DPText-DETR: Towards Better Scene Text Detection with Dynamic Points in Transformer

Maoyuan Ye[1], Jing Zhang[2], Shanshan Zhao[3], Juhua Liu[1]*, Bo Du[4]*, Dacheng Tao[3,2]

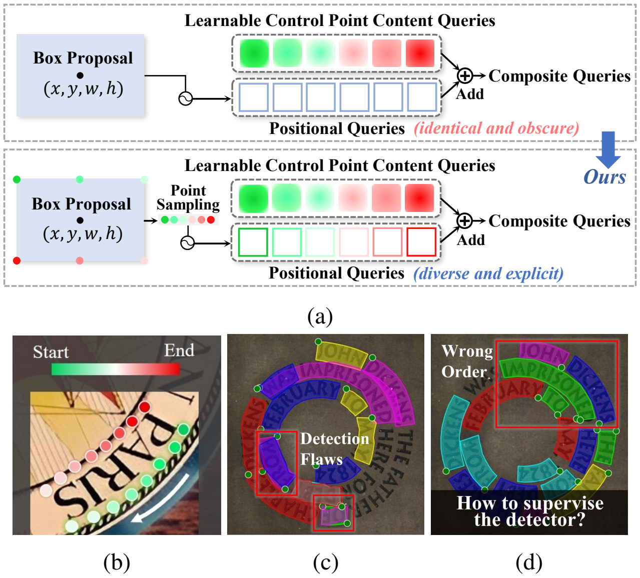
[1] Research Center for Graphic Communication, Printing and Packaging, Institute of Artificial Intelligence, Wuhan University
[2] The University of Sydney
[3] JD Explore Academy
[4] National Engineering Research Center for Multimedia Software, Institute of Artificial Intelligence, School of Computer Science and Hubei Key Laboratory of Multimedia and Network Communication Engineering, Wuhan University
{yemaoyuan, liujuhua, dubo}@whu.edu.cn, jing.zhang1@sydney.edu.au, {sshan.zhao00, dacheng.tao}@gmail.com

Abstract

Recently, Transformer-based methods, which predict polygon points or Bezier curve control points for localizing texts, are popular in scene text detection. However, these methods built upon detection transformer framework might achieve sub-optimal training efficiency and performance due to coarse positional query modeling. In addition, the point label form exploited in previous works implies the reading order of humans, which impedes the detection robustness from our observation. To address these challenges, this paper proposes a concise Dynamic Point Text DEtection TRansformer network, termed DPText-DETR. In detail, DPText-DETR directly leverages explicit point coordinates to generate position queries and dynamically updates them in a progressive way. Moreover, to improve the spatial inductive bias of non-local self-attention in Transformer, we present an Enhanced Factorized Self-Attention module which provides point queries within each instance with circular shape guidance. Furthermore, we design a simple yet effective positional label form to tackle the side effect of the previous form. To further evaluate the impact of different label forms on the detection robustness in real-world scenario, we establish an Inverse-Text test set containing 500 manually labeled images. Extensive experiments prove the high training efficiency, robustness, and state-of-the-art performance of our method on popular benchmarks. The code and the Inverse-Text test set are available at https://github.com/ymy-k/DPText-DETR.

Introduction

Text reading and understanding have aroused increasing research interest in the computer vision community (Liao et al. 2021; Liao et al. 2020a; Liu et al. 2020b, 2021a; Zhang et al. 2020; Singh et al. 2019; He et al. 2022; Du et al. 2022; Liu et al. 2020a; Qiao et al. 2021; Zhou et al. 2021), due to the wide range of practical applications, such as autonomous driving (Zhang and Tao 2020). To achieve it, as a prerequisite, scene text detection has been studied extensively. However, the distinction of scene text, *e.g.*, complex styles and arbitrary shapes make detection remain challenging.



(a)

(b) (c) (d)

Figure 1: (a) Comparison of coarse (top) and our explicit (bottom) positional query modeling. (b) The original label implies the reading order of humans. (c) The original label induces the detector to implicitly learn the reading order, resulting in some flaws, such as false positives. Green points are the predicted start points of clockwise reading order. (d) The detector cannot learn the reading order well even with extensive rotation augmentation.

Recently, DETR (Carion et al. 2020) introduces Transformer (Vaswani et al. 2017) to object detection, forming a concise and seminal end-to-end framework. Following DETR, lots of works (Zhu et al. 2020; Dai et al. 2021b; Wang et al. 2021; Meng et al. 2021; Liu et al. 2022; Wang et al. 2022) further improve the training convergence and performance. For example, DAB-DETR (Liu et al. 2022) offers insights on the query which can be formed by a content part and a positional part, and proves that the positional part is essential for the training convergence. However, the above detection transformers predicting axis-aligned boxes fall short in handling arbitrary-shape scene texts. In response,

*Corresponding author. This work was done during Maoyuan Ye's internship at JD Explore Academy.

recent DETR-like methods (Zhang et al. 2022d; Tang et al. 2022) predict polygon control points or Bezier curve control points following ABCNet (Liu et al. 2020b). Specifically, TESTR (Zhang et al. 2022d) enables Deformable DETR (Zhu et al. 2020) to predict polygon results in a subtle manner. TESTR uses anchor box proposals from the Transformer encoder to generate positional queries and provide position prior for control point content queries, as shown in the top part of Fig. 1(a). However, the position prior from box information is coarse and mismatches the target of predicting points to some extent, which impacts the training efficiency. We abbreviate it as the **query formulation issue**.

In addition, although the scheme of predicting control points enables novel solutions to scene text detection, it also introduces an issue *w.r.t* the order of the points. In detail, previous related works adopt the label form of control points according to the reading order of human, as shown in Fig. 1(b). This scheme is straightforward, while we are curious about whether it is necessary to enable the detector to localize the text as the human does for understanding the text. Previous works do not investigate the impact of such control point label form. However, interestingly, we find this form harms the detection robustness when there are inverse-like texts in the training dataset, even though the ratios of such texts are quite low, *e.g.*, about 2.8% in Total-Text (Ch'ng, Chan, and Liu 2020), 5.2% in CTW1500 (Liu et al. 2019), and 5.3% in ICDAR2019 ArT (Chng et al. 2019). We denote it as the **label form issue**. Some detection flaws caused by this issue are shown in Fig. 1(c). Since there are few inverse-like texts in existing benchmarks, we collect an Inverse-Text test set to further investigate the impact of this label form on detection robustness in real-world scenario. The collected dataset contains 500 scene images with about 40% inverse-like texts. We hope Inverse-Text can inspire and facilitate future researches through the preliminary attempt in data by filling the gap of lacking inverse-like texts in existing test sets.

To address the query formulation issue and the label form issue, we propose a novel Dynamic Point Text DEtection TRansformer network termed DPText-DETR. In terms of the query formulation issue, we propose an Explicit Point Query Modeling (EPQM) method. Specifically, instead of using boxes, we directly utilize point coordinates to get positional queries, as illustrated in the bottom part of Fig. 1(a). With the explicit and complete point formulation, the model is able to dynamically update points in decoder layers. Moreover, non-local self-attention lags behind convolution in capturing spatial inductive bias. Hence, we propose an Enhanced Factorized Self-Attention (EFSA) module leveraging circular convolution (Peng et al. 2020) to explicitly model the circular form of polygon points and complement the pure self-attention. In terms of the label form issue, we design a practical positional label form, which makes the start points independent of the semantic content of texts. With this simple operation, it can noticeably improve the detection robustness.

Overall, our main contributions are three-fold:

- We propose DPText-DETR, which improves the training convergence and the spatial inductive bias of self-attention by exploiting EPQM and EFSA modules.
- We investigate the impact of control point label form and design a practical positional label form to improve the detection robustness. We also establish a novel Inverse-Text test set to fill the gap of lacking inverse-like texts in existing datasets.
- DPText-DETR sets new state-of-the-art on representative arbitrarily-shaped scene text detection benchmarks. It also has fast convergence and promising data efficiency.

Related Work

Detection Transformers

Transformer (Vaswani et al. 2017) originates from machine translation and soon becomes popular in computer vision community (Dosovitskiy et al. 2020; Liu et al. 2021b; Xu et al. 2021; Zhang et al. 2022a,b). Recently, the seminal DETR (Carion et al. 2020) treats object detection as a set prediction problem and proposes a concise end-to-end framework without complex hand-crafted anchor generation and post-processing. However, DETR suffers from significant slow training convergence and inefficient usage of high resolution features, which have sparked the following researches in detection transformers. For example, Deformable-DETR (Zhu et al. 2020) attends to sparse features to address above issues. DE-DETR (Wang et al. 2022) identifies the key factor that affects data efficiency is sparse feature sampling. DAB-DETR (Liu et al. 2022) uses dynamic anchor boxes as position queries in Transformer decoder to facilitate training. In comparison, in our study, we recast the query in point formulation to handle arbitrary-shape scene texts and speed up training.

Contour-based Text Detection and Spotting

From the perspective of modeling text contour, ABCNet (Liu et al. 2020b) predicts Bezier curve control points to adaptively fit arbitrarily-shaped texts for the first time. To enhance the capability to localize highly-curved texts, FCENet (Zhu et al. 2021) models text instances with Fourier contour fitting. In contrast, TextBPN and its extension (Zhang et al. 2021, 2022c) segment various probability maps and use them as priors to generate coarse boundary proposals, then iteratively refine boundary points with graph convolution or Transformer encoder. Considering that segmentation might be sensitive to noise, PCR (Dai et al. 2021a) proposes to progressively evolve the initial text proposal to arbitrarily shaped contours in a top-down manner in the convolution framework. More recently, inspired by detection transformers, FSG(Tang et al. 2022) samples a few representative features and uses Transformer encoder layers to implicitly group them, then predicts Bezier curve control points for localization. In comparison, TESTR (Zhang et al. 2022d) proposes a box-to-polygon scheme for text contour modeling, which utilizes box proposals from Transformer encoder as position queries to guide learnable control points content queries. We conjecture that the box information is coarse for point target in detection, which hinders efficient training. Hence, our work investigates the explicit and complete point query formulation in detection transformer framework.

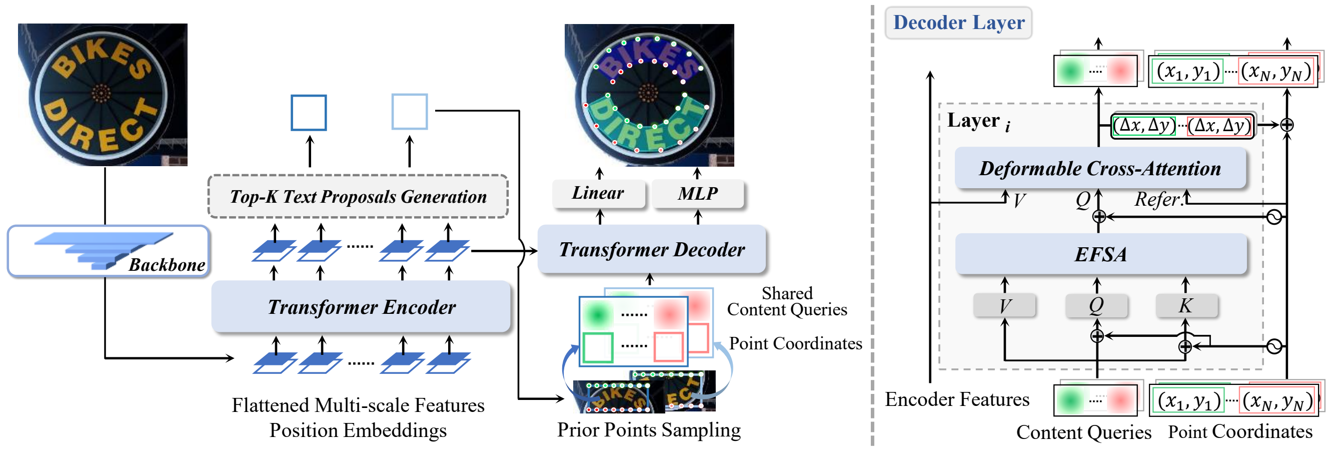


Figure 2: The architecture of DPText-DETR, which is built upon Deformable-DETR (Zhu et al. 2020), mainly consists of a CNN backbone, a Transformer encoder and decoder. Explicit points are calculated by Prior Points Sampling and encoded into positional queries. The point coordinates are progressively refined to form the final polygon predictions.

Methodology

This paper studies the scene text detection problem by developing an efficient Transformer-based decoder and investigating the influence of control point labels on the detection robustness. In more details, we propose an Explicit Point Query Modeling (including Prior Points Sampling and Point Update) method and an Enhanced Factorized Self-Attention module. In this section, we first briefly describe the overall pipeline and then detail the implementation.

Overview

The overall model architecture is illustrated in Fig. 2. In general, given a scene text image, we use a CNN backbone followed by a Transformer encoder to extract features. After the final encoder layer, multiple axis-aligned boxes are generated as proposals. With the center point and scale information of each box, a certain number of initial control point coordinates can be uniformly sampled on the top and bottom sides. In this way, these point coordinates can be used as suitable reference points for the deformable cross-attention module. In the decoder, point coordinates are encoded and added to corresponding control point content queries to form composite queries. The composite queries are firstly sent to EFSA to further mine their relative relationships and then fed into the deformable cross-attention module. Then control point coordinate prediction head is adopted to dynamically update the reference points layer-by-layer to better fit arbitrary-shape scene text. Finally, prediction heads are used to generate class confidence scores and N control point coordinates for each text instance. During training, we follow (Zhang et al. 2022d) to calculate losses for classification and control points. More details are described as follows.

Positional Label Form

The original label form shown in Fig. 1(b) is in line with human reading order. However, this form induces detector to implicitly learn the order, which increases the learning burden and confuses the model when the texts are in different orders during training. Moreover, even with sufficient rotation augmentations during training, it is difficult for the detector to correctly predict the reading order from visual features alone, as shown in Fig. 1(d).

To ease the difficulty, we present a positional label form to guide the detector to distinguish the top and bottom sides of scene text in a pure spatial sense without considering the concrete content of texts. As illustrated in Fig. 3, the positional label form mainly follows two simple rules: clockwise order and independent of text content. Specifically, we make the order of all original point labels in clockwise. If the original top side of text instance lies in the bottom position, the starting point is adjusted to the other side. When two sides are arranged left and right, if there is one side with a smaller minimum y value (origin in left-top), the starting point is adjusted to this side, otherwise, it is on the fixed default side.

Explicit Point Query Modeling

Prior Points Sampling. It is remarkable to transform axis-aligned box predictions into polygons that fit scene text with a concise yet effective operation, *i.e.*, the box-to-polygon scheme proposed by TESTR (Zhang et al. 2022d). Here, we briefly review this scheme. Concretely, after the final encoder layer, each anchor box provided by a top-K proposals generator is encoded, and then shared by N control point content queries. The resulting composite queries $Q^{(i)}(i = 1, \ldots, K)$ can be formulated as follows:

$$Q^{(i)} = P^{(i)} + C = \varphi((x, y, w, h)^{(i)}) + (p_1, \ldots, p_N), \quad (1)$$

where P and C represent the positional and the content part of each composite query, respectively. φ is the *sine* positional encoding function followed with a linear and normalization layer. (x, y, w, h) represents the center coordinate and scale information of each anchor box. $(p_1, \ldots, p_N)$ is the N learnable control point content queries shared across K composite queries. Note that we set the detector with the query formulation in Eq. (1) as our baseline. From Eq. (1),

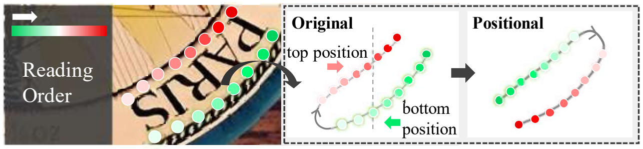


Figure 3: The original and positional label form. The points of each original label are sampled from the Bezier curves (Liu et al. 2020b) which fit the annotated polygon.

we can find that different control point content queries share the same anchor box prior information in each instance. Although the prior facilitates the prediction of control point positions, it mismatches the point targets to some extent. Content queries lack respective explicit position priors to exploit in the box sub-region.

Motivated by the positional label form and the shape prior that the top and bottom side of a scene text are usually close to the corresponding side on bounding box, we sample $\frac{N}{2}$ point coordinates $point_n (n = 1, \ldots, N)$ uniformly on the top and bottom side of each anchor box, respectively:

$$point_n = \begin{cases} (x - \frac{w}{2} + \frac{(n-1)\times w}{\frac{N}{2}-1}, y - \frac{h}{2}), & n \leq \frac{N}{2} \\ (x - \frac{w}{2} + \frac{(N-n)\times w}{\frac{N}{2}-1}, y + \frac{h}{2}), & n > \frac{N}{2} \end{cases} . \quad (2)$$

With $(point_1, \ldots, point_N)$, we can generate composite queries using the following complete point formulation:

$$Q^{(i)} = \varphi'((point_1, \ldots, point_N)^{(i)}) + (p_1, \ldots, p_N). \quad (3)$$

In this way, N control point content queries enjoy their respective explicit position prior, resulting in the superior training convergence.

Point Update. With the point coordinates, we can refine point positions layer-by-layer and use the updated positions as new reference points for deformable cross-attention. In comparison, TESTR directly adopts the anchor boxes information to generate position queries. Therefore, it is hard to perform refinement between decoder layers. Specifically, in our model, we update control points in each decoder layer after getting respective offsets $(\Delta x, \Delta y)$ by a prediction head, as illustrated in the decoder layer part of Fig. 2.

Discussion. We also notice that the very recent work BoundaryFormer (Lazarow, Xu, and Tu 2022) adopts a similar point query formulation in the instance segmentation task. BoundaryFormer aims at predicting polygons yet uses instance mask supervision. In BoundaryFormer, a fully convolutional detector (Ren et al. 2015; Tian et al. 2019) is exploited to predict object boxes. Next, a diamond is initialized for each box. Then, a Transformer decoder is used to refine the position of vertexes. Between decoder layers, new points are inserted between existing ones to produce fine polygons. In comparison, we aim to address the training concerned issue by modeling explicit and complete point queries. In our model, a fixed number of points are sampled on the top and bottom sides of each proposal box before the Transformer decoder, according to the shape prior that texts can be located with only two sides. The explicit point formulation enables the decoder to iteratively refine points for more precise final predictions in both BoundaryFormer and our model. However, in our DETR-based model, the explicit point query formulation is further explored to address the relatively slow convergence issue.

Enhanced Factorized Self-Attention

Following (Zhang et al. 2022d), we exploit Factorized Self-Attention (FSA) (Dong et al. 2021) in our baseline. In FSA, an intra-group self-attention (SA_{intra}) across N subqueries belonging to each of the $Q^{(i)}$ is firstly exploited to capture the relationship between different points within each text instance. After SA_{intra}, an inter-group self-attention (SA_{inter}) across K composite queries is adopted to capture the relationship between different instances. We conjecture that the non-local SA_{intra} falls short in capturing the circular shape prior of polygon control points. Hence, we leverage the local circular convolution (Peng et al. 2020) to complement FSA, forming the Enhanced Factorized Self-Attention. Concretely, SA_{intra} is firstly performed to get queries $Q_{intra} = SA_{intra}(Q)$, where keys are the same as Q while values exclude the positional part. Meanwhile, locally enhanced queries are generated: $Q_{local} = ReLU(BN(CirConv(Q)))$. Then, fused queries can be obtained: $Q_{fuse} = LN(FC(C + LN(Q_{intra} + Q_{local})))$, where C represents the content queries used as a shortcut, FC is a fully connected layer, BN is BatchNorm, and LN is LayerNorm. Next, the relationships between different instances are mined: $Q_{inter} = SA_{inter}(Q_{fuse})$. After that, Q_{inter} is sent to the deformable cross-attention module. Using one circular convolution layer with four-neighborhood achieves the best trade-off between performance and inference speed. We adopt this setting for experiments.

Experiments

We conduct experiments on three arbitrary-shape scene text benchmarks: Total-Text (Ch'ng, Chan, and Liu 2020), CTW1500 (Liu et al. 2019) and ICDAR19 ArT (Chng et al. 2019). Ablation studies are conducted on Total-Text to verify the effectiveness of each component of our methods.

Datasets

First, we briefly introduce the exploited datasets. **SynthText 150K** (Liu et al. 2020b) is a synthesized dataset for arbitrary-shape scene text, containing 94,723 images with multi-oriented text and 54,327 images with curved text. **Total-Text** (Ch'ng, Chan, and Liu 2020) consists of 1,255 training images and 300 test images. Word-level polygon annotations are provided. **Rot.Total-Text** is a test set derived from the Total-Text test set. Since the original label form induces model to generate unstable prediction as shown in Fig. 1(c), we apply large rotation angles (45°, 135°, 180°, 225°, 315°) on images of the Total-Text test set to examine the model robustness, resulting in 1,800 test images including the original test set. **CTW1500** (Liu et al. 2019) contains 1,000 training images and 500 test images. Text-line level annotations are presented. **ICDAR19 ArT** (Chng

Method	Backbone	Total-Text			CTW1500			ICDAR19 ArT		
		P	R	F	P	R	F	P	R	F
TextSnake (Long et al. 2018)	VGG16	82.7	74.5	78.4	67.9	85.3	75.6	–	–	–
PAN (Wang et al. 2019)	Res-18	89.3	81.0	85.0	86.4	81.2	83.7	–	–	–
CRAFT (Baek et al. 2019) †	VGG16	87.6	79.9	83.6	86.0	81.1	83.5	77.2	68.9	72.9
TextFuseNet (Ye et al. 2020) †	Res50	87.5	83.2	85.3	85.8	85.0	85.4	82.6	69.4	75.4
DB (Liao et al. 2020b)	Res50-DCN	87.1	82.5	84.7	86.9	80.2	83.4	–	–	–
PCR (Dai et al. 2021a)	DLA34	88.5	82.0	85.2	87.2	82.3	84.7	84.0	66.1	74.0
ABCNet-v2 (Liu et al. 2021a)	Res50	90.2	84.1	87.0	85.6	83.8	84.7	–	–	–
I3CL (Du et al. 2022)	Res50	89.2	83.7	86.3	87.4	84.5	85.9	82.7	71.3	76.6
TextBPN++ (Zhang et al. 2022c)	Res50	91.8	85.3	88.5	87.3	83.8	85.5	81.1	71.1	75.8
FSG (Tang et al. 2022)	Res50	90.7	85.7	88.1	88.1	82.4	85.2	–	–	–
TESTR-polygon (Zhang et al. 2022d)	Res50	93.4	81.4	86.9	92.0	82.6	87.1	–	–	–
SwinTextSpotter (Huang et al. 2022)	Swin	–	–	88.0	–	–	88.0	–	–	–
DPText-DETR (ours)	Res50	91.8	86.4	**89.0**	91.7	86.2	**88.8**	83.0	73.7	**78.1**

Table 1: Quantitative detection results on benchmarks. “P”, “R” and “F” denote Precision, Recall and F-measure, respectively. “†” means that the results on ICDAR19 ArT are collected from the official website (Chng et al. 2019).

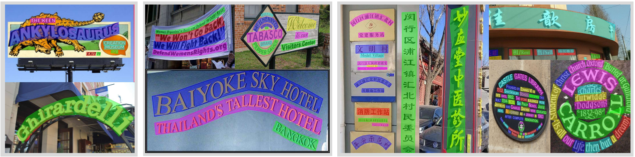


Figure 4: Qualitative results on Total-Text, CTW1500, and ICDAR19 ArT, from left to right.

et al. 2019) is a large arbitrary-shape scene text benchmark. It contains 5,603 training images and 4,563 test images.

Inverse-Text established in our work, consists of 500 test images. It is a arbitrary-shape scene text test set with about 40% inverse-like instances. A few instances are mirrored due to photographing. Some images are selected from existing benchmark test sets, *i.e.*, 121 images from ICDAR19 ArT, 7 images from Total-Text, and 3 images from CTW1500. Other images are collected from the Internet. Word-level polygon annotations are provided. Some samples are shown in Fig. 6.

Implementation Details

We adopt ResNet-50 (He et al. 2016) as the backbone. We use 8 heads for multi-head attention and 4 sampling points for deformable attention. The number of both encoder and decoder layers is set to 6. The composite queries number K is 100 and default control points number N is 16. We follow the hyper-parameter setting of loss used in the detection part of (Zhang et al. 2022d). Models are trained with 4 NVIDIA A100 (40GB) GPUs and tested with 1 GPU.

In ablation studies, we do not pre-train models to intuitively reveal the training convergence on Total-Text. We train models on Total-Text for 120k without rotation data augmentation and directly test them on Rot.Total-Text and Inverse-Text to verify the robustness. To help the model adapt to different text orders, we additionally rotate Total-Text training images with six angles ($-45°$, $-30°$, $-15°$, $15°$, $30°$, $45°$) representing normal cases, and rotate all normal cases for $180°$ representing inverse cases. When using rotation data, we train models for 200k iterations.

The complete training process is divided into two stages: pre-training stage and finetuning stage. The batch size is set to 8. For Total-Text and CTW1500, following (Zhang et al. 2022d; Liu et al. 2021a), the detector is pre-trained on a mixture of SynthText 150K, MLT (Nayef et al. 2019) and Total-Text for 350k iterations. The initial learning rate (lr) is 1×10^{-4} and is decayed to 1×10^{-5} at 280k. We finetune it on Total-Text for 20k iterations, with 5×10^{-5} lr which is divided by 10 at 16k. We adopt 13k finetuning iterations for CTW1500, with 2×10^{-5} lr. For ICDAR19 ArT, following (Du et al. 2022; Baek et al. 2020), we adopt LSVT (Sun et al. 2019) during pre-training. We use a mixture of SynthText 150K, MLT, ArT and LSVT to pre-train the model for 400k iterations. lr is 1×10^{-4} and is decayed to 1×10^{-5} at 320k. Then, we finetune it on ArT for 50k iterations, with 5×10^{-5} lr which is divided by 10 at 40k. We use the AdamW optimizer (Loshchilov and Hutter 2019) with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and weight decay of 10^{-4}. Data augmentation strategies such as random crop, random blur, brightness adjusting, and color change are applied. Note that rotation data mentioned above is only used in finetuning stage for each benchmark. We adopt multi-scale training strategy with the shortest edge ranging from 480 to 832, and the longest edge kept within 1600.

Comparison with State-of-the-art Methods

We test our method on Total-Text, CTW1500, and ICDAR19 ArT. Quantitative results compared with previous methods are presented in Tab. 1. Our method achieves consistent state-of-the-art performance. Compared with other detectors, for example, DPText-DETR outperforms TexBPN++ by 0.5%, 3.3%, and 2.3% in terms of F-measure on Total-

ID	Pos.Label	EPQM	EFSA	Rotation	Total-Text		Rot.Total-Text		Inverse-Text	
					F	FPS	F	FPS	F	FPS
1					83.90	**18.5**	70.02	**20.2**	77.63	**18.9**
2	✓				84.58	18.5	73.92	20.2	80.70	18.9
3		✓			85.15	17.9	71.28	19.7	79.44	18.5
4		✓	✓		85.86	17.3	72.60	18.9	80.33	18.3
5	✓	✓			85.28	17.9	74.87	19.7	81.56	18.5
6	✓	✓	✓		**86.17**	17.3	**74.99**	18.9	**81.99**	18.3
7				✓	84.98	**18.5**	83.99	**20.2**	84.28	**18.9**
8	✓			✓	86.07	18.5	84.52	20.2	86.69	18.9
9		✓		✓	86.16	17.9	84.15	19.7	83.79	18.5
10		✓	✓	✓	86.21	17.3	84.53	18.9	85.95	18.3
11	✓	✓		✓	86.46	17.9	84.86	19.7	86.57	18.5
12	✓	✓	✓	✓	**86.79**	17.3	**84.95**	18.9	**86.78**	18.3

Table 2: Ablations on test sets. "Pos.Label" denotes the positional label form. Without EFSA means the FSA is used instead.

Text, CTW1500, and ICDAR2019 ArT, respectively. Moreover, DPText-DETR leads I3CL by 2.7%, 2.9%, and 1.5% F-measure on the three benchmarks. Compared with FSG, our method achieves 0.9% and 3.6% higher F-measure on Total-Text and CTW1500. DPText-DETR also outperforms the state-of-the-art SwinTextSpotter by 1.0% and 0.8% in terms of F-measure on Total-Text and CTW1500. Some visual results are provided in Fig. 4. It shows that DPText-DETR performs well on straight, curve, and even dense long texts. A failure case is also shown, *i.e.*, the right bottom image in ICDAR19 ArT, where the polygon prediction is affected by extremely compact curved texts.

Ablation Studies

As mentioned before, pre-training is not used in all experiments of this subsection. Main ablation results are reported in Tab. 2. Notably, compared with previous pre-trained models, DPText-DETR without pre-training can still achieve competitive performance (F-measure: 86.79%).

Positional Label Form. As shown in Tab. 2, when the positional label form is used, the F-measure scores on all test sets are improved. For example, the comparison between the line 1 and line 2 in the table demonstrates that the F-measure is improved by 0.68% on Total-Text, 3.90% on Rot.Total-Text and 3.07% on Inverse-Text, which validates the effectiveness for model robustness. Moreover, positional label form can synergize better with rotation augmentation than the original form to improve the detection performance and robustness. When using rotation, the positional label form also contributes to faster convergence as shown in Fig. 5(a).

EPQM. In Tab. 2, we investigate the effectiveness of EPQM. EPQM intuitively boosts the performance and makes the major contribution to the convergence as shown in Fig. 5(a). Moreover, EPQM significantly enhances the few-shot learning ability. As shown in Tab. 3, when the training iterations and data volume are decreased, huge performance degradation of baseline models turns up, while the models with EPQM are far less affected.

EPQM	EFSA	TD-Ratio	Total-Text		Inverse-Text	
			F	Improv.	F	Improv.
		100%	73.92	–	70.90	–
✓		100%	82.99	9.07	78.18	7.28
✓	✓	100%	**83.66**	**9.74**	**79.09**	**8.19**
		50%	30.45	–	22.62	–
✓		50%	78.90	48.45	72.78	50.16
✓	✓	50%	**80.22**	**49.77**	**73.97**	**51.35**
		25%	14.94	–	6.98	–
✓		25%	58.54	43.6	52.32	45.34
✓	✓	25%	**70.49**	**55.55**	**60.15**	**53.17**

Table 3: Fewer iterations and training data test. The positional label form is adopted. "TD-Ratio": the training data ratio compared with the original one. "Improv.": the improvement on F-measure. In the first three rows, models are only trained for 12k iterations on Total-Text without rotation augmentation and directly tested on Inverse-Text. In the rest parts, we randomly sample training data according to TD-Ratio while keeping the equivalent training epochs as used in the first three rows. We train 6k iterations for the middle three models and 3k iterations for the last ones.

EFSA. In Tab. 2 and Tab. 3, we verify the effectiveness of EFSA. The comparison between line 5 and line 6 in Tab. 2 shows that EFSA can improve the F-measure by 0.89%. Tab. 3 shows that EFSA enables the model to learn better with fewer samples. For example, when the training data volume is 25%, compared with the model only equipped with EPQM, the model with both EPQM and EFSA achieves an extra gain of 11.95% F-measure on Total-Text and 7.83% F-measure on Inverse-Text. Moreover, as shown in Fig. 5(a), EFSA can further promote the training convergence and the model with all components achieves about six times faster convergence than the baseline in the initial training stage. We find EFSA is more effective when predicting polygon control points. Since Bezier curve control points do not al-

Method	Prior Points Sampling	Point Update	F
			83.90
Baseline	✓		84.13
	✓	✓	**85.15**

Table 4: Quantitative analysis on EPQM. The results are test on Total-Text without using positional label and EFSA.

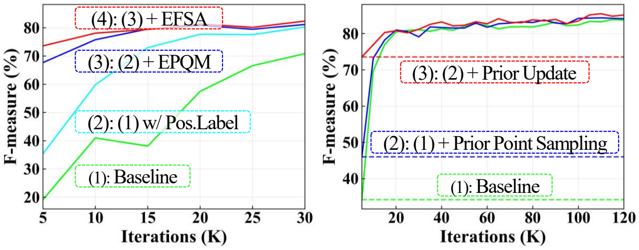


Figure 5: Convergence curves on Rot.Total-Text (left) and Total-Text (right).

ways form in circular shape and sometimes they are far apart, it is not suitable to combine circular convolution with self-attention for the Bezier variant.

In summary, the positional label form mainly improves the model robustness while EPQM and EFSA boost the overall performance, training convergence, and few-shot learning ability. DPText-DETR achieves ideal trade-off between performance gain and inference speed drop.

What Makes Faster Training Convergence?

We conduct further ablation studies on EPQM to reveal what makes convergence faster. Quantitative results and convergence curves are shown in Tab. 4 and Fig. 5(b). Referring to the blue curve in Fig. 5(b), the convergence at the initial stage is improved when only Prior Points Sampling is used. Referring to the red curve and Tab. 4, Point Update further boosts the convergence by a large margin and makes the major contribution to the performance. It demonstrates that the explicit position modeling for sparse points is the key to faster convergence. The explicit formulation is the prerequisite for dynamically updating points in decoder layers. Dynamic updating provides more precise reference points for deformable cross-attention, resulting in better performance. Prior works (Liu et al. 2022; Wang et al. 2022) have proved that box query formulation and sparse box area features extracted by ROIAlign can improve the training efficiency of DETR-based models. In our DPText-DETR designed for scene text detection, the Prior Points Sampling scheme can be regarded as a soft grid-sample operation, and it is also proved that the point query formulation, which is more sparse than the box, is more beneficial to training.

Further Discussion

In addition, we further investigate the performance of some arbitrary-shape scene text spotters on Inverse-Text. Recent methods can be roughly categorized into point-based and segmentation-based methods. For point-based methods, we select ABCNet-v2 (Liu et al. 2021a) and TESTR (Zhang et al. 2022d) that exploits dual Transformer decoders for parallel detection and recognition. For segmentation-based methods, we select SwinTextSpotter (Huang et al. 2022) as a representative. We finetune the official models trained on Total-Text with rotation augmentation as mentioned in implementation details for better adaptation to inverse-like texts. Results are reported in Tab. 5. For ABCNet-v2 and TESTR, we also test the influence of the positional label form. As shown in Tab. 5, the detection F-measures are improved when the positional label form is used, which validates the positive effect on detection.

Method	Det-F	End-to-End	
		None	Full
ABCNet-v2 (Our repro.)	78.0	57.2	69.5
ABCNet-v2 w/ Pos.Label (Our repro.)	87.2	62.2	**76.7**
TESTR (Our repro.)	86.8	62.1	74.7
TESTR w/ Pos.Label (Our repro.)	87.2	61.9	74.1
TESTR w/ Pos.Label (Our detector)	<u>87.3</u>	**63.1**	<u>75.4</u>
SwinTextSpotter (Our repro.)	**89.3**	<u>62.9</u>	74.7

Table 5: Results of spotters on Inverse-Text. “repro.” and “None” indicates our experiment using official released code and the end-to-end results without using lexicon.

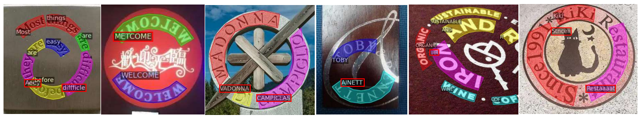


Figure 6: Qualitative results on Inverse-Text. Recognition failures on hard inverse-like texts are marked with red boxes.

We further replace the detection decoder of TESTR with ours, and find that the modified spotter still works well. It indicates that the detection decoder can iteratively refine control points with explicit point information while the recognition decoder remains to learn semantics from a coarse text anchor box sub-region. However, the modified spotter suffers from unsynchronized convergence between detection and recognition. We plan to explore a training efficient Transformer-based spotter in the future. Some visualizations are presented in Fig. 6.

Conclusion

We present a concise yet effective scene text detection transformer network, which transforms composite queries into explicit and complete point formulation. We investigate the effect of control point labels on model robustness and point out a practical positional label form. Extensive experiments demonstrate the state-of-the-art performance, training efficiency, and robustness of our proposed DPText-DETR. We also establish an Inverse-Text test set to facilitate future research in this area.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grants 62076186, 62141112, and 62225113, and in part by the Science and Technology Major Project of Hubei Province (Next-Generation AI Technologies) under Grant 2019AEA170. Dr. Jing Zhang is supported by the ARC project FL-170100117. The numerical calculations in this paper have been done on the supercomputing system in the Supercomputing Center of Wuhan University.

References

Baek, Y.; Lee, B.; Han, D.; Yun, S.; and Lee, H. 2019. Character region awareness for text detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9365–9374.

Baek, Y.; Shin, S.; Baek, J.; Park, S.; Lee, J.; Nam, D.; and Lee, H. 2020. Character region attention for text spotting. In *European Conference on Computer Vision*, 504–521. Springer.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-end object detection with transformers. In *European conference on computer vision*, 213–229. Springer.

Chng, C. K.; Liu, Y.; Sun, Y.; Ng, C. C.; Luo, C.; Ni, Z.; Fang, C.; Zhang, S.; Han, J.; Ding, E.; et al. 2019. ICDAR2019 robust reading challenge on arbitrary-shaped text-RRC-ArT. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, 1571–1576. IEEE.

Ch'ng, C.-K.; Chan, C. S.; and Liu, C.-L. 2020. Total-text: toward orientation robustness in scene text detection. *International Journal on Document Analysis and Recognition (IJDAR)*, 23(1): 31–52.

Dai, P.; Zhang, S.; Zhang, H.; and Cao, X. 2021a. Progressive contour regression for arbitrary-shape scene text detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 7393–7402.

Dai, X.; Chen, Y.; Yang, J.; Zhang, P.; Yuan, L.; and Zhang, L. 2021b. Dynamic detr: End-to-end object detection with dynamic attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2988–2997.

Dong, Q.; Tu, Z.; Liao, H.; Zhang, Y.; Mahadevan, V.; and Soatto, S. 2021. Visual relationship detection using part-and-sum transformers with composite queries. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3550–3559.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*.

Du, B.; Ye, J.; Zhang, J.; Liu, J.; and Tao, D. 2022. I3CL: Intra-and Inter-Instance Collaborative Learning for Arbitrary-shaped Scene Text Detection. *International Journal of Computer Vision*, 1–17.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

He, Y.; Chen, C.; Zhang, J.; Liu, J.; He, F.; Wang, C.; and Du, B. 2022. Visual semantics allow for textual reasoning better in scene text recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 888–896.

Huang, M.; Liu, Y.; Peng, Z.; Liu, C.; Lin, D.; Zhu, S.; Yuan, N.; Ding, K.; and Jin, L. 2022. SwinTextSpotter: Scene Text Spotting via Better Synergy between Text Detection and Text Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4593–4603.

Lazarow, J.; Xu, W.; and Tu, Z. 2022. Instance Segmentation With Mask-Supervised Polygonal Boundary Transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4382–4391.

Liao, M.; Lyu, P.; He, M.; Yao, C.; Wu, W.; and Bai, X. 2021. Mask TextSpotter: An End-to-End Trainable Neural Network for Spotting Text with Arbitrary Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(2): 532–548.

Liao, M.; Pang, G.; Huang, J.; Hassner, T.; and Bai, X. 2020a. Mask textspotter v3: Segmentation proposal network for robust scene text spotting. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, 706–722. Springer.

Liao, M.; Wan, Z.; Yao, C.; Chen, K.; and Bai, X. 2020b. Real-time scene text detection with differentiable binarization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 11474–11481.

Liu, J.; Chen, Z.; Du, B.; and Tao, D. 2020a. ASTS: A unified framework for arbitrary shape text spotting. *IEEE Transactions on Image Processing*, 29: 5924–5936.

Liu, S.; Li, F.; Zhang, H.; Yang, X.; Qi, X.; Su, H.; Zhu, J.; and Zhang, L. 2022. DAB-DETR: Dynamic Anchor Boxes are Better Queries for DETR. In *International Conference on Learning Representations*.

Liu, Y.; Chen, H.; Shen, C.; He, T.; Jin, L.; and Wang, L. 2020b. Abcnet: Real-time scene text spotting with adaptive bezier-curve network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9809–9818.

Liu, Y.; Jin, L.; Zhang, S.; Luo, C.; and Zhang, S. 2019. Curved scene text detection via transverse and longitudinal sequence connection. *Pattern Recognition*, 90: 337–345.

Liu, Y.; Shen, C.; Jin, L.; He, T.; Chen, P.; Liu, C.; and Chen, H. 2021a. ABCNet v2: Adaptive Bezier-Curve Network for Real-time End-to-end Text Spotting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–1.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021b. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10012–10022.

Long, S.; Ruan, J.; Zhang, W.; He, X.; Wu, W.; and Yao, C. 2018. Textsnake: A flexible representation for detecting text of arbitrary shapes. In *Proceedings of the European conference on computer vision (ECCV)*, 20–36.

Loshchilov, I.; and Hutter, F. 2019. Decoupled Weight Decay Regularization. In *ICLR*.

Meng, D.; Chen, X.; Fan, Z.; Zeng, G.; Li, H.; Yuan, Y.; Sun, L.; and Wang, J. 2021. Conditional detr for fast training convergence. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 3651–3660.

Nayef, N.; Patel, Y.; Busta, M.; Chowdhury, P. N.; Karatzas, D.; Khlif, W.; Matas, J.; Pal, U.; Burie, J.-C.; Liu, C.-l.; et al. 2019. ICDAR2019 robust reading challenge on multilingual scene text detection and recognition—RRC-MLT-2019. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, 1582–1587. IEEE.

Peng, S.; Jiang, W.; Pi, H.; Li, X.; Bao, H.; and Zhou, X. 2020. Deep snake for real-time instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8533–8542.

Qiao, L.; Chen, Y.; Cheng, Z.; Xu, Y.; Niu, Y.; Pu, S.; and Wu, F. 2021. Mango: A mask attention guided one-stage scene text spotter. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 2467–2476.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 28: 91–99.

Singh, A.; Natarajan, V.; Shah, M.; Jiang, Y.; Chen, X.; Batra, D.; Parikh, D.; and Rohrbach, M. 2019. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8317–8326.

Sun, Y.; Ni, Z.; Chng, C.-K.; Liu, Y.; Luo, C.; Ng, C. C.; Han, J.; Ding, E.; Liu, J.; Karatzas, D.; et al. 2019. ICDAR 2019 competition on large-scale street view text with partial labeling-RRC-LSVT. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, 1557–1562. IEEE.

Tang, J.; Zhang, W.; Liu, H.; Yang, M.; Jiang, B.; Hu, G.; and Bai, X. 2022. Few Could Be Better Than All: Feature Sampling and Grouping for Scene Text Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4563–4572.

Tian, Z.; Shen, C.; Chen, H.; and He, T. 2019. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9627–9636.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, 5998–6008.

Wang, W.; Cao, Y.; Zhang, J.; and Tao, D. 2021. Fp-detr: Detection transformer advanced by fully pre-training. In *International Conference on Learning Representations*.

Wang, W.; Xie, E.; Song, X.; Zang, Y.; Wang, W.; Lu, T.; Yu, G.; and Shen, C. 2019. Efficient and accurate arbitrary-shaped text detection with pixel aggregation network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8440–8449.

Wang, W.; Zhang, J.; Cao, Y.; Shen, Y.; and Tao, D. 2022. Towards data-efficient detection transformers. In *European Conference on Computer Vision*, 88–105. Springer.

Xu, Y.; Zhang, Q.; Zhang, J.; and Tao, D. 2021. Vitae: Vision transformer advanced by exploring intrinsic inductive bias. *Advances in Neural Information Processing Systems*, 34: 28522–28535.

Ye, J.; Chen, Z.; Liu, J.; and Du, B. 2020. TextFuseNet: Scene Text Detection with Richer Fused Features. In *IJCAI*, volume 20, 516–522.

Zhang, J.; and Tao, D. 2020. Empowering things with intelligence: a survey of the progress, challenges, and opportunities in artificial intelligence of things. *IEEE Internet of Things Journal*, 8(10): 7789–7817.

Zhang, P.; Xu, Y.; Cheng, Z.; Pu, S.; Lu, J.; Qiao, L.; Niu, Y.; and Wu, F. 2020. Trie: End-to-end text reading and information extraction for document understanding. In *Proceedings of the 28th ACM International Conference on Multimedia*, 1413–1422.

Zhang, Q.; Xu, Y.; Zhang, J.; and Tao, D. 2022a. Vitaev2: Vision transformer advanced by exploring inductive bias for image recognition and beyond. *arXiv preprint arXiv:2202.10108*.

Zhang, Q.; Xu, Y.; Zhang, J.; and Tao, D. 2022b. VSA: Learning Varied-Size Window Attention in Vision Transformers. In *European Conference on Computer Vision*. Springer.

Zhang, S.-X.; Zhu, X.; Yang, C.; Wang, H.; and Yin, X.-C. 2021. Adaptive Boundary Proposal Network for Arbitrary Shape Text Detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1305–1314.

Zhang, S.-X.; Zhu, X.; Yang, C.; and Yin, X.-C. 2022c. Arbitrary Shape Text Detection via Boundary Transformer. *arXiv preprint arXiv:2205.05320*.

Zhang, X.; Su, Y.; Tripathi, S.; and Tu, Z. 2022d. Text Spotting Transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9519–9528.

Zhou, Y.; Xie, H.; Fang, S.; Wang, J.; Zha, Z.; and Zhang, Y. 2021. TDI TextSpotter: Taking Data Imbalance into Account in Scene Text Spotting. In *Proceedings of the 29th ACM International Conference on Multimedia*, 2510–2518.

Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; and Dai, J. 2020. Deformable DETR: Deformable Transformers for End-to-End Object Detection. In *International Conference on Learning Representations*.

Zhu, Y.; Chen, J.; Liang, L.; Kuang, Z.; Jin, L.; and Zhang, W. 2021. Fourier contour embedding for arbitrary-shaped text detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 3123–3131.