Pattern Recognition Letters 172 (2023) 8–14

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

CDText: Scene text detector based on context-aware deformable transformer

Yirui Wu[a,b,c], Qiran Kong[a,b], Lai Yong[c], Fabio Narducci[d], Shaohua Wan[e,f,*]

[a] Key Laboratory of Water Big Data Technology of Ministry of Water Resources, Hohai University, Nanjing 210093, China
[b] College of Computer and Information, Hohai University, Nanjing 210093, China
[c] Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130015, China
[d] Department of Computer Science, University of Salerno, Fisciano, Italy
[e] Key Laboratory of AI and Information Processing, Hechi University, Guangxi, 546300, Yizhou, China
[f] Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen 518110, China

ARTICLE INFO

Article history:
Received 27 December 2022
Revised 18 April 2023
Accepted 22 May 2023
Available online 23 May 2023

Edited by: Maria De Marsico

MSC:
41A05
41A10
65D05
65D17

Keywords:
Disentangled representation learning
Group-disentangled feature representation
Thoracic pathologic prediction

ABSTRACT

Scene text detection task aims to precisely locate text regions in natural scenes. However, the existing methods still face challenges in detecting arbitrary-shaped text, due to their limited feature representation capability. To alleviate this problem, we propose a scene text detector, i.e., CDText, based on structure of context-aware deformable transformer. Specifically, CDText firstly adopts different convolution kernel designs for feature extraction, which designs receptive fields with different size for multi-scale feature perception and fusion. Meanwhile, multi-head self-attention mechanism is used to strengthen the reasoning ability of CDText in a global sense, thus enhancing feature maps with abundant context information by extracting implicit relationship between multi-scale text features. Moreover, CDText designs a segmentation head to segment text instances of arbitrary shapes from rectangular detection boxes. Experiments show that CDText is superior to comparative methods in detection accuracy, achieving *F*-scores of 92.7, 81.9, and 82.9 on ICDAR2013, Total Text, and CTW-1500 datasets, respectively.

1. Introduction

A wide range of applications are built on scene text detectors, such as autonomous driving, real-time translation, document analysis and so on. Since deep learning methods have proved effectiveness in image analysis tasks, researchers have made great progress in scene text detection with deep learning.

To achieve multi-directional text detection, Liu and Jin [1] directly use quadrilateral boxes to detect oblique text. Since directly predicting polygon vertices leads to label confusion due to disorder issue of vertex, Liu et al. [2], Wu et al. [3] propose to first predict key edges of detection boxes through discretization, and then learn label information through a multi-class classifier. Inspired by anchor-free detection method, i.e., DenseBox, Zhou et al. [4], Zhang et al. [5] use a Fully Convolutional Neural Network to generate text detection boxes and the corresponding confidence on pixel-level feature map. One-stage detector with DenseBox cannot handle the detection of long and large texts well. Therefore, Zhong et al. [6] use DenseBox to replace the original Region Proposal Network (RPN) of Faster R-CNN, where the updated method is no longer limited by the anchor box generation mechanism, thus preserving high accuracy of text detection in multiple directions. The above methods mainly consider using a rectangular or a quadrilateral box for text detection, where the curved text cannot be closely surrounded. Inspired by R-FCN for curved text detection, Liu et al. [7] modify the bounding box regression module to use a tighter 14-vertex polygon detection box, where the text candidate regions are further refined by the Recurrent Neural Network (RNN) to be more accurate. Arguing that a fixed 14-point multilateral shape is not enough for long or curved text, Wang et al. [8] use RNN to predict polygon boxes with different numbers of vertices for text regions of different shapes. The idea of instance segmentation is also used to detect curved text, where [9] use Mask R-CNN, the top-down instance segmentation framework, to perform text instance segmentation, thus detecting texts in any shape.

[*] Corresponding author.
E-mail addresses: wuyirui@hhu.edu.cn (Y. Wu), 211307030003@hhu.edu.cn (Q. Kong), laiy@jlu.edu.cn (L. Yong), fnarducci@unisa.it (F. Narducci), shaohua.wan@uestc.edu.cn (S. Wan).

https://doi.org/10.1016/j.patrec.2023.05.025
0167-8655/© 2023 Elsevier B.V. All rights reserved.

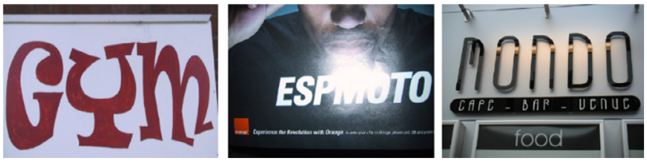

Fig. 1. Difficult samples of text detection in natural scene, such as word art, over-exposure and background complexity arranging from left to right.

Even though deep learning methods have achieved superior performance, accurate text detection in natural scenes is still a challenging task, due to several difficulties shown in Fig. 1. Firstly, texts often appear with arbitrary shapes, different sizes and colors in natural scenes. Compared with texts in documents, illegible or curved texts like wordart are possible to appear as shown in Fig. 1(a). Secondly, background in natural scenes is often complex, where non-uniform lighting, partial occlusion, perspective changes, and image blurring make texts difficult to recognize as showcase of non-uniform lighting in Fig. 1(b). Last but not least, objects in background are easily mistaken for texts, where local features of bricks, fences, windows, vegetation and so on have great similarity with texts as shown in Fig. 1(c). All these difficulties requires text detectors to own strong capabilities in feature extraction.

To detect texts of arbitrary shapes, different sizes and colors, we regard texts as special objects, where a segmentation head is proposed to segment text instances of any shape on the basis of rectangular detection boxes. Essentially, feature maps generated by traditional CNN often lack of contextual information, leading representation capability of text features to be low facing complex natural scenes. To resolve this problem, Yu and Koltun [10], Wu et al. [11] propose the dilated convolution. Different from traditional convolution methods that improve the network receptive field by combining convolution operations with pooling operations, dilated convolution eliminates pooling operations but insert gaps between the kernel elements. It increases the spatial range covered by each convolution kernel without downsampling the feature map, thus avoids information loss. Dilated convolution is useful in tasks such as object recognition and semantic segmentation.

Based on the above ideas, we propose CDText based on context-aware deformable transformer. Specifically, CDTex consists of context-aware feature extractor and transformer structure. The former not only uses dilated convolution operations of different receptive fields to enhance perception capability, but also fuses multi-scale features by pyramid structure design. Meanwhile, the latter samples a small set of key points to focus on informative feature subspace with multi-head self-attention mechanism, thus strengthening the reasoning ability of CDText in a global sense. The generated feature map are rich in context information to detect characters of any shape even in complex background, thus solving the inaccuracy detection with traditional CNN.

The contribution of this paper is three-fold:

- CDT-CAD could detect texts of arbitrary shapes in complex natural scene, owing to the informative contextual information encoding.
- The proposed context-aware feature extractor refines feature map with context information, which exploits and fuses multi-scale interdependencies described by dilated context encoding blocks.
- The proposed deformable transformer aggregates text feature representation of rectangle detection boxes for instance segmentation, thus generating curved and compact bounding boxes.

2. Methodology

2.1. Overall structure

As shown in Fig. 2, CDText consists of ResNet backbone network, context-aware feature extractor, position encoding, Transformer, and segmentation head. Without post-processing steps like non-maximum suppression, CDText could firstly output horizontal text detection boxes, and then segment text instances based on these rectangle boxes, where steps are (1) Send a text image to backbone network for feature extraction; (2) Extract contextual feature information on original feature, where context-aware feature extractor is composed of iterative Dilated context encoder (DCE) blocks to generate multi-scale context feature maps; (3) Expand the obtained feature map to output sequential image features, where position coding is fused to obtain image features containing spatial information; (4) Results are sent to Transformer

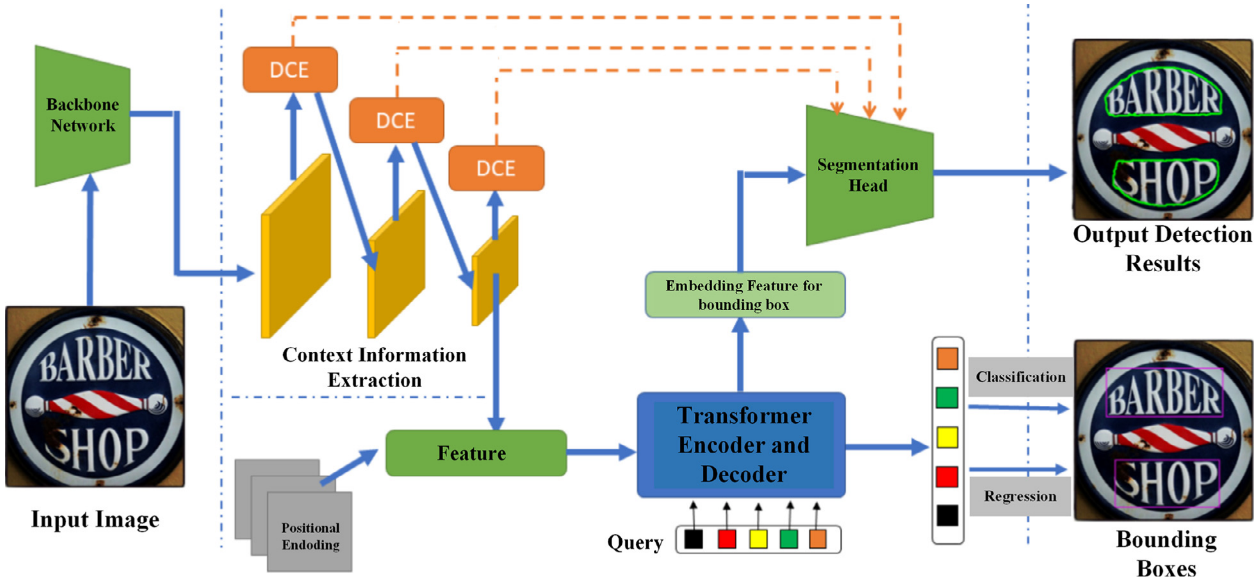

Fig. 2. Network architecture of the proposed CDText. The context-aware feature extractor is composed of iterative Dilated context encoder (DCE) blocks to generate multi-scale context feature maps, transformer module generates classification and embedding features for downstream tasks i.e. classification, detection and segmentation, and segmentation Head module converts the input embedding feature for bounding box into masks of any shape as text detection results.

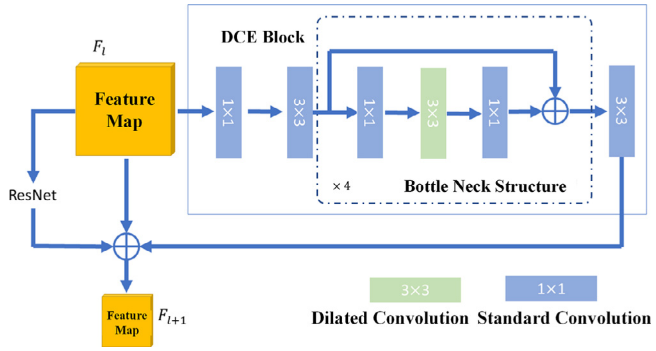

Fig. 3. Structure design of the proposed Dilated context encoder (DCE) blocks. Each block contains several standard convolution layers (for 1 × 1 and 3 × 3) and a di-alated convolutional layer. Dilated convolution is used to obtain a lager receptive field.

with multi-head self-attention mechanism; (5) Results are sent to the feedforward neural network to generate a fixed number of horizontal detection boxes, and their corresponding features; (6) Text detection results are obtained through the segmentation head on the basis of horizontal detection boxes, which could segment text instances in multiple directions and any shape.

2.2. Design of context-aware feature extractor

The proposed context-aware feature extractor adopts an iterative approach for multi-scale feature fusion as shown in Fig. 3. Essentially, feature maps at different levels in ResNet contain different information. For example, shallow feature maps contain spatial information, while deep feature maps correspond to rich semantic information. To effectively fuse multi-scale features and extract key information, CDText adopts Feature Pyramid Network (FPN) [13,14] structure to fuse feature maps from top to bottom. The feature map of each level passes through DCE blocks to perceive context information, which is then encoded into the original feature map with a way similar to skip connections. Afterwards, we reduce the size of feature map by down-sampling, and sum the reduced feature map with the original one for output. The obtained feature map not only encodes abundant context information, but also integrates feature information belonging to top and down levels, thereby continuously improving representation ability of feature map.

Specifically, operations of context-aware feature extractor can be expressed as:

$$\begin{cases} F_l = F_{l-1} + F_{DCE}(F_{l-1}) \\ F_{l+1} = f_{down}(F_{FP}(F_l)) \end{cases} \tag{1}$$

where F_l represents feature map of the lth layer generated after down-sampling operations, + refers to element-wise addition, functions $F_{DCE}()$, $f_{down}()$ and $Res_l()$ represent operations of DCE block, down-sampling and the lth Conv layer of ResNet, where l ranges from 2 to 5. It's noted that dimensions of input and output feature maps remains unchanged after $F_{DCE}()$ and is reduced to half after $f_{down}()$.

The structure of DCE block is shown in Fig. 3. Features pass through a 3 × 3 and a 1 × 1 convolutional operation for finetune, and are further sent to multiple bottleneck structures for enhancing. Specifically, the bottleneck structure firstly uses a 1 × 1 convolutional to reduce the number of channels and the amount of computation. Then, a 3 × 3 Dilated convolutional operation is used to expand the receptive field, which is further restored by a 1 × 1 convolutional operation.

At last, both expanded and original feature are merged by skip connections. It's noted that each bottleneck structure uses dilated convolution with different dilated rates, resulting in receptive fields with different sizes. Therefore, DCE blocks can effectively enhance receptive fields in multiple scales, then fusing multi-scale information to capture informative parts for generation of feature maps rich in context information.

2.3. Encoder and decoder structure in transformer

Compared with traditional convolutional neural networks, Transformer can effectively capture global and key features to distinguish text and background areas.

Since self-attention mechanism in Transformer has property of translation invariance, feature map will lose spatial information after being expanded into sequential features. Therefore, it's necessary to use position encoding to re-add spatial information on the basis of the sequential feature map, which can be expressed as

$$O_{box} = FFN_2(FFN_1(Trans(S))) \tag{2}$$

where O_{box} represents results of detecting rectangle boxes, S is the input serialized feature, function $Trans()$ represents encoder and decoder structure of Transformer, functions $FFN_1()$ and $FFN_2()$ are the fully connected networks to generate features for detection and the horizontal detection boxes respectively, both of which consist of multi-layer linear full connections, activation functions and regularization processing. Both encoder and decoder include multiple layers of encoding and decoding layers, which is set to 4 by experiments. Specifically, the encoding layer consists of a multi-head attention mechanism module and a feed-forward neural network, which requires to add position encoding for supplement of spatial information. The decoding layer share the same design of encoding layer. Unlike decoder layer in natural language processing to compute an element each iteration, the decoding operation of CDText is running in parallel, where a positional encoding number, being the same as number of bounding boxes, is used to generate different detections.

2.4. Design of segmentation head

Inspired by Mask R-CNN [15], we design segmentation head to segment text instances based on the input text rectangle boxes, thus generating masks of any shape as text detection results.

The structure design of segmentation head is shown in Fig. 4, where features of detection boxes computed by Transformer decoder are fed into the multi-head attention mechanism, which outputs heat map with size $N \times M \times H/32 \times W/32$, where N is the number of detection boxes, M represents the number of heads with multi-head attention, H and W are height and width of the input image. It's noted the dimension of M attention heatmaps are

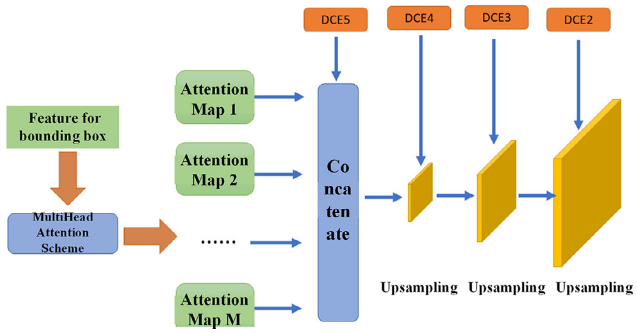

Fig. 4. Structure design of the proposed segmentation head. The multi-head attention scheme shown in the left of this figure generates attention map (heat map) for the latter upsampling progresses. During the upsampling process, heat map is fused with the corresponding DCE feature maps to obtain multiscale spatial information.

the same as the output of last layer of DCE blocks, which is defined as size of input image by down-sampling with 32. Afterwards, M attention heatmaps are concatenated by channel, which is further sent to a FPN-similar structure for three times up-sampling to obtain a confidence feature map for mask prediction with size $N \times H/4 \times W/4$. During these operations, they are fused with output feature map of different DCE blocks containing abundant contextual information through long-distance skip connections. With four stages of feature fusion, multi-scale contextual information can be utilized to help accurately segmente text instances.

2.5. Loss function design

The total loss function is defined as

$$L = \sum_{i=1}^{N}(L_p + \zeta_{p_i \neq \emptyset} L_b + \zeta_{p_i \neq \emptyset} L_s) \tag{3}$$

where L_p, L_b and L_s represent classification loss, regression loss of bounding boxes and segmentation loss respectively, function $\zeta()$ means the output value is 1 once the condition is true, otherwise is 0, $p_i \neq \emptyset$ means the predicted category of the ith element is true, that is, text and background are correctly distinguished. Since text detection is a binary classification problem where only two categories required to be classified, we define such condition as if $P_i > 0.5$.

Among these losses, L_p adopts cross entropy loss:

$$L_p(p_\omega(i), \hat{p}_i) = -p_\omega(i)\log(\hat{p}_i) \tag{4}$$

where $p_\omega(i)$ represents the ith label after the optimal replacement found by the Hungarian algorithm. When the label indicates text, its value is 1. Otherwise, it's 0. $\hat{p}_i$ represents the probability of the ith prediction. L_b is defined with GIoU loss and L1 loss:

$$L_b(b_\omega(i), \hat{b}_i) = \lambda_{iou} L_{giou}(b_\omega(i), \hat{b}_i) + \lambda_{L1}\|b_\omega(i) - \hat{b}_i\|_1 \tag{5}$$

where $b_\omega(i)$ and $\hat{b}_i$ represent the ith label and the coordinates of the ith predicted detection box respectively, function $L_{giou}()$ represent GIoU loss defined as the ratio of the overlapping area between two boxes to the area of the smallest rectangle that completely covers two boxes, $\| \cdot \|_1$ represents the L1 norm, λ_{iou} and λ_{L1} are hyperparameters defined as 0.4 and 0.6, respectively. L_s is defined as the Dice loss related to the intersection-over-union ratio (IoU) between masks:

$$L_S(S_\omega(i), \hat{S}_i) = 1 - \frac{2S_\omega(i)\sigma(\hat{S}_i) + 1}{\sigma(\hat{S}_i) + S_\omega(i) + 1} \tag{6}$$

where $S_\omega(i)$ and $\hat{S}_i$ are the ground-truth and predicted ith text instance segmentation results respectively, and function $\sigma()$ is the sigmoid activation function.

To train CDText, we firstly trains a sub-network for rectangular box text detection, then freezes the weight of the sub-network part, and finally fine-tunes the network with the additional segmentation head. After three steps of training, we could achieve an arbitrary-shaped text detector as CDText.

3. Experiments

3.1. Datasets and measurements

We test CDText on three datasets, i.e., ICDAR2013, SCUT CTW-1500 and Total Text. Specifically, ICDAR2013 is a dataset with the horizontal English texts, where 229 images are used as training set, and the remaining 225 images are used as test set. SCUT CTW-1500 is a dataset for detecting irregularly shaped text with both English and Chinese characters, where 1000 images are used for training and the remaining 500 images are used for testing. In dataset, text instances are marked with polygon coordinates, with a total of 14 boundary points for marking. Total Text is a challenging English dataset with horizontally oriented text, slanted text, and some curved text, where 1255 images are used for training and 300 images are used for testing. Unlike CTW-1500, the total number of labeling points is not fixed. Measurements are defined as Precision (P for short), Recall (R for short) and F-score (F for short). While ICDAR2013 dataset focuses more on straight text layout, Total Text and CTW-1500 datasets has more irregular text layout samples. Due to their different characteristics, we train 3 different models for each dataset.

Table 1
Performance comparison with the existing methods on ICDAR2013, total text and CTW-1500 dataset.

Methods	ICDAR 2013			Total text			CTW-1500		
	P	R	F	P	R	F	P	R	F
CTPN [16]	93.0	83.0	88.0	–	–	–	60.4	53.8	56.9
SegLink [17]	87.7	83.0	85.3	30.3	23.8	26.7	42.3	40.0	40.8
TextBoxes [18]	88.0	74.0	81.0	62.1	45.5	52.5	–	–	–
Mask R-CNN [15]	91.5	89.2	90.2	80.5	79.5	80.0	80.2	82.9	81.5
DDR [19]	92.0	81.0	86.0	–	–	–	–	–	–
CTD + TLOC [20]	–	–	–	74.0	71.0	73.0	77.4	69.8	73.4
EAST [4]	93.0	93.0	87.0	50.0	36.2	42.0	78.7	49.1	60.4
PixelLink + MS [21]	88.6	87.5	88.1	–	–	–	–	–	–
RRD + MS [22]	92.0	86.0	89.0	–	–	–	–	–	–
TextSnake [23]	–	–	–	82.7	74.5	78.4	67.9	85.3	75.6
LOMO [24]	–	–	–	87.6	79.3	83.3	85.7	76.5	80.8
MSR [25]	91.8	88.5	90.1	85.2	78.6	78.6	84.1	79.0	81.5
PSENet [26]	93.7	87.8	90.7	84.0	77.9	80.9	80.6	75.6	78.0
CSE [27]	93.7	89.7	91.7	81.4	79.1	80.2	81.0	76.0	78.4
TextDragon [28]	–	–	–	85.6	75.7	80.3	84.5	82.8	83.6
TextField [29]	–	–	–	81.2	79.9	80.6	79.8	83.0	81.4
ATRR [30]	–	–	–	80.9	76.2	78.5	80.1	80.1	80.1
The proposed	94.1	91.4	92.7	82.0	81.8	81.9	82.6	83.3	82.9

3.2. Comparison experiments

Horizontal text detection As shown in Table 1, the experimental results on ICDAR2013 dataset show high detection performance of CDText on horizontal texts, since ICDAR2013 mainly consist of horizontal texts. Moreover, segmentation head of CDText is not trained for comparison, due to the exitance of only horizontal texts. Compared with Mask R-CNN [15], CDText achieves superior results in precision, recall and F-score. With the help of context-aware feature extractor and Transformer with self-attention mechanism, CDText has a larger receptive field to captures global information, thus effectively improving the ability to distinguish background and text areas. It should be noticed that we only keep the box regression branch and classification branch of the Mask R-CNN (mask branch is removed from that scheme). The model is trained and fine-tuned on the three datasets in our experiment.

In Table 1, MS indicates the use of multi-scale testing. F-score of the RRD [22] with MS improves F-score with 8 points. Even though CDText adopts a single-scale strategy in testing, its performance still exceeds methods with MS strategy. We thus conclude that CDText can effectively perceive context information in multiple scales to extract abundant and global information. Since evaluation indicators of ICDAR2013 dataset tolerate one-to-one and many-to-many detection results to a certain extent, F-scores on ICDAR2013 dataset are relatively high, where CDText reach more than 90.

Arbitrary Shape Text Detection Table 1 shows the results of the comparative experiments on Total Text and CTW-1500 datasets with irregularly shaped texts, where we can observe F-score of CDText reach 81.9 and 82.9 respectively, exceeding all other text detection methods and proving the ability of CDText to detect texts of arbitrary shapes.

Similar to Mask R-CNN [15], CDText achieve segmentation of text instances based on detection boxes with an upsampling strategy of FPN [13]. However, CDText outperforms Mask R-CNN in terms of precision, recall, and *F*-score on both datasets containing texts of arbitrary shapes, which proves the effectiveness of contextual modeling idea in CDText to help segment text instances of any shape. It's noted that CDText remains consist performance on both tasks of detecting horizontal and arbitrary texts. On the contrary, methods such as East and SegLink, are not optimized for curved text detection, where these methods perform slightly worse on Total Text and CTW-1500. Compared with other text detection methods based on semantic segmentation like TextSnake, CDText is 3.5 and 7.3 points higher on Total Text and CTW-1500 datasets, respectively, meanwhile recall is only slightly lower on CTW-1500 dataset. In fact, semantic segmentation on the whole image often leads to insufficient accuracy, such as the unclear boundary of text, and the broken text instance. Therefore, distinguishing text instances on semantic segmentation results in slightly worse detection performance. CDText segments text instances on the basis of rectangular detection boxes, leading to a certain degree of accuracy advantage.

It's noted that CDText obtains best results on *F*-score on Total Text and CTW-1500 datasets. However, we fail to achieve best performance in either precision or recall. In fact, some methods are designed to emphasize parameter precision or recall with specific structure, where CDText take both parameters into account for a better *F*-score performance.

3.3. Parameter setting experiments

In this subsection, several sets of parameter, i.e., the number of DCE layers, the number of bottleneck structures, the number of Transformer encoder and decoder layers, and the way to achieve position encoding, are tested on Total text dataset to determine. We show experimental results in Table 4 for further analysis.

Number of DCE layers Note that number equals 0 means the context-aware feature extractor is completely removed. When equalling 4, we adopt the last 4 convolutional layers of backbone ResNet to be connected with the proposed DCE layer. When equalling 3, the second convolutional layer of ResNet is no longer connected with DCE layer to directly compute. When equalling 5, the first convolutional layer of ResNet is also connected with DCE layer. In all these experiment, the number of bottleneck structures is set to 4 in all DCE layers, and the dilation rate of dilated convolution is 3, 5, 7, and 3, respectively.

Compared with CDText without the context-aware feature extractor, *F*-score increases from 79.9 to 81.9 by setting the number of DCE layers to 4. When number of DCE layers is less than 4, more DCE Layers lead to higher *F*-score. When number of DCE layer is defined as 5, text detection performance is slightly reduced, which proves that 4 is the optimal option. In fact, DCE blocks of different layers perceive context information of different scales, where more DCE layers could increase the multi-scale perception ability of CDText. However, too shallow feature maps contain insufficient semantic information, thus DCE blocks with the first layer failing to bring significant improvement.

Settings bottleneck structures The number of DCE layers is fixed to 4, when testing the optimized number of bottleneck structures. Similar with tests on number of DCE Layers, remove of bottleneck structures starts from the last bottleneck structure. It's proved by results in Table 4 that more bottleneck structures brings higher *F*-score. In fact, different bottleneck structures use dilated convolutions with different dilated rates, resulting in different receptive fields with abundant context information encoded. When increasing from 4 to 5, the detection performance reaches saturation with only a slight increase in precision (Table 2).

Table 2
Performance comparison with different parameter settings on TotalText dataset.

Parameter		Setting	P	R	F
DCE layers		0	80.2	79.7	79.9
		1	81.4	80.5	80.9
		2	81.8	81.4	81.6
		3	81.7	81.8	81.7
		4	**82.0**	**81.8**	**81.9**
		5	81.8	81.8	81.8
Bottleneck	Dilated conv	1	80.4	80.5	80.4
		2	81.3	81.2	81.2
		3	81.8	81.6	81.7
		4	82.0	**81.8**	**81.9**
		5	**82.1**	81.8	81.9
	Tradition conv	1	80.3	79.4	79.3
		2	80.2	79.9	79.9
		3	80.1	80.2	80.5
		4	80.2	80.5	80.2
		5	80.4	80.0	79.8
Transformer Layer		2	73.2	69.5	71.3
		3	80.4	79.6	80.0
		4	82.0	81.8	81.9
		6	**82.5**	**82.0**	**82.2**
Positional Coding		Without	73.9	66.4	69.9
		Learnable	81.0	80.9	80.9
		Sin-code	**82.0**	**81.8**	**81.9**

Moreover, in order to verify the effectiveness of the deflated convolution in the module, we also conduct a new experiment by replacing the dilated conv-layers in the bottleneck blocks with traditional conv-layers of the same kernal size. Experimental results show that the performance of the model deteriorates significantly as the dilated conv-layers are replaced. This strongly demonstrates that the traditional convolution is not able to satisfy the needs of multiscale feature extraction due to information loss and small receptive fields.

Number of transformer encoder and decoder layers In general, more Transformer coding layers brings a higher *F*-value. Compared between performance when setting number as 2 and 4, *F*-score increases from 71.3 to 81.9. When continuing to increase number of layers to 6, *F*-score only increase by 0.3. Since self-attention mechanism of Transformer can perform global reasoning on feature maps, using too few encoder-decoder layers would lack the ability to perform complex global reasoning. Meanwhile, the improvement would converge by more layers. We thus conclude that the reasoning of global information can effectively improve ability of CDText to distinguish text and background.

Positional coding Since the spatial information in feature would lost as input of Transformer, positional coding should be used to complement the missing spatial information. We compares the detection performance of different network structures, i.e., without using positional coding, using learnable positional coding and using sin-wave positional coding. No matter using the learnable or the sine wave positional coding, *F*-score is much higher than that of CDText without positional coding. Compared with the learnable way, detection performance of using sin-wave positional coding is better, which proves the necessity and importance to adopt sin-wave position coding for CDText with Transformer structure.

3.4. Visualization of results

The last layer of Transformer decoder can generate an attention weight for each text instance, which can be used to generate detection boxes. Figure 5 shows the attention weight map and the rectangular detection box of each text instance in an input image, where the right one is the final detection result after segmenting text instances. It can be observed that CDText can effectively focus

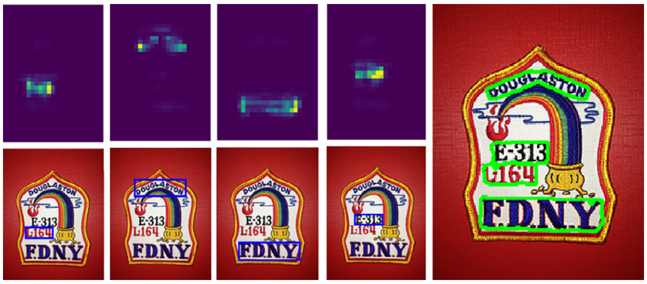

Fig. 5. Sample of attention weight map to show the capability of CDText to focus on key areas.

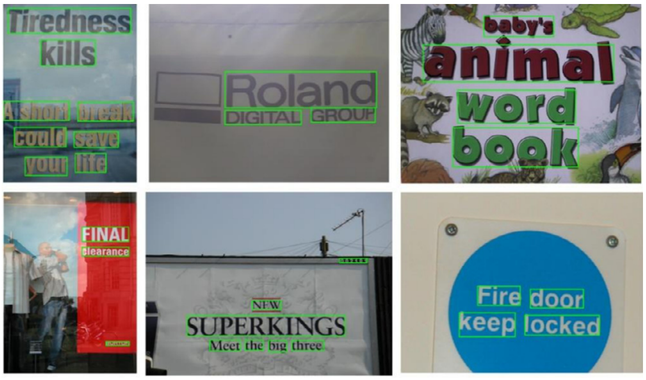

Fig. 6. Samples of detecting texts achieved by CDText on ICDAR2013 dataset.

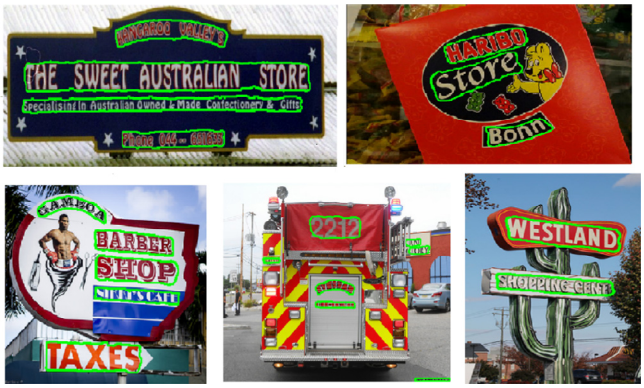

Fig. 7. Samples of detecting texts achieved by CDText on CTW-1500 dataset.

on key areas with text instance inside, so as to accurately detect and segment text instances.

Figure 6 shows the detection results of CDText on ICDAR2013 dataset. Since detecting texts on ICDAR2013 is easy with clear and horizontal texts, samples show the capability of CDText to accurately detect horizontal texts. Figures 7 and 8 show the detection results of CDText on CTW-1500 and Total Text datasets, respectively. It can be seen that texts in CTW-1500 is lined in images, while texts in Total Text are segmented by word form. Compared with ICDAR2013, both datasets are significantly more difficult with curved texts, where CDText still accurately detect texts with any directions and arbitrary shapes, even curved text instances.

3.5. Time consumption test

We randomly selected 900 images (300 images from each dataset) for speed test. We selected two ordinary devices as the experimental hardware: a Core i7-9750H 2.60 GHz Laptop and a Dimensity 700 2.2 GHz smartphone. The average time spent on

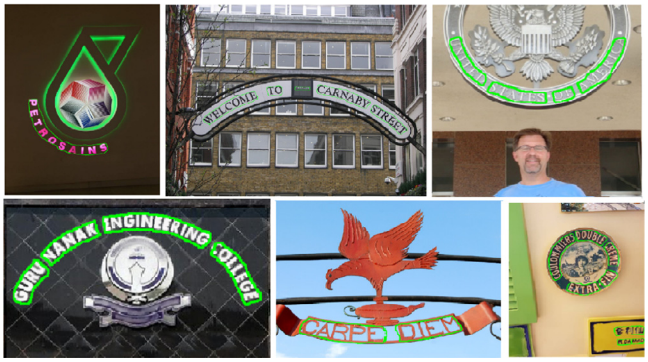

Fig. 8. Samples of detecting texts achieved by CDText on Total Text dataset.

a single image is about 225 ms on laptop and 412 ms on smartphone.

3.6. Implementation details

We use weights pre-trained on ImageNet as the initial value of the backbone ResNet-50 network. We then use Adam optimizer to pre-train CDText on ICDAR2017 MLT dataset, where the initial learning rate and weight decay are set to $1e^{-4}$ and $1e^{-4}$. It's noted that 125 epochs are pre-trained on the detection box sub-network, and 25 another epochs are pre-trained on the segmentation head. For the ICDAR2013 dataset, the detection box subnet is fine-tuned for 300 epochs, and the learning rate is reduced to 1/10 of the original every 200 epochs. On CTW-1500 and Total Text, the detection box subnet is fine-tuned for 800 epochs, the learning rate is reduced to 1/10 every 300 epochs, and the segmentation head is fine-tuned for 100 epochs instead. For training images, randomly scale the short side of images to 480–800, and ensure that its long side doesn't exceed 1333. In the detection box sub-network training phase, the batch size of each GPU is set to 2, meanwhile it's settled to 1 when training on segmentation head phase.

4. Conclusion

This paper proposes a context-aware and Transformer-based approach for scene text detection. It's characterized by using Transformer to increase the global reasoning ability, and using the context-aware feature extractor to perceive and fuse multi-scale feature information to obtain abundant context information. Experiments show that the proposed method achieve accurate performance for text detection in natural scenes by detecting texts with any directions and arbitrary shapes.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgement

This work was supported in part by Key Laboratory of AI and Information Processing (Hechi University), Education Department of Guangxi Zhuang Autonomous Region under Grant 2022GXZDSY014.

References

[1] Y. Liu, L. Jin, Deep matching prior network: toward tighter multi-oriented text detection, in: Proceedings of CVPR, 2017, pp. 3454–3461.

[2] Y. Liu, S. Zhang, L. Jin, L. Xie, Y. Wu, Z. Wang, Omnidirectional scene text detection with sequential-free box discretization, in: Proceedings of IJCAI, 2019, pp. 3052–3058.

[3] Y. Wu, L. Zhang, Z. Gu, H. Lu, S. Wan, Edge-ai-driven framework with efficient mobile network design for facial expression recognition, ACM Trans. Embedded Comput. Syst. 22 (3) (2018) 1–17.

[4] X. Zhou, C. Yao, H. Wen, Y. Wang, S. Zhou, W. He, J. Liang, EAST: an efficient and accurate scene text detector, in: Proceedings of CVPR, 2017, pp. 2642–2651.

[5] Y. Zhang, F. Zhang, Y. Jin, Y. Cen, V. Voronin, S. Wan, Local correlation ensemble with GCN based on attention features for cross-domain person re-id, ACM Trans. Multimed. Comput., Commun. Appl. 19 (2) (2023) 1–22.

[6] Z. Zhong, L. Sun, Q. Huo, An anchor-free region proposal network for faster R-CNN-based text detection approaches, Int. J. Doc. Anal. Recognit. 22 (3) (2019) 315–327.

[7] Y. Liu, L. Jin, S. Zhang, C. Luo, S. Zhang, Curved scene text detection via transverse and longitudinal sequence connection, Pattern Recognit. 90 (2019) 337–345.

[8] X. Wang, Y. Jiang, Z. Luo, C. Liu, H. Choi, S. Kim, Arbitrary shape scene text detection with adaptive text region representation, in: Proceedings of CVPR, 2019, pp. 6449–6458.

[9] Y. Xiao, M. Xue, T. Lu, Y. Wu, S. Palaiahnakote, A text-context-aware CNN network for multi-oriented and multi-language scene text detection, in: Proceedings of ICDAR, 2019, pp. 695–700.

[10] F. Yu, V. Koltun, Multi-scale context aggregation by dilated convolutions, in: Proceedings of ICLR, 2016.

[11] Y. Wu, Q. Kong, L. Zhang, A. Castiglione, M. Nappi, S. Wan, CDT-CAD: context-aware deformable transformers for end-to-end chest abnormality detection on X-ray images, IEEE/ACM Trans. Comput. Biol. Bioinf. (2023), doi:10.1109/TCBB.2023.3258455.

[12] Y. Wu, H. Guo, C. Chakraborty, M. Khosravi, S. Berretti, S. Wan, Edge computing driven low-light image dynamic enhancement for object detection, IEEE Trans. Netw. Sci. Eng. (2022), doi:10.1109/TNSE.2022.3151502.

[13] T. Lin, P. Dollár, R.B. Girshick, K. He, B. Hariharan, S.J. Belongie, Feature pyramid networks for object detection, in: Proceedings of CVPR, 2017, pp. 936–944.

[14] J. Gou, L. Sun, B. Yu, S. Wan, W. Ou, Z. Yi, Multi-level attention-based sample correlations for knowledge distillation, IEEE Trans. Ind. Inf. 19 (5) (2022) 7099–7109, doi:10.1109/TII.2022.3209672.

[15] K. He, G. Gkioxari, P. Dollár, R.B. Girshick, Mask R-CNN, in: Proceedings of ICCV, 2017, pp. 2980–2988.

[16] Z. Tian, W. Huang, T. He, P. He, Y. Qiao, Detecting text in natural image with connectionist text proposal network, in: Proceedings of ECCV, vol.9912, 2016, pp. 56–72.

[17] B. Shi, X. Bai, S.J. Belongie, Detecting oriented text in natural images by linking segments, in: Proceedings of CVPR, 2017, pp. 3482–3490.

[18] M. Liao, B. Shi, X. Bai, X. Wang, W. Liu, Textboxes: a fast text detector with a single deep neural network, in: Proceedings of AAAI, 2017, pp. 4161–4167.

[19] W. He, X. Zhang, F. Yin, C. Liu, Deep direct regression for multi-oriented scene text detection, in: Proceedings of ICCV, 2017, pp. 745–753.

[20] Y. Liu, L. Jin, S. Zhang, S. Zhang, Detecting curve text in the wild: new dataset and new solution, CoRR abs/1712.02170 (2017).

[21] D. Deng, H. Liu, X. Li, D. Cai, Pixellink: detecting scene text via instance segmentation, in: Proceedings of AAAI, 2018, pp. 6773–6780.

[22] M. Liao, Z. Zhu, B. Shi, G. Xia, X. Bai, Rotation-sensitive regression for oriented scene text detection, in: Proceedings of CVPR, 2018, pp. 5909–5918.

[23] S. Long, J. Ruan, W. Zhang, X. He, W. Wu, C. Yao, Textsnake: a flexible representation for detecting text of arbitrary shapes, in: Proceedings of ECCV, vol.11206, 2018, pp. 19–35.

[24] C. Zhang, B. Liang, Z. Huang, M. En, J. Han, E. Ding, X. Ding, Look more than once: an accurate detector for text of arbitrary shapes, in: Proceedings of CVPR, 2019, pp. 10552–10561.

[25] C. Xue, S. Lu, W. Zhang, MSR: multi-scale shape regression for scene text detection, in: S. Kraus (Ed.), Proceedings of IJCAI, 2019, pp. 989–995.

[26] W. Wang, E. Xie, X. Li, W. Hou, T. Lu, G. Yu, S. Shao, Shape robust text detection with progressive scale expansion network, in: Proceedings of CVPR, 2019, pp. 9336–9345.

[27] Z. Liu, G. Lin, S. Yang, F. Liu, W. Lin, W.L. Goh, Towards robust curve text detection with conditional spatial expansion, in: Proceedings of CVPR, 2019, pp. 7269–7278.

[28] W. Feng, W. He, F. Yin, X. Zhang, C. Liu, Textdragon: an end-to-end framework for arbitrary shaped text spotting, in: Proceedings of ICCV, 2019, pp. 9075–9084.

[29] Y. Xu, Y. Wang, W. Zhou, Y. Wang, Z. Yang, X. Bai, Textfield: learning a deep direction field for irregular scene text detection, IEEE Trans. Image Process. 28 (11) (2019) 5566–5579.

[30] X. Liu, G. Zhou, R. Zhang, X. Wei, An accurate segmentation-based scene text detector with context attention and repulsive text border, in: Proceedings of CVPR Workshops, 2020, pp. 2344–2352.