This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Turning a CLIP Model into a Scene Text Detector

Wenwen Yu[*,1], Yuliang Liu[*,1], Wei Hua[1], Deqiang Jiang[2], Bo Ren[2], Xiang Bai[†,1]
[1]Huazhong University of Science and Technology [2]Tencent YouTu Lab
{wenwenyu,ylliu,whua_hust,xbai}@hust.edu.cn, {dqiangjiang,timren}@tencent.com

Abstract

The recent large-scale Contrastive Language-Image Pretraining (CLIP) model has shown great potential in various downstream tasks via leveraging the pretrained vision and language knowledge. Scene text, which contains rich textual and visual information, has an inherent connection with a model like CLIP. Recently, pretraining approaches based on vision language models have made effective progresses in the field of text detection. In contrast to these works, this paper proposes a new method, termed TCM, focusing on Turning the CLIP Model directly for text detection without pretraining process. We demonstrate the advantages of the proposed TCM as follows: (1) The underlying principle of our framework can be applied to improve existing scene text detector. (2) It facilitates the few-shot training capability of existing methods, e.g.*, by using 10% of labeled data, we significantly improve the performance of the baseline method with an average of 22% in terms of the F-measure on 4 benchmarks. (3) By turning the CLIP model into existing scene text detection methods, we further achieve promising domain adaptation ability. The code will be publicly released at https://github.com/wenwenyu/TCM.*

1. Introduction

Scene text detection is a long-standing research topic aiming to localize the bounding box or polygon of each text instance from natural images, as it has wide practical applications scenarios, such as office automation, instant translation, automatic driving, and online education. With the rapid development of fully-supervised deep learning technologies, scene text detection has achieved remarkable progresses. Although supervised approaches have made remarkable progress in the field of text detection, they require extensive and elaborate annotations, *e.g.*, character-level, word-level, and text-line level bounding boxes, especially polygonal boxes for arbitrarily-shaped scene text. Therefore, it is very important to investigate text detection methods under small amount of labeled data, *i.e.*, few-shot training.

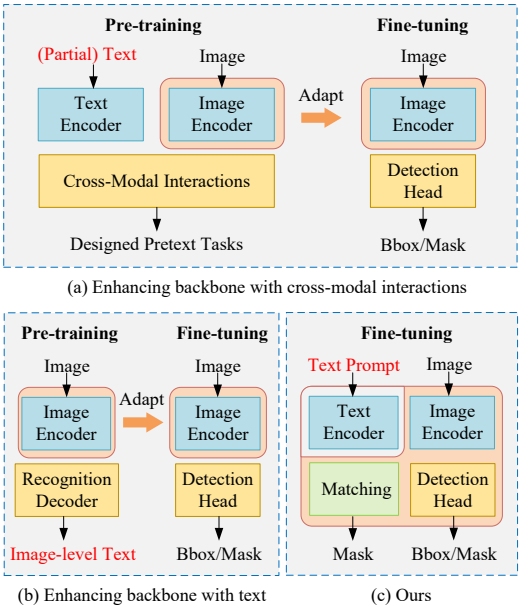


Figure 1. Comparisons of different paradigms of using text knowledge for scene text detection.

Recently, through leveraging the pretrained vision and language knowledge, the large-scale Contrastive Language-Image Pretraining (CLIP) model [26] has demonstrated its significance in various downstream tasks. *e.g.*, image classification [53], object detection [5], and semantic segmentation [12, 27, 43].

Compared to general object detection, scene text in natural images usually presents with both visual and rich character information, which has a natural connection with the CLIP model. Therefore, how to make full use of cross-modal information from visual, semantic, and text knowledge to improve the performance of the text detection models receives increasing attentions in recent studies. For examples, Song *et al.* [29], inspired by CLIP, adopts fine-grained cross-modality interaction to align unimodal embeddings for learning better representations of backbone via carefully designed pretraining tasks. Xue *et al.* [46] presents a weakly supervised pretraining method to jointly learn and align vi-

[*]Equal contribution. [†]Corresponding author.

sual and partial textual information for learning effective visual text representations for scene text detection. Wan *et al.* [35] proposes self-attention based text knowledge mining to enhance backbone via an image-level text recognition pretraining tasks.

Different from these works, as shown in Figure 1, this paper focuses on turning the CLIP model for text detection without pretraining process. However, it is not trivial to incorporate the CLIP model into a scene text detector. The key is seeking a proper method to exploit the visual and semantic prior information conditioned on each image. In this paper, we develop a new method for scene text detection, termed as TCM, short for **T**urning a **C**LIP **M**odel into a scene text detector, which can be easily plugged to improve the scene text detection frameworks. We design a cross-modal interaction mechanism through visual prompt learning, which is implemented by cross-attention to recover the locality feature from the image encoder of CLIP to capture fine-grained information to respond to the coarse text region for the subsequent matching between text instance and language. Besides, to steer the pretrained knowledge from the text encoder conditioned independently on different input images, we employ the predefined language prompt, learnable prompt, and a language prompt generator using simple linear layer to get global image information. In addition, we design an instance-language matching method to align the image embedding and text embedding, which encourages the image encoder to explicitly refine text regions from cross-modal visual-language priors. Compared to previous pretraining approaches, our method can be directly finetuned for the text detection task without pretraining process, as elaborated in Fig. 1. In this way, the text detector can absorb the rich visual or semantic information of text from CLIP. We summarize the advantages of our method as follows:

- We construct a new text detection framework, termed as TCM, which can be easily plugged to enhance the existing detectors.
- Our framework can enable effective few-shot training capability. Such advantage is more obvious when using less training samples compared to the baseline detectors. Specifically, by using 10% of labeled data, we improve the performance of the baseline detector by an average of 22% in terms of the F-measure on 4 benchmarks.
- TCM introduces promising domain adaptation ability, *i.e.*, when using training data that is out-of-distribution of the testing data, the performance can be significantly improved. Such phenomenon is further demonstrated by a NightTime-ArT text dataset[1], which we collected from the ArT dataset.
- Without pretraining process using specific pretext tasks, TCM can still leverage the prior knowledge from the CLIP model, outperforming previous scene text pretraining methods [29, 35, 46].

[1] NightTime-ArT Download Link

2. Related works

Unimodal Scene Text Detection. Unimodal scene text detection represents the method directly adopts the bounding boxes annotation only [20]. It can be roughly divided into two categories: Segmentation-based methods and regression-based methods. The segmentation-based methods usually conduct pixel-level [13, 16, 17, 33, 37, 41, 45], segment-level [1, 22, 28, 30, 32, 44, 48, 51], or contour-level [36, 39] segmentation, then grouping segments into text instances via post-processing. The regression-based methods [7–9, 14, 38, 50, 52, 55, 57] regards text as a whole object and regress the bounding boxes of the text instances directly.

Cross-modal Assisted Scene Text Detection. Unlike unimodal based scene text detection, cross-modal assisted scene text detection aims to make full use of cross-modal information including visual, semantic, and text knowledge to boost the performance. Wan *et al.* [35] utilized an image-level text recognition pretraining tasks to enhance backbone via the proposed self-attention based text knowledge mining mechanism. Song *et al.* [29], inspired by CLIP, designed three pretraining fine-grained cross-modality interaction tasks to align unimodal embeddings for learning better representations of backbone. Xue *et al.* [46] jointly learned and aligned visual and partial text instances information for learning effective visual text representations via the proposed weakly supervised pretraining method. Long *et al.* [21] proposed an end-to-end model to perform unified scene text detection and visual layout analysis simultaneously. The above methods explicitly leverage text or visual information to assist text detection. Instead, our method focuses on improving the performance results by turning a CLIP model into a scene text detector via leveraging pretrained text knowledge.

3. Methodology

We begin by illustrating the CLIP model which we used for fetching the prior knowledge. Next, we introduce the technical details of TCM as well as the rationale behind it. An overview of our approach is shown in Fig. 2.

3.1. Contrastive Language-Image Pretraining

CLIP [26], which collects 400 million image-text pairs without human annotation for model pretraining, has well demonstrated the potential of learning transferable knowledge and open-set visual concepts. Previous study [4] shows

that different neurons in CLIP model can capture the corresponding concept literally, symbolically, and conceptually, As shown in Fig. 4, the CLIP model is an inborn text-friendly model which can effectively abstract the mapping space between image and text [25]. During training, CLIP learns a joint embedding space for the two modalities via a contrastive loss. Given a batch of image-text pairs, for each image, CLIP maximizes the cosine similarity with the matched text while minimizing that with all other unmatched text. For each text, the loss is computed similarly as each image. In this way, CLIP can be used for zero-shot image recognition [53]. However, to exploit the relevant information from such a model, there are two prerequisites: 1) A proper method to effectively request the prior knowledge from the CLIP. 2) The original model can only measure the similarity between an integrated image and a single word or sentence. For scene text detection, there are usually many text instances per image, which are all required to be recalled equivalently.

3.2. Turning a CLIP into a Text Detector

To turn the CLIP model into the scene text detector, we propose TCM, as shown in Fig. 2 and Fig.3. TCM is a pluggable module that can be directly applied to enhance the existing scene text detectors. It extracts the image and text embeddings from the image encoder and text encoder of CLIP model, respectively. We then design a cross-modal interaction mechanism through visual prompt learning to recover the locality feature from the image encoder of CLIP, which can capture fine-grained information to respond to the coarse text region for the subsequent matching between text instance and language. For better steering the pretrained knowledge, we introduce a language prompt generator to generate conditional cue for each image and design a visual prompt generator that learns image prompts for adapting the frozen clip text encoder for the text detection task. The TCM can be directly applicable to broader text detection methods only with some minor modifications. In addition, we design an instance-language matching method to align the image embedding and text embedding, which encourages the image encoder to explicitly refine text regions from cross-modal visual-language priors.

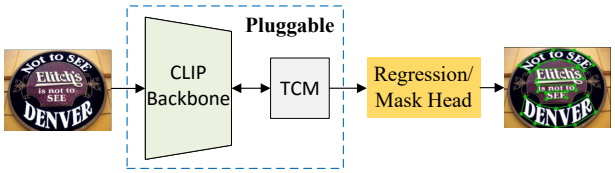


Figure 2. The overall framework of our approach.

Image Encoder. We use the pretrained ResNet50 [6] of CLIP as the image encoder, which produces an embedding vector for every input pixel. Given the input image

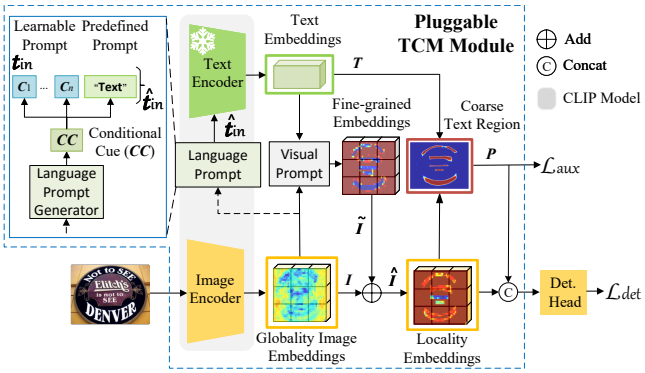


Figure 3. The details of the TCM. The image encoder and text encoder are directly from the CLIP model. Det. Head short for detection head.

$\boldsymbol{I}' \in \mathbb{R}^{H\times W\times 3}$, image encoder outputs image embedding $\boldsymbol{I} \in \mathbb{R}^{\tilde{H}\times\tilde{W}\times C}$, where $\tilde{H} = \frac{H}{s}$, $\tilde{W} = \frac{W}{s}$, and C is the image embedding dimension (C is set to 1024) and s is the downsampling ratio (s is empirically set to 32), which can be expressed as:

$$\boldsymbol{I} = \text{ImageEncoder}(\boldsymbol{I}')\,. \tag{1}$$

Text Encoder. The text encoder takes input a number of of K classes prompt and embeds it into a continuous vector space $\mathbb{R}^C$, producing text embeddings $\boldsymbol{T} = \{\boldsymbol{t}_1, \ldots, \boldsymbol{t}_K\} \in \mathbb{R}^{K\times C}$ as outputs of the text encoder, where $\boldsymbol{t}_i \in \mathbb{R}^C$. Specifically, we leverage the frozen pretrained text encoder of CLIP throughout as the text encoder can provide language knowledge prior for text detection. K is set to 1 because there is only one text class in text detection task. Different from the original model that uses templates like "a photo of a [CLS].", we predefine discrete language prompt as "*Text*". Then, a part of the text encoder input $\boldsymbol{t}'_{in}$ is defined as follows:

$$\boldsymbol{t}'_{in} = \text{WordEmbedding}(\text{Text}) \in \mathbb{R}^{\text{D}}\,, \tag{2}$$

where $\text{WordEmbedding}(\cdot)$ denotes word embedding for predefined prompt "Text" class. D is the word embedding dimension and set to 512.

Inspired by CoOp [53, 54], we also add learnable prompt $\{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n\}$ to learn robust transferability of text embedding for facilitating zero-shot transfer of CLIP model, where n is the number of learnable prompt, which is set to 4 by default, and $\boldsymbol{c}_i \in \mathcal{R}^D$. Thus, the input $\boldsymbol{t}_{in}$ of the text encoder is as follows:

$$\boldsymbol{t}_{in} = [\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n, \boldsymbol{t}'_{in}] \in \mathbb{R}^{(n+1)\times D}\,. \tag{3}$$

The text encoder takes $\boldsymbol{t}_{in}$ as input and generates text embedding $\boldsymbol{T} = \{\boldsymbol{t}_1\} \in \mathbb{R}^C$, and $\boldsymbol{T}$ is donated by $\boldsymbol{t}_{out} \in \mathcal{R}^C$ for simplification:

$$\boldsymbol{t}_{out} = \text{TextEncoder}(\boldsymbol{t}_{in}) \in \mathbb{R}^C\,. \tag{4}$$

Figure 4. The neurons in the clip model can directly respond to the text. The source images are from [4].

Language Prompt Generator. Although the predefined prompt and learnable prompt are effective for steering the CLIP model, it may suffer from limited few-shot or generalization ability to open-ended scenarios where the testing text instance is out-of-distribution from the training images. To this end, we present a language prompt generator to generate a feature vector, termed as conditional cue ($\boldsymbol{cc}$). For each image, the $\boldsymbol{cc}$ is then combined with the input of the text encoder $\boldsymbol{t}_{in}$, formulated as follows:

$$\hat{\boldsymbol{t}}_{in} = \boldsymbol{cc} + \boldsymbol{t}_{in} \in \mathbb{R}^{(n+1)\times D}, \tag{5}$$

where $\hat{\boldsymbol{t}}_{in}$ is the new prompt input of the text encoder conditioned on the input image, and we replace $\boldsymbol{t}_{in}$ with $\hat{\boldsymbol{t}}_{in}$ in Eq. 4.

In practice, the language prompt generator is built with a two-layer feed-forward network, which is applied to generate conditional cue ($\boldsymbol{cc}$) from the globality image embedding $\boldsymbol{I}$. This consists of two layer normalization followed by linear transformations, with a ReLU activation in between, which is formulated as follows:

$$\boldsymbol{cc} = \mathrm{LN}(\sigma(\mathrm{LN}(\bar{\boldsymbol{I}})\boldsymbol{W}_1 + \boldsymbol{b}_1))\boldsymbol{W}_2 + \boldsymbol{b}_2 \in \mathbb{R}^{D}, \tag{6}$$

where $\bar{\boldsymbol{I}} \in \mathbb{R}^{\mathrm{C}}$ is the global image-level feature generated from image embedding $\boldsymbol{I}$ by the same global attention pooling layer as in CLIP. $\boldsymbol{W}_1 \in \mathbb{R}^{C\times C}$, $\boldsymbol{W}_2 \in \mathbb{R}^{C\times D}$, $\boldsymbol{b}_1 \in \mathbb{R}^{C}$, $\boldsymbol{b}_2 \in \mathbb{R}^{D}$, and we broadcast $\boldsymbol{cc}$ with $\boldsymbol{t}_{in}$ to get $\hat{\boldsymbol{t}}_{in}$ in Eq. 5.

Visual Prompt Generator. We design a visual prompt generator to adaptively propagate fine-grained semantic information from textual features to visual features. Formally, we use the cross-attention mechanism in Transformer [34] to model the interactions between image embedding ($\boldsymbol{Q}$) and text embedding ($\boldsymbol{K}$, $\boldsymbol{V}$). The visual prompt $\tilde{\boldsymbol{I}}$ is then learned for transferring the information prior from image-level to text instance-level, which is defined as:

$$\tilde{\boldsymbol{I}} = \mathrm{TDec}(Q = \boldsymbol{I}, K = \boldsymbol{t}_{out}, V = \boldsymbol{t}_{out}) \in \mathbb{R}^{\tilde{H}\times\tilde{W}\times C}, \tag{7}$$

where TDec denotes the Transformer Decoder.

Based on the conditional visual prompt, the original image embedding $\boldsymbol{I}$ is equipped with $\tilde{\boldsymbol{I}}$ to produce the prompted text-aware locality embeddings $\hat{\boldsymbol{I}}$ used for instance-language matching (Eq. 9) and downstream detection head:

$$\hat{\boldsymbol{I}} = \boldsymbol{I} + \tilde{\boldsymbol{I}}. \tag{8}$$

Instance-language Matching. Given the output of the text encoder and image encoder, we perform text instance-language matching alignment on text-aware locality image embedding $\hat{\boldsymbol{I}}$ and text embedding $\boldsymbol{t}_{out}$ by the dot product followed by sigmoid activation to get binary score map. The mixture of the generated conditional fine-grained embedding $\tilde{\boldsymbol{I}}$ and visual embedding $\boldsymbol{I}$ can allow text instance existing in visual features to be better matched with pretrained language knowledge in collaboration. The matching mechanism is formulated as follows:

$$\boldsymbol{P} = \mathrm{sigmoid}(\hat{\boldsymbol{I}}\boldsymbol{t}_{out}^{T}/\tau) \in \mathbb{R}^{\tilde{H}\times\tilde{W}\times 1}, \tag{9}$$

where $\boldsymbol{t}_{out}$ is text embedding because of only one text class in text detection scenarios, and $\boldsymbol{P}$ is the binary text segmentation map. The segmentation maps are supervised using the ground-truths as an auxiliary loss and concatenated by the prompted embedding $\hat{\boldsymbol{I}}$ for downstream text detection head to explicitly incorporate language priors for detection. During training, we minimize a binary cross-entropy loss between the segmentation map $\boldsymbol{P}$ and ground-truth, which is defined as follows:

$$\mathcal{L}_{aux} = \sum_{i}^{\tilde{H}}\sum_{j}^{\tilde{W}} y_{ij}\log(P_{ij}) + (1-y_{ij})\log(1-P_{ij}), \tag{10}$$

where y_{ij} and P_{ij} are the label and predicted probability of pixel (i, j) belonging to the text instances, respectively.

Optimization. The loss function $\mathcal{L}_{total}$ is the sum of detection loss $\mathcal{L}_{det}$ and auxiliary loss $\mathcal{L}_{aux}$, formulated as follows:

$$\mathcal{L}_{total} = \mathcal{L}_{det} + \lambda\mathcal{L}_{aux}, \tag{11}$$

where λ is a trade-off hyper-parameters and set to 1 in this paper. $\mathcal{L}_{det}$ depends on downstream text detection method including segmentation and regression categories. In the inference period, we use the output of the detection head as the final result.

4. Experiments

We conduct four sets of experiments to validate TCM. Our first set of experiment examines how TCM can be incorporated into existing text detectors to achieve consistent performance improvements. Next, we demonstrate the few-shot training capability and generalization ability by incorporating the TCM method. In the third set of experiments, we compare our method with previous pretraining methods. Finally, we provide thorough experiments to evaluate the sensitivity w.r.t. the proposed designs.

Datasets. Our experiments are conducted on a number of commonly known scene text detection benchmarks including ICDAR2013 (IC13) [11], ICDAR2015 (IC15) [10],

MSRA-TD500 (TD) [47], CTW1500 (CTW) [19], Total-Text (TT) [3], ArT [2], MLT17 [24], and MLT19 [23]. More details of the datasets refer to appendix.

Evaluation Metric. We use intersection over union (IoU) to determine whether the model correctly detects the region of text, and we calculate precision (P), recall (R), and F-measure (F) for comparison following common practice [11]. For fair comparisons, text regions labeled with either "do not care" or "###" will be ignored in all datasets during training and testing.

Implementation Details. For text detection tasks, we experiment with the popular text detection methods including DBNet (DB) [17][2], PAN [37][3], and FCENet (FCE) [57][4] to evaluate TCM. For consistent settings with these methods, we train the detector using both SynthText and the real datasets. Specifically, the backbone is instantiated with the pretrained image encoder ResNet50 [6] of the CLIP unless specified. The visual prompt generator has 3 transformer decoder layers with 4 heads; transformer width is 256; and the feed-forward hidden dimension is set to 1024. We use the corresponding detection head of the DBNet, PAN, and FCENet to predict the final results. For testing few-shot learning of model, we directly train on the benchmark with different proportions of training data without pretraining and test it on the corresponding test data. For testing the generalization ability, we use the model trained on the corresponding source datasets and evaluating it on the target dataset that has dissimilar distribution. We consider two kinds of adaptation including synthtext-to-real and real-to-real, to validate the domain adaptation of the TCM. The ablation studies are conducted w.r.t. the predefined prompt, the learnable prompt, the language prompt generator, the visual prompt generator, and the different settings. The DBNet is used as baseline for TCM.

4.1. Cooperation with Existing Methods

We report the text detection results of our TCM combined with three text detection methods on IC15, TD, and CTW in Table 1. Our method is +0.9%, +1.7%, and +1.9% higher than the original FCENet, PAN, and DBNet, respectively, in terms of F-measure on IC15. TD and CTW also have similar consistent improvement. Note that the inference speed of our method is 18, 8.4, and 10 FPS evaluated on IC15, TD, and CTW datasets, respectively, with PAN, FCENet, and DBNet, remaining the high efficiency of the detector.

We visualize our method in Fig. 7. It shows that the fine-grained features $\tilde{\boldsymbol{I}}$ containing text information is recovered from the global image embedding $\boldsymbol{I}$, demonstrating that TCM can identify text regions and provide this prior cues for downstream text detection.

[2] https://github.com/MhLiao/DB

[3] https://github.com/whai362/pan_pp.pytorch

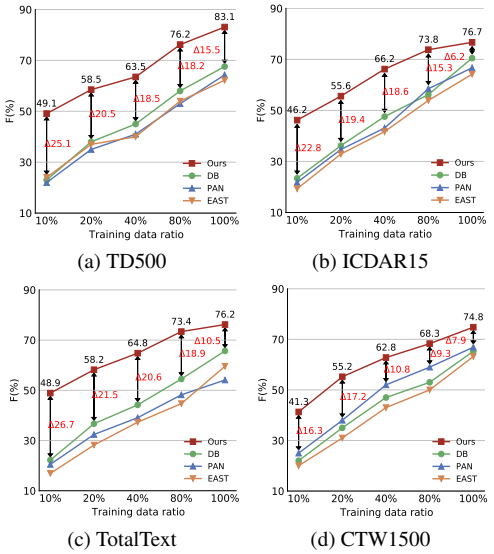
[4] https://github.com/open-mmlab/mmocr/tree/main/configs/textdet/fcenet

	Method	IC15		TD		CTW		FPS
		F	Δ	F	Δ	F	Δ	
Reg.	FCENet [57]	86.2	-	85.4†	-	85.5	-	11.5
	TCM-FCENet	**87.1**	**+0.9**	**86.9**	**+1.5**	**85.9**	**+0.4**	8.4
Seg.	PAN [37]	82.9	-	84.1	-	83.7	-	36
	TCM-PAN	**84.6**	**+1.7**	**85.3**	**+1.2**	**84.3**	**+0.6**	18
	DBNet [17]	87.3	-	84.9	-	83.4	-	14.5
	TCM-DBNet	**89.2**	**+1.9**	**88.8**	**+3.9**	**84.9**	**+1.5**	10

Table 1. Text detection results of cooperating with existing methods on IC15, TD, and CTW. † indicates the results from [49]. Reg. and Seg. short for regression and segmentation methods, respectively. FPS are reported with ResNet50 backbone on a single V100.

4.2. Few-shot Training Ability

To further verify the few-show training ability of our method, we directly train our model on real datasets using various training data ratio without pretraining, and evaluate it on the corresponding 4 benchmarks. As shown in Fig. 5, our method shows robust on limited data and outperforms the three baseline methods including DB, PAN and EAST [55]. The results show that the TCM can capture the inherent characteristic of text via leveraging the pretrained vision and language knowledge of the zero-shot trained CLIP model.



(a) TD500 (b) ICDAR15

(c) TotalText (d) CTW1500

Figure 5. Few-shot training ability with varying training data ratio. "F" represents F-measure.

Method	ST → IC13	ST → IC15
EAST† [55]	67.1	60.5
PAN [37]	-	54.8
CCN [42]	-	65.1
ST3D [15]	73.8	67.6
DBNet [17]	71.7	64.0
TCM-DBNet	**79.6**	**76.7**

Table 2. Synthtext-to-real adaptation. † indicates the results from [40]. ST indicates SynthText. F-measure (%) is reported.

4.3. Generalization Ability

We conduct two types of experiments including synthtext-to-real adaptation and real-to-real adaptation, as shown in Table 2 and Table 3, respectively. From the tables, we can see that by plugging the TCM to DBNet, we significantly improve the performance by an average of 8.2% in terms of F-measure for four different settings including synthtext-to-real and real-to-real, which further demonstrates the effectiveness of our method for domain adaptation.

Method	IC13 → IC15	IC13 → TD
EAST† [55]	53.3	46.8
GD(AD) [49]	64.4	58.5
GD(10-AD) [49]	69.4	62.1
CycleGAN [56]	57.2	-
ST-GAN [18]	57.6	-
CycleGAN+ST-GAN	60.8	-
TST [40]	52.4	-
DBNet [17]	63.9	53.8
TCM-DBNet	**71.9**	**65.1**

Table 3. Real-to-real adaptation. † indicates that the results are from [49]. Note that the proposed method outperforms other methods. F-measure (%) is reported.

4.4. Comparison with Pretraining Methods

The pretraining methods based on specifically designed pretext tasks has made effective progress in the field of text detection. In contrast to these efforts, TCM can turn the CLIP model directly into a scene text detector without pretraining process. The comparison results are shown in Table 4, from which we can see that without pretext tasks for pretraining, DB+TCM consistently outperforms previous methods including DB+STKM [35], DB+VLPT [29], and DB+oCLIP [46]. Especially on IC15, our method outperforms previous state-of-the-art pretraining method by a large margin, with 89.4% versus 86.5% in terms of the F-measure.

	Methods	Pretext task	IC15	TT	TD	CTW
Convention	SegLink [28]	×	-	-	77.0	-
	PSENet-1s [13]	×	85.7	80.9	-	82.2
	LOMO [50]	×	87.2	81.6	-	78.4
	MOST [7]	×	88.2	-	86.4	-
	Tang *et al.* [31]	×	89.1	-	88.1	-
VLP	DB+ST†	×	85.4	84.7	84.9	-
	DB+STKM† [35]	✓	86.1	85.5	85.9	-
	DB+VLPT† [29]	✓	86.5	86.3	88.5	-
	DB+oCLIP* [46]	✓	-	-	-	84.4
	DB+TCM(Ours)	×	**89.4**	85.9	**88.8**	**85.1**

Table 4. Comparison with existing scene text pretraining techniques on DBNet (DB). † indicates the results from [29]. ST and VLP denote SynthText pretraining and visual-language pretraining methods, respectively. * stand for our reimplementation results. F-measure (%) is reported.

4.5. Ablation Studies

Pretrained CLIP Backbone. First, we conduct experiments that we only replace the original backbone of the DBNet with the pretrained image encoder ResNet50 of the CLIP to quantify the performance variance of the backbones. As shown in Table 5, the original pretrained model of CLIP is insufficient for leveraging the visual-language knowledge of the CLIP. Therefore, it is necessary to use a proper method to excavate the knowledge of the CLIP model.

Method	BB	IC15	TD	TT	CTW
DBNet	R50	87.3	84.9	84.7	83.4
DBNet	CR50	87.7 (+0.4)	86.8 (+1.9)	84.7	83.4

Table 5. Ablation study of the ResNet50 backbone on IC15, TD, TT, and CTW. BB indicates Backbone. R50 and CR50 represent the ResNet50 backbones of the DBNet and the CLIP, respectively. F-measure (%) is reported.

Ablation Study for the Predefined Prompt. When using the predefined prompt, as illustrated in the second row of Table 6, the performances are slightly improved on all four datasets (IC15, TD, TT, and CTW), with 0.05%, 0.2%, 0.04%, and 0.1% higher than the baseline method, respectively.

Ablation Study for the Learnable Prompt. Besides, results combing the learnable prompt with the predefined prompt on four datasets are provided in the third row of Table 6. We notice that a consistent improvement can be achieved by adding the learnable prompt. We also show the influence of using different numbers of the learnable prompt in row 4 to row 6 of Table 6. We observe that as the value of the number of the learnable prompt increases, the performance increases gradually on all datasets. Compared to the value 4, the value 32 obtains obvious improvements on CTW, TD, and TT. We conjecture that this is because the larger number

Method	PP	LP	LG	VG	IC15	TD	TT	CTW
					F	F	F	F
BSL	×	×	×	×	87.7	86.8	84.7	83.4
BSL+	✓	×	×	×	87.75	87.0	84.74	83.5
BSL+	✓	4	×	×	88.0	87.1	84.8	83.6
BSL+	×	4	×	×	87.8	87.7	85.1	83.9
BSL+	×	18	×	×	88.1	87.8	85.3	83.9
BSL+	×	32	×	×	88.4	88.2	85.4	84.5
BSL+	✓	4	✓	×	88.6	88.4	85.5	84.6
TCM	✓	4	✓	✓	89.2	88.8	85.6	84.9
TCM	✓	32	✓	✓	**89.4**	**88.8**	**85.9**	**85.1**
Δ					+1.7	+2.0	+1.2	+1.7

Table 6. Ablation study of our proposed components on IC15, TD, TT and CTW. "BSL", "PP", "LP", "LG", and "VG" represent the baseline method DBNet, the predefined prompt, the learnable prompt, the language prompt generator, and the visual prompt generator, respectively. F (%) represents F-measure. Δ represents the variance.

of the learnable prompt can better steer the pretrained text encoder knowledge which is useful for text detection. In the following experiments, the default number of the learnable prompt is set to 4 for simplicity.

Ablation Study for the Language Prompt Generator. Furthermore, we evaluate the performance of the proposed language prompt generator shown in 7_{th} row of Table 6. With the help of the language prompt generator, we find that TCM achieves further improvements on all four datasets, especially on ICDAR2015, indicating that the conditional cue generated by the language prompt generator for each image can ensure better generalization over different types of datasets.

Ablation Study for the Visual Prompt Generator. Finally, combining the proposed visual prompt generator with the above other components, the improvement of F-measure is better than the baseline on all four datasets, with larger margins of 1.7% and 2.0% on IC15 and TD, respectively. The reason for this obvious complementary phenomenon is that the visual prompt generator can propagate fine-grained visual semantic information from textual features to visual features. Besides, the prompted locality image embedding generated by the visual prompt generator can guide the model to obtain more accurate text instance-level visual representations, which boosts the ability of instance-language matching and generates a precise segmentation score map that is useful for downstream detection head.

Ablation Study for the VG and LG on Generalization Performance. As described in Table 7, removing the VG and LG elements from TCM dramatically deteriorates the generalization performance, which further indicates the effectiveness of the VG and LG.

Ablation Study for Image Encoder and Text Encoder. We have investigated how the quality of the frozen text encoder and image encoder affects the performance via adjusting the corresponding learning rate (LR) factor. The experimental results of TCM-DBNet on the TD500 dataset are shown in Table 8. The results show that using a lower learning rate for both encoders and fixing the text encoder is the optimal setting for training the whole model. Note that we observe performance degradation when directly using $1.0\times$ learning rate for both encoders, which suggests the frozen text encoder can stabilize the training process. The cores of the architecture, including the language prompt generator and visual prompt generator, are designed to better steer knowledge of the pretrained CLIP. Appropriate design of the network architecture and the use of the pretrained CLIP are complementary.

Ablation Study for Different Amount of Data. To further explore whether the TCM can learn the additional knowledge which is hard to be obtained from increasing data, we have trained the model on a large-scale public joint data including IC13, IC15, TD, CTW, TT, and MLT17, with total 13,784 image, and testing it on a NightTime-ArT data (326 images) carefully collected from ArT. The nighttime examples of ArT are shown in Fig. 6. Results are shown in Table 9. The results show that even with the addition of large amounts of training data, existing methods still show limitation to the nighttime data that is obviously out-of-distribution from the training set. However, TCM can still perform robust in such case, indicating its irreplaceable potential generalization ability.

Figure 6. The examples of our constructed NightTime-ArT.

Ablation Study for the Parameters Comparison. For a fair comparison, we have increased the parameters of DBNet by replacing the backbone with a larger ResNet and then conduct experiments on TD500 dataset. Trainable parameters and FLOPs are calculated with an input size 1280×800. Results are shown in Table 10. The results show that TCM-DBNet has better performance than DBNet with less model size and computation overhead, demonstrating its effectiveness for scene text detection.

Ablation Study for the Auxiliary Loss. We further compare the results of with and without auxiliary loss on TD500 dataset, as shown in Table 11. We see that using auxiliary loss achieves higher performance. The results indicate auxiliary loss is beneficial to train the model via imposing constraints on instance-language matching score map. In addition, the improvement of the performance suggests that it might help the image encoder of pretrained CLIP to perceive locality text region effectively.

Method	IC13 → IC15	IC13 → TD	IC15 → MLT17(en)	TT → ArT(-)	ST → IC13	ST → IC15
TCM	71.9	65.1	85.1	68.9	79.5	76.7
w/o VG	68.4 (-3.5)	59.4 (-5.7)	81.8 (-3.3)	59.1 (-9.8)	76.3 (-3.2)	72.6 (-4.1)
w/o LG	66.1 (-5.8)	56.8 (-8.3)	79.7 (-5.4)	57.8 (-11.1)	74.5 (-5.0)	68.2 (-8.5)
w/o VG & LG	64.8 (-7.1)	55.7 (-9.4)	78.4 (-6.7)	54.2 (-14.7)	71.7 (-7.8)	63.9 (-12.8)

Table 7. Ablation study of the effect of LG and VG on generalization performance. F-measure (%) is reported.

	Image encoder	Text encoder	F (%)
LR Factor	0.1	0.0	**88.7**
	0.1	0.1	87.8
	0.1	1.0	87.1
	1.0	1.0	86.3

Table 8. Ablation study of exploration on image encoder and text encoder. "LR" represents the learning rate.

Method	Training Data	Testing Data	F (%)
FCENet	Joint data	NightTime-ArT	55.2
DBNet	Joint data	NightTime-ArT	52.8
TCM-DBNet	Joint data	NightTime-ArT	**70.2**

Table 9. Ablation study of exploration on large amounts of training data.

Method	Backbone	Params	FLOPs	F (%)
DBNet	R50	26 (M)	98 (G)	84.9
DBNet	R101	46 (M)	139 (G)	85.9
DBNet	R152	62 (M)	180 (G)	87.3
TCM-DBNet	R50	50 (M)	156 (G)	**88.7**

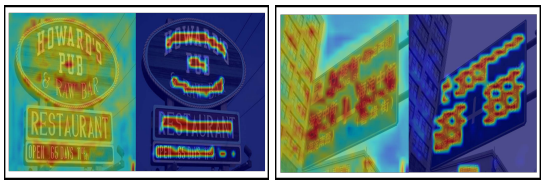
Table 10. Ablation study of the parameters comparison with DBNet.

Model	F (%)
TCM-DBNet with auxiliary loss	**88.7**
TCM-DBNet w/o auxiliary loss	85.1

Table 11. Ablation study of the auxiliary Loss.

5. Discussion of Failure Cases

There are some insightful failed cases as shown in Figure 8. The instance-language matching score map generates false positive region that is very similar to the characteristics of text, as shown in the region of the red circle in Fig. 8, which will be considered as noise. Therefore, it is necessary that the downstream text detection head can further refine this initial score map instead of directly using the score map of instance-language matching as the final results. We leave this problem as future work to alleviate the false positive score map of instance-language matching.



(a) CTW1500 (b) MSRA-TD500

Figure 7. Visualization results of our method. For each pair, the left is the image embedding $\boldsymbol{I}$, and the right is the generated visual prompt $\tilde{\boldsymbol{I}}$. Best view in screen. More results can be found in appendix.

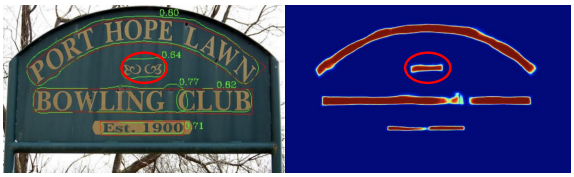


Figure 8. Failure cases. Red circle means false positive region.

6. Conclusion

This paper proposes the TCM, which can directly excavate the prior knowledge from the CLIP model into a scene text detector without pretraining process. Such a new text detection paradigm reveals the importance of using visual-language prior for seeking information from the zero-shot off-the-rack model, and thus guiding the text detector adapting to small-scale data, divergent data distribution, and complicated scenes, without relying on carefully-designed pretraining tasks. Experiments comprehensively demonstrate the effectiveness of our method. It is worth mentioning that we also construct a NightTime-ArT dataset to further demonstrate that the TCM can steer useful prior knowledge from the CLIP model. As the CLIP model is an inborn-friendly framework for text, extension of TCM to scene text spotting is also a promising direction for future work.

Acknowledgements This work was supported by the National Natural Science Foundation of China (No.62225603, No.6220073278, No.62206103), and the National Key Research and Development Program (No.2022YFC3301703, No.2022YFC2305102).

References

[1] Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. Character region awareness for text detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9357–9366, 2019. 2

[2] Chee-Kheng Chng, Yuliang Liu, Yipeng Sun, Chun Chet Ng, Canjie Luo, Zihan Ni, ChuanMing Fang, Shuaitao Zhang, Junyu Han, Errui Ding, et al. ICDAR2019 Robust Reading Challenge on Arbitrary-Shaped Text (RRC-ArT). In *ICDAR*, pages 1571–1576, 2019. 5

[3] Chee-Kheng Ch'ng, Chee Seng Chan, and Cheng-Lin Liu. Total-text: toward orientation robustness in scene text detection. *IJDAR*, pages 1–22, 2019. 5

[4] Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. Multimodal neurons in artificial neural networks. *Distill*, 6(3):e30, 2021. 2, 4

[5] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. In *ICLR*, pages 1–20, 2022. 1

[6] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 3, 5

[7] Minghang He, Minghui Liao, Zhibo Yang, Humen Zhong, Jun Tang, Wenqing Cheng, Cong Yao, Yongpan Wang, and Xiang Bai. Most: A multi-oriented scene text detector with localization refinement. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8809–8818, 2021. 2, 6

[8] Pan He, Weilin Huang, Tong He, Qile Zhu, Yu Qiao, and Xiaolin Li. Single shot text detector with regional attention. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3066–3074, 2017. 2

[9] Wenhao He, Xu-Yao Zhang, Fei Yin, and Cheng-Lin Liu. Deep direct regression for multi-oriented scene text detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 745–753, 2017. 2

[10] D. Karatzas, L. Gomez-Bigorda, et al. ICDAR 2015 competition on robust reading. In *ICDAR*, pages 1156–1160, 2015. 4

[11] Dimosthenis Karatzas, Faisal Shafait, Seiichi Uchida, M. Iwamura, Lluís Gómez i Bigorda, Sergi Robles Mestre, Joan Mas Romeu, David Fernández Mota, Jon Almazán, and Lluís-Pere de las Heras. Icdar 2013 robust reading competition. In *2013 12th International Conference on Document Analysis and Recognition*, pages 1484–1493, 2013. 4, 5

[12] Boyi Li, Kilian Q. Weinberger, Serge J. Belongie, Vladlen Koltun, and René Ranftl. Language-driven semantic segmentation. In *ICLR*, 2022. 1

[13] Xiang Li, Wenhai Wang, Wenbo Hou, Ruo-Ze Liu, Tong Lu, and Jian Yang. Shape robust text detection with progressive scale expansion network. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9328–9337, 2019. 2, 6

[14] Minghui Liao, Baoguang Shi, Xiang Bai, Xinggang Wang, and Wenyu Liu. Textboxes: A fast text detector with a single deep neural network. In *AAAI*, pages 4161–4167, 2017. 2

[15] Minghui Liao, Boyu Song, Minghang He, Shangbang Long, Cong Yao, and Xiang Bai. Synthtext3d: synthesizing scene text images from 3d virtual worlds. *Science China Information Sciences*, 63:1–14, 2020. 6

[16] Minghui Liao, Zhaoyi Wan, Cong Yao, Kai Chen, and Xiang Bai. Real-time scene text detection with differentiable binarization and adaptive scale fusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45:919–931, 2019. 2

[17] Minghui Liao, Zhaoyi Wan, Cong Yao, Kai Chen, and Xiang Bai. Real-time scene text detection with differentiable binarization. In *AAAI*, pages 11474–11481, 2020. 2, 5, 6

[18] Chen-Hsuan Lin, Ersin Yumer, Oliver Wang, Eli Shechtman, and Simon Lucey. St-gan: Spatial transformer generative adversarial networks for image compositing. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9455–9464, 2018. 6

[19] Yuliang Liu, Lianwen Jin, Shuaitao Zhang, Canjie Luo, and Sheng Zhang. Curved scene text detection via transverse and longitudinal sequence connection. *Pattern Recognition*, 90:337–345, 2019. 5

[20] Shangbang Long, Xin He, and Cong Yao. Scene text detection and recognition: The deep learning era. *International Journal of Computer Vision*, 129:161–184, 2020. 2

[21] Shangbang Long, Siyang Qin, Dmitry Panteleev, A. Bissacco, Yasuhisa Fujii, and Michalis Raptis. Towards end-to-end unified scene text detection and layout analysis. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1039–1049, 2022. 2

[22] Shangbang Long, Jiaqiang Ruan, W. Zhang, Xin He, Wenhao Wu, and Cong Yao. Textsnake: A flexible representation for detecting text of arbitrary shapes. In *ECCV*, pages 1–17, 2018. 2

[23] Nibal Nayef, Yash Patel, Michal Busta, Pinaki Nath Chowdhury, Dimosthenis Karatzas, Wafa Khlif, Jiri Matas, Umapada Pal, Jean-Christophe Burie, Cheng-lin Liu, et al. ICDAR2019 Robust Reading Challenge on Multi-lingual Scene Text Detection and Recognition–RRC-MLT-2019. In *ICDAR*, pages 1454–1459, 2019. 5

[24] Nibal Nayef, Fei Yin, Imen Bizid, Hyunsoo Choi, Yuan Feng, Dimosthenis Karatzas, Zhenbo Luo, Umapada Pal, Christophe Rigaud, Joseph Chazalon, Wafa Khlif, Muhammad Muzzamil Luqman, Jean-Christophe Burie, Cheng-Lin Liu, and Jean-Marc Ogier. Icdar2017 robust reading challenge on multi-lingual scene text detection and script identification - rrc-mlt. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 01, pages 1454–1459, 2017. 5

[25] Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H. Miller, and Sebastian Riedel. Language models as knowledge bases? In *EMNLP*, page 1772–1791, 2019. 3

[26] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry,

Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, pages 1–16, 2021. 1, 2

[27] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18061–18070, 2022. 1

[28] Baoguang Shi, Xiang Bai, and Serge J. Belongie. Detecting oriented text in natural images by linking segments. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3482–3490, 2017. 2, 6

[29] Sibo Song, Jianqiang Wan, Zhibo Yang, Jun Tang, Wenqing Cheng, Xiang Bai, and Cong Yao. Vision-language pre-training for boosting scene text detectors. In *CVPR*, pages 15681–15691, 2022. 1, 2, 6

[30] Jun Tang, Zhibo Yang, Yongpan Wang, Qi Zheng, Yongchao Xu, and Xiang Bai. Seglink++: Detecting dense and arbitrary-shaped scene text by instance-aware component grouping. *Pattern Recognit.*, 96:106954, 2019. 2

[31] Jingqun Tang, Wenqing Zhang, Hong yi Liu, Mingkun Yang, Bo Jiang, Guan-Nan Hu, and Xiang Bai. Few could be better than all: Feature sampling and grouping for scene text detection. In *CVPR*, pages 4563–4572, 2022. 6

[32] Zhi Tian, Weilin Huang, Tong He, Pan He, and Yu Qiao. Detecting text in natural image with connectionist text proposal network. In *ECCV*, pages 1–16, 2016. 2

[33] Zhuotao Tian, Michelle Shu, Pengyuan Lyu, Ruiyu Li, Chao Zhou, Xiaoyong Shen, and Jiaya Jia. Learning shape-aware embedding for scene text detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4229–4238, 2019. 2

[34] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 1–11, 2017. 4

[35] Qi Wan, Haoqin Ji, and Linlin Shen. Self-attention based text knowledge mining for text detection. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5979–5988, 2021. 2, 6

[36] Fangfang Wang, Yifeng Chen, Fei Wu, and Xi Li. Textray: Contour-based geometric modeling for arbitrary-shaped scene text detection. In *Proceedings of the 28th ACM International Conference on Multimedia*, page 111–119, 2020. 2

[37] Wenhai Wang, Enze Xie, Xiaoge Song, Yuhang Zang, Wenjia Wang, Tong Lu, Gang Yu, and Chunhua Shen. Efficient and accurate arbitrary-shaped text detection with pixel aggregation network. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8439–8448, 2019. 2, 5, 6

[38] Xiaobing Wang, Yingying Jiang, Zhenbo Luo, Cheng-Lin Liu, Hyunsoo Choi, and Sungjin Kim. Arbitrary shape scene text detection with adaptive text region representation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6442–6451, 2019. 2

[39] Yuxin Wang, Hongtao Xie, Zhengjun Zha, Mengting Xing, Zilong Fu, and Yongdong Zhang. Contournet: Taking a further step toward accurate arbitrary-shaped scene text detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11750–11759, 2020. 2

[40] Weijia Wu, Ning Lu, Enze Xie, Yuxiang Wang, Wenwen Yu, Cheng Yang, and Hong Zhou. Synthetic-to-real unsupervised domain adaptation for scene text detection in the wild. In *ACCV*, pages 1–14, 2020. 6

[41] Enze Xie, Yuhang Zang, Shuai Shao, Gang Yu, Cong Yao, and Guangyao Li. Scene text detection with supervised pyramid context network. In *AAAI*, pages 9038–9045, 2019. 2

[42] Linjie Xing, Zhi Tian, Weilin Huang, and Matthew R. Scott. Convolutional character networks. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9125–9135, 2019. 6

[43] Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A simple baseline for zero-shot semantic segmentation with pre-trained vision-language model. In *ECCV*, 2021. 1

[44] Yongchao Xu, Yukang Wang, Wei Zhou, Yongpan Wang, Zhibo Yang, and Xiang Bai. Textfield: Learning a deep direction field for irregular scene text detection. *IEEE Transactions on Image Processing*, 28:5566–5579, 2019. 2

[45] Chuhui Xue, Shijian Lu, and Wei Zhang. Msr: Multi-scale shape regression for scene text detection. In *IJCAI*, pages 989–995, 2019. 2

[46] Chuhui Xue, Wenqing Zhang, Yu Hao, Shijian Lu, Philip H. S. Torr, and Song Bai. Language matters: A weakly supervised vision-language pre-training approach for scene text detection and spotting. In *ECCV*, pages 1–19, 2022. 1, 2, 6

[47] Cong Yao, Xiang Bai, Wenyu Liu, Yi Ma, and Zhuowen Tu. Detecting texts of arbitrary orientations in natural images. In *2012 IEEE conference on computer vision and pattern recognition*, pages 1083–1090. IEEE, 2012. 5

[48] Jian Ye, Zhe Chen, Juhua Liu, and Bo Du. Textfusenet: Scene text detection with richer fused features. In *International Joint Conference on Artificial Intelligence*, 2020. 2

[49] Fangneng Zhan, Chuhui Xue, and Shijian Lu. Ga-dan: Geometry-aware domain adaptation network for scene text detection and recognition. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9104–9114, 2019. 5, 6

[50] Chengquan Zhang, Borong Liang, Zuming Huang, Mengyi En, Junyu Han, Errui Ding, and Xinghao Ding. Look more than once: An accurate detector for text of arbitrary shapes. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10544–10553, 2019. 2, 6

[51] Shi-Xue Zhang, Xiaobin Zhu, Jie-Bo Hou, Chang Liu, Chun Yang, Hongfa Wang, and Xu-Cheng Yin. Deep relational reasoning graph network for arbitrary shape text detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9696–9705, 2020. 2

[52] Zheng Zhang, Chengquan Zhang, Wei Shen, Cong Yao, Wenyu Liu, and Xiang Bai. Multi-oriented text detection with fully convolutional networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4159–4167, 2016. 2

[53] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *CVPR*, pages 16816–16825, 2022. 1, 3

[54] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, page 2337–2348, 2022. 3

[55] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. East: An efficient and accurate scene text detector. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2642–2651, 2017. 2, 5, 6

[56] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251, 2017. 6

[57] Yiqin Zhu, Jianyong Chen, Lingyu Liang, Zhuanghui Kuang, Lianwen Jin, and Wayne Zhang. Fourier contour embedding for arbitrary-shaped text detection. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3122–3130, 2021. 2, 5