h_3}^* - p_{h_3}}{h_{chr}}$, $d_{w_4} = \frac{p_{w_4}^* - p_{w_4}}{w_{chr}}$, $d_{h_4} = \frac{p_{h_4}^* - p_{h_4}}{h_{chr}}$. This can be thought of as fine regression from an quadrilateral sliding window to a nearby ground-truth box.

3.3. Smooth Ln loss

Different from [19, 26], our approach uses a proposed smooth Ln loss instead of smooth L_1 loss to further localize scene text. Smooth L_1 loss is less sensitive to outliers than the L_2 loss used in R-CNN [7], however, this loss is not stable enough for adjustment of a data, which means the regression line may jump a large amount for small adjustment or just a little modification was used for big adjustment. As for proposed smooth Ln loss, the regressive parameters are continuous functions of the data, which means for any small adjustment of a data point, the regression line will always move only slightly, improving the precision in localizing small text. For bigger adjustment, the regression can always move to a moderate step based on smooth Ln loss, which can accelerate the converse of training procedure in practice. As mentioned in section 3.2, the recursive loss, $Lreg$, is defined over a tuple of true bounding-box regression targets p^* and a predicted tuple p for class text. The Smooth L_1 loss proposed in [6] is given by:

$$Lreg(p; p^*) = \sum_{i \in S} smooth_{L_1}(p_i, p^*), \tag{1}$$

in which,

$$smooth_{L_1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise.} \end{cases} \tag{2}$$

The x in the function represents the error between predicted value and ground truth ($x = w \cdot (p - p_*)$). The deviation function of $smooth_{L_1}$ is:

$$deviation_{L_1}(x) = \begin{cases} x & \text{if } |x| < 1 \\ sign(x) & \text{otherwise.} \end{cases} \tag{3}$$

As equation 3 shows, the deviation function is a piecewise function while the smooth Ln loss is a continuous derivable function. The proposed Smooth Ln loss is given by:

$$Lreg(p; p^*) = \sum_{i \in S} smooth_{Ln}(p_i, p^*), \tag{4}$$

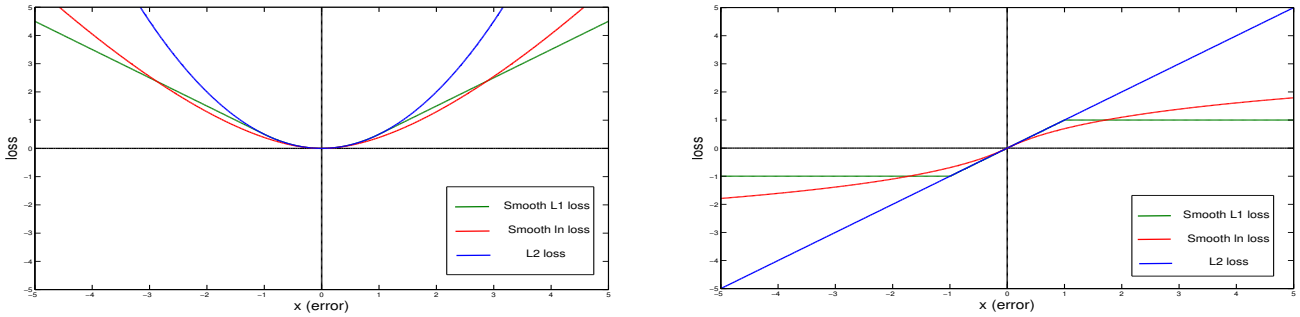
in which,

$$smooth_{Ln}(x) = (|d| + 1)ln(|d| + 1) - |d|, \tag{5}$$

and the deviation function of $smooth_{Ln}$ is:

$$deviation_{Ln}(x) = sign(x) \cdot ln(sign(x) \cdot x + 1). \tag{6}$$

Equation 5 and equation 6 are both continuous function with a single equation. For equation 6, it is easy to prove



(a) forward loss functions. (b) backward deviation functions.

Figure 6. Visualization of differences among three loss functions ($L2$, smooth $L1$ and smooth Ln). Here, the $L2$ function uses the same coefficient 0.5 with smooth $L1$ loss.

property	$L2$ loss	smooth L_1 loss	smooth Ln loss
Robustness	Worst	Best	Good
Stability	Good	Worst	Best

Table 1. Different properties of different loss functions. **Robustness** represents the ability of resistance to outliers in the data and **stability** represents the capability of adjusting regressive step.

$|x| \geq |deviation_{Ln}(x)|$, which means the smooth Ln loss is also less sensitive to outliers than the L_2 loss used in R-CNN [7]. A intuitive representation of the differences among three loss functions is shown in Figure 6. The comparisons of properties in terms of robustness and stability are summarized in Table 1. The results demonstrate that the smooth Ln loss promises better text localization and relatively tighter bounding boxes around the texts.

4. Experiments

Our testing environment is a desktop running Ubuntu 14.04 64bit version with TitanX. In this section, we quantitatively evaluate our method on the public dataset: ICDAR 2015 Competition Challenge 4: "Incidental Scene Text" [14], and as far as we know, this is the only one dataset in which texts are both word-level and multi-oriented. All results of our methods are evaluated from its online evaluation system, which would calculate the recall rate, precision and F-measure to rank the submitted methods. The general criteria of these three index can be explained below:

- **Recall rate** evaluates the ability of finding text.
- **Precision** evaluates the reliability of predicted bounding box.
- **F-measure** is the harmonic mean (Hmean) of recall rate and precision, which is always used for ranking the methods.

Particularly, we simply use official 1000 training images as our training set without any extra data augmentation, but we have modified some rectangular labels to quadrilateral labels for adapting to our method.

Dataset - ICDAR 2015 Competition Challenge 4 "Incidental Scene Text" [14]. Different from the previous ICDAR competition, in which the text are well-captured, horizontal, and typically centered in images. The datasets includes 1000 training images and 500 testing incidental scene image in where text may appear in any orientation and any location with small size or low resolution and the annotations of all bounding boxes are marked at the word level.

Baseline network. The main structure of DMPNet is based on the VGG-16 model [28], Similar to Single Shot Detector [19], we use the same intermediate convolutional layers to apply quadrilateral sliding windows. All input images would be resized to a 800x800 for preserving tiny texts.

Experimental results. For comprehensively evaluating our algorithm, we collect and list the competition results [14] in Table 2. The previous best method of this dataset, proposed by Yao *et al*. [33], achieved a F measure of 63.76% while our approach obtains 70.64%. The precision of these two methods are comparable but the recall rate of our method has greatly increased, which is mainly due to the quadrilateral sliding windows described in section 3.1.

Figure 7 shows several detected results taken from the test set of ICDAR 2015 challenge 4. DMPNet can robustly localize all kinds of scene text with less background noise. However, due to the complexity of incidental scene, some false detections still exist, and our method may fail to recall some inconspicuous text as shown in the last column of Figure 7.

5. Conclusion and future work

In this paper, we have proposed an CNN based method, named Deep Matching Prior Network (DMPNet), that can effectively reduce the background interference. The DMPNet is the first attempt to adopt quadrilateral sliding win-

Table 2. Evaluation on the ICDAR 2015 competition on robust reading challenge 4 "Incidental Scene Text" localization.

Algorithm	Recall (%)	Precision (%)	Hmean (%)
Baseline (SSD-VGGNet)	25.48	63.25	36.326
Proposed DMPNet	**68.22**	73.23	**70.64**
Megvii-Image++ [33]	56.96	72.40	63.76
CTPN [29]	51.56	74.22	60.85
MCLAB_FCN [14]	43.09	70.81	53.58
StardVision-2 [14]	36.74	**77.46**	49.84
StardVision-1 [14]	46.27	53.39	49.57
CASIA_USTB-Cascaded [14]	39.53	61.68	48.18
NJU_Text [14]	35.82	72.73	48.00
AJOU [16]	46.94	47.26	47.10
HUST_MCLAB [14]	37.79	44.00	40.66
Deep2Text-MO [36]	32.11	49.59	38.98
CNN Proposal [14]	34.42	34.71	34.57
TextCatcher-2 [14]	34.81	24.91	29.04

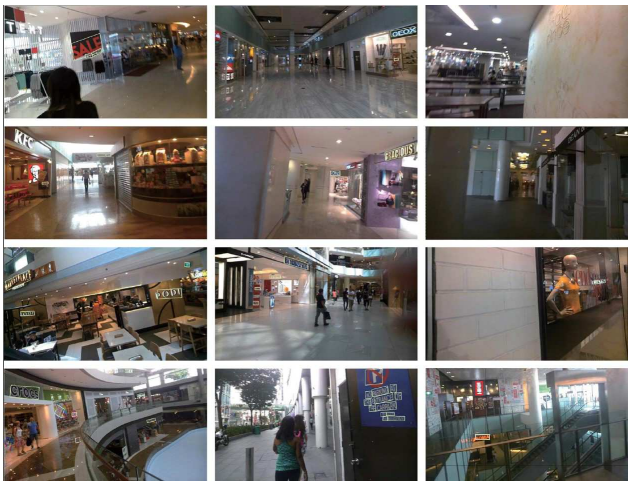


Figure 7. Experimental results of samples on ICDAR 2015 Challenge 4, including multi-scale and multi-language word-level text. Our method can tightly localize text with less background information as shown in the first two columns. Top three images from last column are the failure recalling cases of the proposed method. Specially, some labels are missed in some images, which may reduce our accuracy as the red bounding box listed in the fourth image of the last column.

dows, which are designed based on the priori knowledge of textual intrinsic shape, to roughly recall text. And we use a proposed sequential protocol and a relative regressive method to finely localize text without self-contradictory. Due to the requirement of computing numerous polygonal overlapping area in the rough procedure, we proposed a shared Monte-Carlo method for fast and accurate calculation. In addition, a new smooth Ln loss is used for further adjusting the prediction, which shows better overall performance than $L2$ loss and smooth $L1$ loss in terms of robustness and stability. Experiments on the well-known ICDAR 2015 robust reading challenge 4 dataset demonstrate that DMPNet can achieve state-of-the-art performance in detecting incidental scene text. In the following, we discuss an issue related to our approach and briefly describe our future work.

Ground truth of the text. Texts in camera captured images are always with perspective distortion. However rectangular constraints of labeling data may bring a lot of background noise, and it may lose information for not containing all texts when labeling marginal text. As far as we know, ICDAR 2015 Challenge 4 is the first dataset to use quadrilateral labeling, and our method prove the effectiveness of utilizing quadrilateral labeling. Thus, quadrilateral labeling for scene text may be more reasonable.

Future Work. The high recall rate of the DMPNet mainly depends on numerous prior-designed quadrilateral sliding windows. Although our method have been proved effective, the man-made shape of sliding window may not be the optimal designs. In future, we will explore using shape-adaptive sliding windows toward tighter scene text detection.

Acknowledgement

This research is supported in part by NSFC (Grant No.: 61472144), the National Key Research & Development Plan of China (No. 2016YFB1001405), GDSTP (Grant No.: 2015B010101004, 2015B010130003, 2015B010131004) , GZSTP(no. 201607010227).

References

[1] A. Bissacco, M. Cummins, Y. Netzer, and H. Neven. Photoocr: Reading text in uncontrolled conditions. In *IEEE International Conference on Computer Vision*, pages 785–792, 2013. 1

[2] X. Chen and A. L. Yuille. Detecting and reading text in natural scenes. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 366–373, 2004. 1, 2

[3] H. Cho, M. Sung, and B. Jun. Canny text detector: Fast and robust scene text localization algorithm. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3566–3573, 2016. 2

[4] B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in natural scenes with stroke width transform. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2963–2970. IEEE, 2010. 1, 2

[5] M. Galetzka and P. O. Glauner. A correct even-odd algorithm for the point-in-polygon (pip) problem for complex polygons. *CVPR*, 2012. 4

[6] R. Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1440–1448, 2015. 2, 5

[7] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. 5, 6

[8] S. M. Hanif and L. Prevost. Text detection and localization in complex scene images using constrained adaboost algo-

rithm. In *2009 10th International Conference on Document Analysis and Recognition*, pages 1–5. IEEE, 2009. 2

[9] W. Huang, Z. Lin, J. Yang, and J. Wang. Text localization in natural images using stroke feature transform and text covariance descriptors. In *International Conference on Computer Vision*, pages 1241–1248, 2013. 2

[10] W. Huang, Y. Qiao, and X. Tang. Robust scene text detection with convolution neural network induced mser trees. In *ECCV*, pages 497–511, 2014. 1, 2, 3

[11] M. Jaderberg, A. Vedaldi, and A. Zisserman. Deep features for text spotting. In *European conference on computer vision*, pages 512–528. Springer, 2014. 1, 2

[12] H. Kai and A. Agathos. The point in polygon problem for arbitrary polygons. *Computational Geometry*, 20(3):131–144, 2001. 4

[13] L. Kang, Y. Li, and D. Doermann. Orientation robust text line detection in natural images. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 4034–4041. IEEE, 2014. 1

[14] D. Karatzas, S. Lu, F. Shafait, S. Uchida, E. Valveny, L. Gomezbigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, and M. Iwamura. Icdar 2015 competition on robust reading. In *International Conference on Document Analysis and Recognition*, 2015. 1, 6, 7

[15] K. I. Kim, K. Jung, and H. K. Jin. Texture-based approach for text detection in images using support vector machines and continuously adaptive mean shift algorithm. *Pattern Analysis & Machine Intelligence IEEE Transactions on*, 25(12):1631–1639, 2003. 2

[16] H. I. Koo and D. H. Kim. Scene text detection via connected component clustering and nontext filtering. *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society*, 22(6):2296–2305, 2013. 7

[17] J.-J. Lee, P.-H. Lee, S.-W. Lee, A. L. Yuille, and C. Koch. Adaboost for text detection in natural scene. In *ICDAR*, pages 429–434, 2011. 2

[18] M. Li and I. K. Sethi. Confidence-based active learning. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 28(8):1251–61, 2006. 2

[19] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed. Ssd: Single shot multibox detector. *arXiv preprint arXiv:1512.02325*, 2015. 2, 3, 5, 6

[20] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image & Vision Computing*, 22(10):761–767, 2004. 2

[21] A. Neubeck and L. V. Gool. Efficient non-maximum suppression. In *International Conference on Pattern Recognition*, pages 850–855, 2006. 2

[22] L. Neumann and J. Matas. Real-time scene text localization and recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3538–3545, 2012. 2

[23] L. Neumann and J. Matas. Scene text localization and recognition with oriented stroke detection. In *IEEE International Conference on Computer Vision*, pages 97–104, 2013. 1, 2

[24] D. Nistr and H. Stewnius. Linear time maximally stable extremal regions. In *Computer Vision - ECCV 2008, European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings*, pages 183–196, 2008. 2

[25] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. *arXiv preprint arXiv:1506.02640*, 2015. 2

[26] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, pages 1–1, 2016. 2, 3, 5

[27] C. Shi, C. Wang, B. Xiao, Y. Zhang, S. Gao, and Z. Zhang. Scene text recognition using part-based tree-structured character detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2961–2968, 2013. 2

[28] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2, 6

[29] Z. Tian, W. Huang, T. He, P. He, and Y. Qiao. *Detecting Text in Natural Image with Connectionist Text Proposal Network*. Springer International Publishing, 2016. 7

[30] Z. Tu, Y. Ma, W. Liu, X. Bai, and C. Yao. Detecting texts of arbitrary orientations in natural images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1083–1090, 2012. 2, 3

[31] J. J. Weinman, Z. Butler, D. Knoll, and J. Feild. Toward integrated scene text reading. *IEEE Transactions on Software Engineering*, 36(2):375–87, 2014. 1

[32] J. J. Weinman, E. Learned-Miller, and A. R. Hanson. Scene text recognition using similarity and a lexicon with sparse belief propagation. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 31(10):1733–46, 2009. 1

[33] C. Yao, J. Wu, X. Zhou, C. Zhang, S. Zhou, Z. Cao, and Q. Yin. Incidental scene text understanding: Recent progresses on icdar 2015 robust reading competition challenge 4. *PAMI*, 2015. 2, 6, 7

[34] Q. Ye and D. Doermann. Text detection and recognition in imagery: A survey. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 37(7):1480–1500, 2015. 1

[35] C. Yi and Y. Tian. Text string detection from natural scenes by structure-based partition and grouping. *IEEE Transactions on Image Processing*, 20(9):2594–605, 2011. 1, 2

[36] X. C. Yin, W. Y. Pei, J. Zhang, and H. W. Hao. Multi-orientation scene text detection with adaptive clustering. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 37(9):1930–7, 2015. 1, 2, 7

[37] X. C. Yin, X. Yin, K. Huang, and H. W. Hao. Robust text detection in natural scene images. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 36(5):970–83, 2014. 1

[38] A. Zamberletti, L. Noce, and I. Gallo. Text localization based on fast feature pyramids and multi-resolution maximally stable extremal regions. In *Asian Conference on Computer Vision*, pages 91–105. Springer, 2014. 2

[39] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai. Multi-oriented text detection with fully convolutional networks. *arXiv preprint arXiv:1604.04018*, 2016. 1, 2, 3

[40] Y. Zhu, C. Yao, and X. Bai. Scene text detection and recognition: recent advances and future trends. *Frontiers of Computer Science*, 10(1):19–36, 2016. 1, 2