Feature Enhancement with Text-Specific Region Contrast for Scene Text Detection

Xurui Sun[1,2], Jiahao Lyu[1,2], Yifei Zhang[1,2], Gangyan Zeng[3], Bo Fang[1,2], Yu Zhou[1(✉)], Enze Xie[1,2], and Can Ma[1]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
{sunxurui,lvjiahao,zhangyifei0115,fangbo,zhouyu,xieenze,macan}@iie.ac.cn
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3] School of Information and Communication Engineering, Communication University of China, Beijing, China
zgy1997@cuc.edu.cn

Abstract. As a fundamental step in most visual text-related tasks, scene text detection has been widely studied for a long time. However, due to the diversity in the foreground, such as aspect ratios, colors, shapes, *etc.*, as well as the complexity of the background, scene text detection still faces many challenges. It is often difficult to obtain discriminative text-level features when dealing with overlapping text regions or ambiguous regions of adjacency, resulting in suboptimal detection performance. In this paper, we propose Text-specific Region Contrast (TRC) based on contrastive learning to enhance the features of text regions. Specifically, to formulate positive and negative sample pairs for contrast-based training, we divide regions in scene text images into three categories, *i.e.*, text regions, backgrounds, and text-adjacent regions. Furthermore, we design a Text Multi-scale Strip Convolutional Attention module, called TextMSCA, to refine embedding features for precise contrast. We find that the learned features can focus on complete text regions and effectively tackle the ambiguity problem. Additionally, our method is lightweight and can be implemented in a plug-and-play manner while maintaining a high inference speed. Extensive experiments conducted on multiple benchmarks verify that the proposed method consistently improves the baseline with significant margins.

Keywords: Scene Text Detection · Contrastive Learning · Feature Enhancement · Lightweight Method

1 Introduction

Scene text conveys valuable information and thus is of critical importance for the understanding of natural scenes. As an essential step prior to many text-related

Supported by the Natural Science Foundation of China (Grant NO 62376266), and by the Key Research Program of Frontier Sciences, CAS (Grant NO ZDBS-LY-7024).

Q. Liu et al. (Eds.): PRCV 2023, LNCS 14431, pp. 3–14, 2024.
https://doi.org/10.1007/978-981-99-8540-1_1

tasks *e.g.*, text recognition [18], text retrieval [5], *etc.*, scene text detection (STD) has received extensive attention from researchers and practitioners alike.

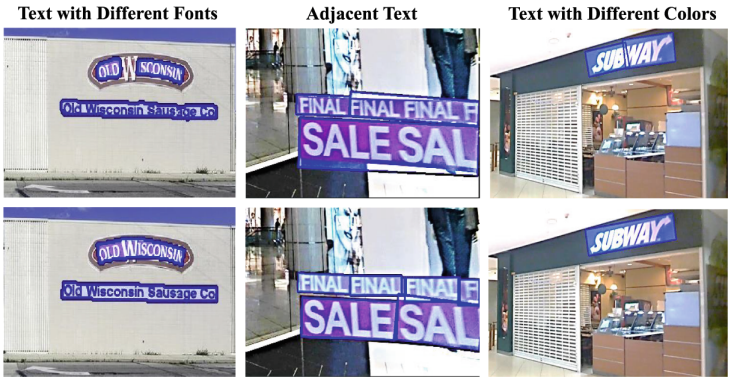

Fig. 1. The left side depicts challenges encountered in scene text detection under various scenarios, while the right side showcases the method's explanation in feature space.

Roughly speaking, most STD methods are inspired by object detection and segmentation paradigms, which can be divided into regression-based methods [12,31] and segmentation-based methods [4,11,17,24,25]. A segmentation-based text detection method has an innate advantage in the detection speed of scene text. Nevertheless, directly applying general segmentation networks for STD suffers from several limitations. First, commonly used semantic segmentation backbones often consist of stacked convolution networks and pooling layers, which can cause a loss of context information and increase false positive samples (e.g., railings misclassified to text category). Second, segmentation-based methods classify images at the pixel level, making adjacent text can not be separated effectively. Third, the larger aspect ratio of scene text makes it difficult for segmentation models designed for general objects to adapt.

Admittedly, there are some efforts to tackle one or more of the problems mentioned above. For example, in targeting the false positive problem, SPCNET [26] utilizes a text context module to strengthen global information and a re-scoring mechanism to filter out false positives. For adjacent text detection, PixelLink [4] introduces the link prediction scheme, which uses 8 neighbors of each pixel to distinguish different instances. PSENet [24] designs a progressive scale expansion algorithm to adapt to text with large aspect ratios. However, due to the diversity of text foreground and the complexity of the background, these methods still struggle to effectively address the challenges of detecting variable text. Moreover, complex feature-processing blocks or post-processing steps are required, which causes a serious decrease in model efficiency.

In this work, aiming to ensure a fast inference speed, we address the aforementioned problems from a feature enhancement perspective. A method termed Text-specific Region Contrast (TRC) is proposed, which utilizes contrastive

learning [19] to obtain more discriminative features for STD. To be specific, different from the common practice [8], we construct a new category of adjacent text samples to form more meaningful contrast pairs accompanied by a novel dynamic region-level sampling strategy. Through these operations, our model aggregates more context information and obtains enhanced features. Furthermore, we design a feature refinement module Text Multi-scale Strip Convolutional Attention (TextMSCA), which utilizes strip convolutions to fit text instances with extreme aspect ratios. As shown in Fig. 1, equipped with the proposed contrastive learning scheme, the detector can obtain more accurate and complete detection results. Extensive experiments conducted on benchmarks show that using TRC and TextMSCA techniques improves the baseline PAN [25] with significant margins. Especially, with the help of SynthText pretraining, our method achieves F-measure performance gains of 2.1%, 2.8%, and 7.5% respectively on the CTW1500 [30], TotalText [3] and MSRA-TD500 [29] datasets while maintaining a high inference speed. The contributions are summarized as follows:

- We propose a novel region-level contrastive learning framework for scene text detection, named Text-specific Region Contrast (TRC), which is able to tackle typical detection challenges via feature enhancement.
- In the proposed framework, text adjacent regions are involved as a new negative category, and a dynamic region-level sampling strategy is implemented based on the weighting of the groundtruth map and score map.
- We design a feature refinement module, *i.e.*, TextMSCA, which could be accustomed to extreme aspect ratios of texts to extract robust representations.
- Extensive experiments demonstrate that the proposed method surpasses the baseline with significant margins and maintains efficient inference overhead.

2 Related Work

Scene Text Detection. Inspired by upstream detection and segmentation frameworks, existing scene text detection methods can be mainly divided into regression-based methods [12,31] and segmentation-based methods [4,11,17,24,25]. Compared with regression-based methods, segmentation-based methods achieve great success to detect curved text instances. In this paper, we focus on segmentation-based methods with lightweight backbones. PAN [25] and DBNet [11] are two representations of real-time methods, which separately design a pixel aggregation and a differentiable binarization mechanism.

Contrastive Learning. Contrastive learning typically pulls positive samples closer and pushes negative samples away. As a pioneering research, MoCo [7] utilizes data augmentation techniques to obtain positive sample pairs and stores negative samples in a memory bank. SimCLR [2] further simplifies this architecture by using other samples in the batch as negative samples. In addition to being applied in unsupervised scenarios, contrastive learning can effectively

assimilate annotation information. Supervised contrastive learning [10] introduces labels into the contrastive paradigm, and this formulation has been successfully applied in semantic segmentation. Wang et al. [23] raise a pixel-to-pixel contrastive learning method for semantic segmentation in the fully supervised setting. Hu et al. [8] form contrastive learning in a region manner.

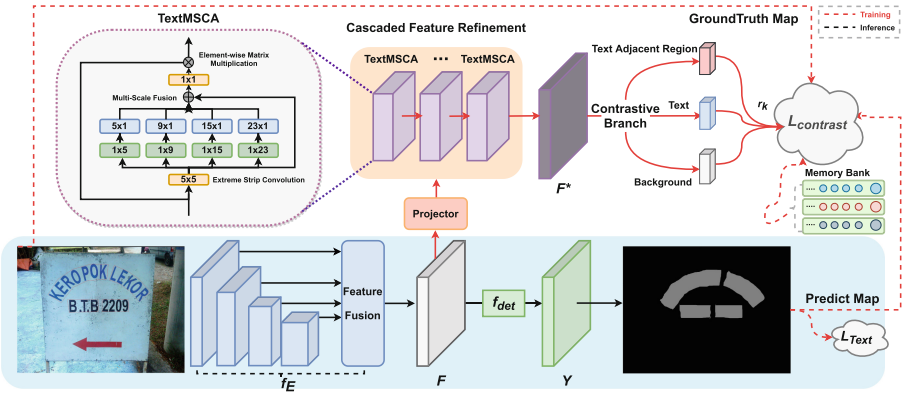

Fig. 2. Illustration of the pipeline. A general segmentation-based scene text detection framework is placed at the bottom, and a plug-and-play contrastive learning branch is placed at the top. Details of the strip convolution setting in TextMSCA are shown in the top left. Text-specific region contrast (TRC) is shown at the top right.

3 Methodology

3.1 Overall Architecture

Previous works [17,24] have demonstrated that segmentation-based text detection methods are effective in handling complex curved text instances. In our experiments, we mainly utilized lightweight segmentation-based pipelines such as PAN [25] as baselines. These models share commonalities: a CNN-based backbone for extracting image features, a feature refinement layer for integrating multi-scale features, and a detection head for classifying dense pixels prior to post-processing the segmentation map.

To state our method better, we pre-define some abbreviations of the general model. We formulate f_E as the combination of the CNN-based backbone and the feature fusion module. Given an input image $x \in \mathbb{R}^{C\times H\times W}$, the feature map $F \in \mathbb{R}^{D\times H\times W}$ can be extracted by feeding x into f_E. C, H, and W represent the channel, height, and width of the image, respectively. D represents the feature dimension of every pixel embedding $F_p \in \mathbb{R}^D, p \in \{p = (i, j)|i = 1, ..., H, j = 1, ..., W\}$ in feature map F. Then a detection head f_{det} maps the feature F into a normalized score map $Y = \{y_p|p\} = f_{det}(F) \in \mathbb{R}^{H\times W}$, which indicates the presence of texts on the position of pixel p.

In our methods, we make use of the feature map F as the embedding input in the refinement and contrast branch. We design a cascaded feature refinement module TextMSCA to transfer F into $F^* \in \mathbb{R}^{D\times H\times W}$ to obtain more suitable text-specific features. And the binary groundtruth map $\hat{Y}$ is the supervised signal in contrastive paradigm. With the supervision of score map Y and the groundtruth map $\hat{Y} \in \mathbb{R}^{H\times W}$, we aggregate features of text regions, background, and text adjacent regions and perform contrastive learning in F^*. Details are presented in Sect. 3.2, and the pipeline of our approach is displayed in Fig. 2.

3.2 Sampling in Text-Specific Region Contrast

The cross-image contrastive paradigm includes two sample groups: one from a mini-batch and another from a memory bank. Both groups have samples from the text or background category. We notice that nearby texts are often linked due to proximity and similarity, requiring special consideration. To handle this, we introduce negative samples called text adjacent regions (shown in Fig. 3). This helps address ambiguity in adjacent text connections. Finally, we use a contrastive loss to bring positive samples closer and separate negative samples. In this section, we provide a detailed account of the negative sample construction and sampling strategy. Additionally, we explain how to construct features of text adjacent regions.

1) Region-Level Sampling. Region-Level feature construction pays attention to designing an overall feature that comes from the average pooling of pixel embedding belonging to the same class in an image. To make full use of labels, we use predicted score maps and groundtruth masks to distinguish true or false foreground pixel samples and assign them different weights, which means to guide the model focusing on samples easy to misclassify. Given the enhanced feature map $F^* \in \mathbb{R}^{D\times H\times W}$ and score map $Y \in \mathbb{R}^{H\times W}$, firstly we use a predefined threshold t which follows the experiment setting in [25] and groundtruth map $\hat{Y} \in \mathbb{R}^{H\times W}$ to obtain right or wrong anchors. For example, for a specified category *text*: $\{k = 1|category = text\}$, right pixel map can be denoted as $RMap^1 = \mathbb{1}(Y_p \geq t) \cap \mathbb{1}(\hat{Y}_p \geq t)$, while wrong pixel map can be denoted as $WMap^1 = \mathbb{1}(Y_p < t) \cap \mathbb{1}(\hat{Y}_p \geq t)$. The definition of *text* region anchor could be set as:

$$r_k = \frac{\sum_p F_p^* \cdot ((1 - y_p) \cdot RMap_p^k + WMap_p^k)}{\sum_p \mathbb{1}|\hat{Y}_p^k|}, \tag{1}$$

where k denotes the class of *text*: $\{k = 1|category = text\}$ or *background*: $\{k = 0|category = background\}$ while $\hat{Y}_p \geq t$ denotes Y_p^{text} and $\hat{Y}_p < t$ denotes $Y_p^{background}$. And $\mathbb{1}(\cdot)$ represents the binary classifier for pixel in $\hat{Y}$ and Y and $\mathbb{1}|\cdot|$ denotes the operation employed on corresponding category k, such as when $k = text$, $\mathbb{1}|\hat{Y}_p^k| = \mathbb{1}|\hat{Y}_p^k > t|$.

Regarding the other group of sampling pixels in the memory bank, we update the region anchor vectors within a mini-batch to the corresponding memory

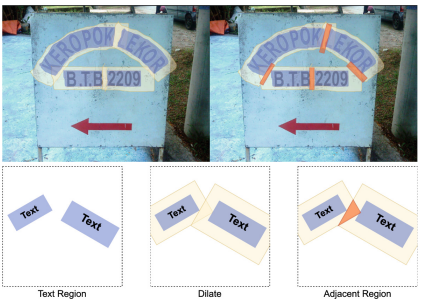

Fig. 3. We visualize and explain the specific design of contrast sampling. As we can see, we add a new category (red areas) of the negative sample to represent the text adjacent region. (Color figure online)

Algorithm 1. Text Adjacent Region Generation

Input: batchsize B, refined feature F^*, dilated coefficient σ, GT map Y, area of text instance set T of each GT map
Output: text adjacent region embedding $r_{adjacent} \in \mathbf{R}^{B \times D}$

```
for i-th feature and GT map in B do
    Refined feature F_i^*, GT map Y_i.
    for k-th text instance T_k in Y_i do
        Dilate region T_k with coef. σ.
    end for
    Get intersection region r_i in Y_i.
    Apply Eq. 2 with F_i^* and Y_i.
end for
Get text adjacent region embedding r_adjacent.
```

queue after computing the training loss. These updated vectors are then used in the subsequent iterations. The complete sampling process is illustrated in the Supplemental Material.

2) Text Adjacent Region Generation. We discovered a false detection issue in some segmentation-based scene text detection models. When two text instances are close to each other, they may merge into one instance. This is difficult to avoid due to dense text and a single category. To address this problem, we creatively constructed negative samples representing adjacent text regions as shown in Fig. 3. By separating these areas from the background, we enable the detection model to recognize the existence of text adjacent areas.

The features of text adjacent regions are calculated separately. We dilate text regions with an offset by the Vatti clipping algorithm and get intersections of dilated regions to represent text adjacent region $TAMap$. The adjacent region feature is calculated by using Eq. 2 and details are shown in Algorithm 1.

$$r_{adjacent} = \frac{\sum_{x,y} F^* \cdot TAMap}{\sum \mathbb{1}|\hat{Y}_{adjacent}|}. \tag{2}$$

3.3 Text-Specific Feature Refinement Module

Features used in contrastive learning mostly arise from general feature fusion layers. Inspired by the MSCA module in SegNeXt [6], we design a text-specific multi-scale strip convolutional attention module (TextMSCA) to refine features F from f_E because text often has strip-like shape and extreme aspect ratios. As shown in Fig. 2, we look into the aspect ratio of text in the dataset and set four special strip convolutions to refine text features.

$$F^* = Conv_{1\times1}(\sum_{i=0}^{4} SC_i(DC(F))) \otimes F \tag{3}$$

Here DC denotes depth-wise convolution and SC_i denotes the i-th strip convolution branch in TextMSCA. $\otimes$ is the element-wise matrix multiplication operation. After putting the feature map into this module, TextMSCA can aggregate the strip context and extract better text-specific features. Therefore, F^* represents scene text features better in more directions and aspects.

3.4 Loss Function and Inference

Contrastive Loss. After describing our contrastive sampling and refinement module, we introduce the supervised contrastive formula. This method is employed when data is labeled. Our method is derived from self-supervised contrastive learning, which uses the InfoNCE [16] loss function. For segmentation-based tasks, we need a more fine-grained loss calculation at the region and pixel levels defined below:

$$\mathcal{L}_{contrast} = \frac{1}{|M_r|} \sum_{r^+ \in M_r} -log \frac{exp(r \cdot r^+/\tau)}{exp(r \cdot r^+/\tau) + \sum_{r^-} exp(r \cdot r^-/\tau)} \tag{4}$$

In detail, r denotes the anchor feature, when faced with region-level contrast loss, r^+ means the positive region samples that belong to the same label as the anchor region r, and r^- means the negative region samples. M_r represents a set of positive regions samples in a mini-batch or the memory bank. Note that region embedding r always comes from the average pooling of pixels embedding in corresponding category regions.

Overall Objective. The overall loss function can be formulated as:

$$\mathcal{L} = \mathcal{L}_{text} + \lambda \cdot \mathcal{L}_{contrast} \tag{5}$$

where $\mathcal{L}_{text}$ denotes the original loss and $\mathcal{L}_{contrast}$ represents the contrast loss used in training stage. λ is a hyper-parameter that balances the weights of $\mathcal{L}_{text}$ and $\mathcal{L}_{contrast}$. The contrast branch does not participate in the inference phase and makes the model maintain the original efficiency.

4 Experiment

4.1 Experimental Setting

Datasets. We utilize 5 typical scene text detection datasets for experiments. **ICDAR 2015** [9] includes many dense and small instances. **CTW1500** [30] and **Total-Text** [3] include instances with various shapes such as horizontal, multi-oriented, and curved situations. **MSRA-TD500** [29] is a multi-language dataset including Chinese and English text instances. At last **MLT-2017** [15] is also a large multi-language dataset that includes 9 languages.

Table 1. Ablation of text adjacent regions as negative samples. TRC(w/o) denotes without consideration of text adjacent regions.

Method	CTW1500			TotalText			TD500		
	P	R	F	P	R	F	P	R	F
PAN	84.6	77.7	81	88	79.4	83.5	80.7	77.3	78.9
PAN+TRC(w/o)	85.7	77.2	81.3	88.3	80	83.9	85.8	78.9	82.1
PAN+TRC	**86.7**	**78.0**	**82.1**	**89.3**	**81.3**	**85.1**	**88.2**	**80.7**	**84.3**

Implementation Details. We use ResNet18 as the lightweight backbone by default and PAN [25] is used as the baseline to evaluate the effectiveness of our method due to its stability and universality. The contrastive learning branch is a plug-and-play component, so the detection branch uses the original baseline settings for hyperparameters. PAN is trained from scratch, and we follow default strategies for data augmentation and optimization. The coefficient λ in Eq. 5 is set to 0.025, and the temperature τ in the contrastive loss is set to 0.7 as in [23]. During inference, we use the same method as the original model, and the TextMSCA module and contrastive branch are not involved.

4.2 Ablation

Effectiveness of Text Adjacent Regions. We created a contrastive sampling strategy specifically tailored to the characteristics of the text, with text adjacent regions added to the memory bank as negative samples. We compared the results of using this strategy versus not using it on three datasets with complex text shapes. Table 1 shows that adding text adjacent regions improves the effectiveness of contrastive learning, especially on the MSRA-TD500 dataset with 2.2% improvement. This design helps the model pay attention to ambiguous regions and effectively distinguish text boundaries, resulting in more robust features and alleviating the problem of connecting adjacent texts.

Effectiveness of TextMSCA. We compared our proposed TextMSCA module with the original MSCA [6] module to validate its effectiveness. We conducted experiments on three datasets, setting the contrastive branch's TRC with two types of negative samples, as in the previous ablation experiment. Table 3 shows that TextMSCA outperforms MSCA by up to 3.1% on MSRA-TD500. The heatmap figure in the supplemental materials further demonstrates that our large-scale strip convolution can better extract text features in various aspect ratios, providing more detailed and robust information in images.

The Flexibility of TRC. To justify the flexibility of TRC in a plug-and-play manner, we transfer the contrastive branch into another segmentation-based scene text detection model DB [11] and explore whether it brings improvements

Table 2. Ablation on DB.

Method	IC15	MLT17	CTW1500	TotalText	TD500
DB	82.3	71.0	81.0	82.8	82.8
DB+TPC	82.1	**72.5**	**81.4**	83.2	83.2
DB+TRC	**82.6**	72.1	**81.4**	**84.0**	**84.1**

Table 3. Ablation of TextMSCA.

Method	CTW1500			TotalText			TD500		
	P	R	F	P	R	F	P	R	F
PAN	84.6	77.7	81	88	79.4	83.5	80.7	77.3	78.9
+MSCA	86.3	77.9	81.9	88.4	80.5	84.3	84.7	78	81.2
+TextMSCA	**86.7**	**78**	**82.1**	**89.3**	**81.3**	**85.1**	**88.2**	**80.7**	**84.3**

Table 4. Comparisons on CTW1500. $*$ means adding TextMSCA.

Method	Ext.	P	R	F	FPS
EAST [32]	–	78.7	49.1	60.4	21.2
TextSnake [14]	✓	67.9	85.3	75.6	1.1
TextField [27]	✓	83.0	79.8	81.4	–
TextRay [21]	✓	82.8	80.4	81.6	–
ABCNet [12]	✓	81.4	78.5	81.6	–
DBNet(R18) [11]	✓	84.8	77.5	81.0	55
CTN(R18) [17]	–	85.5	79.2	82.2	40.8
Fuzzy(R18) [22]	✓	84.6	77.7	81.0	35.2
PAN(R18) [25]	–	84.6	77.7	81.0	39.8
PAN(R18)*+TPC	–	86.9	77.9	82.1	38.2
PAN(R18)*+TRC	–	86.7	78.0	82.1	34.2
PAN(R18)*+TRC	✓	87.2	79.4	**83.1**	34.2

or not. We employ the contrastive branch to train a new DB model without TextMSCA module on five classic datasets and results are shown in Table 2. Text Pixel Contrast (TPC) [23] is a pixel-level sampling method for semantic segmentation tasks. We also compare it with TRC in the experiment. Experimental results show that with the attachment of the contrastive branch, DB gains 1.5%, 1.2%, and 1.3% improvements on MLT17, TotalText, and TD500, respectively. Moreover, TRC outperforms TPC on multiple datasets, which demonstrates the effectiveness of our method.

4.3 Performance Comparison

To evaluate the effectiveness of our method, we conduct thorough experiments on three benchmark datasets CTW1500, TotalText, and MSRA-TD500 in both qualitative and quantitative forms.

Table 5. Comparisons on TotalText. $*$ means adding TextMSCA.

Method	Ext.	P	R	F	FPS
TextSnake [14]	✓	82.7	74.5	78.4	–
PSENet-1 s [24]	✓	84.0	78.0	80.9	3.9
TextField [27]	✓	81.2	79.9	80.6	–
CRAFT [1]	✓	87.6	79.9	83.6	–
DBNet(R18) [11]	✓	88.3	77.9	82.8	50
ABCNet [12]	✓	87.9	81.3	84.5	–
OPMP [31]	✓	85.2	80.3	82.7	3.7
CTN [17]	–	–	–	85.6	24
Fuzzy [22]	✓	88.7	79.9	84.1	24.3
PAN(R18) [25]	–	88.0	79.4	83.5	39.6
PAN(R18)*+TPC	–	88.5	80.4	84.3	39.1
PAN(R18)*+TRC	–	89.3	81.3	85.1	37.9
PAN(R18)*+TRC	✓	90.7	82.4	**86.3**	37.2

Table 6. Comparisons on MSRA-TD500. $*$ means adding TextMSCA.

Method	Ext.	P	R	F	FPS
MCN [13]	✓	88.0	79.0	83.0	–
PixeLink [4]	✓	83.0	73.2	77.8	3
TextSnake [14]	✓	83.2	73.9	78.3	1.1
MSR [28]	✓	87.4	76.7	81.7	–
CRAFT [1]	✓	88.2	78.2	82.9	8.6
SAE [20]	✓	84.2	81.7	82.9	–
DBNet [11]	✓	91.5	79.2	84.9	32
CTN [17]	–	–	–	83.5	20.5
Fuzzy [22]	✓	89.3	81.6	85.3	–
PAN(R18) [25]	–	80.7	77.3	78.9	30.2
PAN(R18)*+TPC	–	85.8	78.9	82.1	33.7
PAN(R18)*+TRC	–	88.2	80.7	84.3	32.8
PAN(R18)*+TRC	✓	89.8	83.3	**86.4**	32.6

Firstly, we present the results under different contrastive settings qualitatively. To reflect the effectiveness that contrastive learning brings to the scene text detection, we follow the training mode for PAN [25] that only uses the **ResNet18** as the backbone **with or without** pretraining using SynthText. As we can see in Table 4, it brings 1.1% improvements to PAN without pretraining. In Table 5 and Table 6, results show that it brings a maximum of 1.6% improvements on TotalText and 5.4% improvements on MSRA-TD500. We also pretrain the model with SynthText and finetune the model on these datasets, and it brings further improvement. Results from Table 4, Table 5 and Table 6 demonstrate that using cross-image region-level contrastive learning assists PAN to get impressive improvements and it even catches up with most methods using ResNet50 as their backbone. FPS would decrease slightly due to the increasing number of connected areas in pixel aggregation before filtering, but the time consumption is negligible. In addition, Fig. 4 shows our visualization results. Intuitively, the contrastive branch with TRC and TextMSCA solves the problem of text detection errors such as stripe-like patterns. When faced with a line of text with different colors or fonts, it also performs better. The quantitative result and the qualitative visualization prove that our proposed model extracts more robust text features from images instead of relying on colors or shapes.

Fig. 4. Visualization of groundtruth (left), results of the original PAN (middle), and PAN with the proposed contrastive branch (right).

5 Conclusion

In this paper, we propose a text-specific region contrast (TRC) method to enhance the feature of text regions for scene text detection. Then, a text multi-scale strip convolutional attention module (TextMSCA) is designed to further refine embedding feature. We conduct extensive experiments and visualizations to demonstrate the effectiveness of contrastive learning in enhancing text detection. Moreover, the proposed contrastive branch is a plug-and-play component

without introducing additional computation during the inference phase. In the future, we will extend our method to address other relevant tasks, for example, multi-language text detection.

References

1. Baek, Y., Lee, B., Han, D., Yun, S., Lee, H.: Character region awareness for text detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9365–9374 (2019)
2. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International Conference on Machine Learning, pp. 1597–1607. PMLR (2020)
3. Ch'ng, C.K., Chan, C.S.: Total-Text: a comprehensive dataset for scene text detection and recognition. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), vol. 1, pp. 935–942. IEEE (2017)
4. Deng, D., Liu, H., Li, X., Cai, D.: PixelLink: detecting scene text via instance segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32 (2018)
5. Gómez, L., Mafla, A., Rusiñol, M., Karatzas, D.: Single shot scene text retrieval. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Computer Vision – ECCV 2018. LNCS, vol. 11218, pp. 728–744. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01264-9_43
6. Guo, M.H., Lu, C.Z., Hou, Q., Liu, Z., Cheng, M.M., Hu, S.M.: SegNeXt: rethinking convolutional attention design for semantic segmentation. arXiv preprint arXiv:2209.08575 (2022)
7. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9729–9738 (2020)
8. Hu, H., Cui, J., Wang, L.: Region-aware contrastive learning for semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 16291–16301 (2021)
9. Karatzas, D., et al.: ICDAR 2015 competition on robust reading. In: 2015 13th International Conference on Document Analysis and Recognition (ICDAR), pp. 1156–1160. IEEE (2015)
10. Khosla, P., et al.: Supervised contrastive learning. Adv. Neural. Inf. Process. Syst. **33**, 18661–18673 (2020)
11. Liao, M., Wan, Z., Yao, C., Chen, K., Bai, X.: Real-time scene text detection with differentiable binarization. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 11474–11481 (2020)
12. Liu, Y., Chen, H., Shen, C., He, T., Jin, L., Wang, L.: ABCNet: real-time scene text spotting with Adaptive Bezier-Curve network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9809–9818 (2020)
13. Liu, Z., Lin, G., Yang, S., Feng, J., Lin, W., Goh, W.L.: Learning Markov clustering networks for scene text detection. arXiv preprint arXiv:1805.08365 (2018)
14. Long, S., Ruan, J., Zhang, W., He, X., Wu, W., Yao, C.: TextSnake: a flexible representation for detecting text of arbitrary shapes. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11206, pp. 19–35. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01216-8_2

15. Nayef, N., et al.: ICDAR 2017 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), vol. 1, pp. 1454–1459. IEEE (2017)
16. Oord, A.V.D., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
17. Sheng, T., Chen, J., Lian, Z.: CentripetalText: an efficient text instance representation for scene text detection. Adv. Neural. Inf. Process. Syst. **34**, 335–346 (2021)
18. Shi, B., Yang, M., Wang, X., Lyu, P., Yao, C., Bai, X.: ASTER: an attentional scene text recognizer with flexible rectification. IEEE Trans. Pattern Anal. Mach. Intell. **41**(9), 2035–2048 (2018)
19. Tian, Y., Krishnan, D., Isola, P.: Contrastive multiview coding. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12356, pp. 776–794. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58621-8_45
20. Tian, Z., et al.: Learning shape-aware embedding for scene text detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4234–4243 (2019)
21. Wang, F., Chen, Y., Wu, F., Li, X.: TextRay: contour-based geometric modeling for arbitrary-shaped scene text detection. In: Proceedings of the 28th ACM International Conference on Multimedia, pp. 111–119 (2020)
22. Wang, F., Xu, X., Chen, Y., Li, X.: Fuzzy semantics for arbitrary-shaped scene text detection. IEEE Trans. Image Process. **32**, 1–12 (2022)
23. Wang, W., Zhou, T., Yu, F., Dai, J., Konukoglu, E., Van Gool, L.: Exploring cross-image pixel contrast for semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 7303–7313 (2021)
24. Wang, W., et al.: Shape robust text detection with progressive scale expansion network. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9336–9345 (2019)
25. Wang, W., et al.: Efficient and accurate arbitrary-shaped text detection with pixel aggregation network. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 8440–8449 (2019)
26. Xie, E., Zang, Y., Shao, S., Yu, G., Yao, C., Li, G.: Scene text detection with supervised pyramid context network. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 9038–9045 (2019)
27. Xu, Y., Wang, Y., Zhou, W., Wang, Y., Yang, Z., Bai, X.: TextField: learning a deep direction field for irregular scene text detection. IEEE Trans. Image Process. **28**(11), 5566–5579 (2019)
28. Xue, C., Lu, S., Zhang, W.: MSR: multi-scale shape regression for scene text detection. arXiv preprint arXiv:1901.02596 (2019)
29. Yao, C., Bai, X., Liu, W., Ma, Y., Tu, Z.: Detecting texts of arbitrary orientations in natural images. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1083–1090. IEEE (2012)
30. Yuliang, L., Lianwen, J., Shuaitao, Z., Sheng, Z.: Detecting curve text in the wild: new dataset and new solution. arXiv preprint arXiv:1712.02170 (2017)
31. Zhang, S., Liu, Y., Jin, L., Wei, Z., Shen, C.: OPMP: an omnidirectional pyramid mask proposal network for arbitrary-shape scene text detection. IEEE Trans. Multimedia **23**, 454–467 (2020)
32. Zhou, X., et al.: EAST: an efficient and accurate scene text detector. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5551–5560 (2017)