SwinTextSpotter: Scene Text Spotting via Better Synergy between Text Detection and Text Recognition

Mingxin Huang[1†] Yuliang Liu[2†] Zhenghao Peng[2] Chongyu Liu[1] Dahua Lin[2]
Shenggao Zhu[3] Nicholas Yuan[3] Kai Ding[4] Lianwen Jin[1,5*]
[1]South China University of Technology [2]Chinese University of Hong Kong [3]Huawei Cloud AI
[4]IntSig Information Co., Ltd [5]Peng Cheng Laboratory
eelwjin@scut.edu.cn

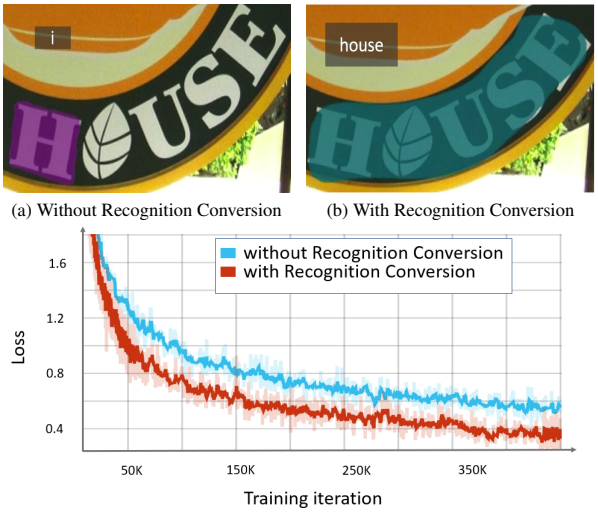
Abstract

End-to-end scene text spotting has attracted great attention in recent years due to the success of excavating the intrinsic synergy of the scene text detection and recognition. However, recent state-of-the-art methods usually incorporate detection and recognition simply by sharing the backbone, which does not directly take advantage of the feature interaction between the two tasks. In this paper, we propose a new end-to-end scene text spotting framework termed SwinTextSpotter. Using a transformer encoder with dynamic head as the detector, we unify the two tasks with a novel Recognition Conversion mechanism to explicitly guide text localization through recognition loss. The straightforward design results in a concise framework that requires neither additional rectification module nor character-level annotation for the arbitrarily-shaped text. Qualitative and quantitative experiments on multi-oriented datasets RoIC13 and ICDAR 2015, arbitrarily-shaped datasets Total-Text and CTW1500, and multi-lingual datasets ReCTS (Chinese) and VinText (Vietnamese) demonstrate SwinTextSpotter significantly outperforms existing methods. Code is available at https://github.com/mxin262/SwinTextSpotter.

1. Introduction

Scene text spotting, which aims to detect and recognize the entire word or sentence in natural images, has raised a lot of attention due to its wide range of applications in autonomous driving [64], intelligent navigation [42, 50], and key entities recognition [51, 65], *etc*. Traditional scene text spotting methods treat detection and recognition as two separate tasks and adopt a pipeline that first localizes and crops the text regions on the input images and then predicts the text sequence by feeding the cropped regions into text recognizer [9, 14, 16, 23, 35]. Such a pipeline may have some limitations, such as (1) error accumulation between these two tasks, e.g., imprecise detection result may heavily hinder the performance of text recognition; (2) separate optimization of the two tasks might not maximize the final performance of text spotting; (3) intensive memory consumption and low inference efficiency.

Therefore, many methods [12,20,27,32] attempt to solve text spotting in end-to-end systems, *i.e.*, optimizing detection and recognition jointly in unified architectures. The recognizer can improve the performance of the detector by eliminating the false positive detection results [20, 21]. In turn, even if the detection is not precise, the recognizer can still correctly predict the text sequence by the large recep-



(a) Without Recognition Conversion (b) With Recognition Conversion

(c) The recognition loss with and without Recognition Conversion.

Figure 1. Effectiveness of Recognition Conversion. The proposed Recognition Conversion explicitly guides the detection, leading to better text spotting performance.

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

tive field of the feature map [21, 27]. Another advantage is that an end-to-end system is easier to maintain and transfer to new domains compared to a cascaded pipeline where the model is coupled with the data and thus requires substantial engineering efforts [30, 55].

However, there are two limitations in most of the existing end-to-end scene text spotting systems [8, 22, 28, 32, 38, 39, 49]. First, if the detection is simply based on the visual information in the input features, the detector is prone to be distracted by background noise and proposes inconsistent detection, as depicted in Figure 1(a). The interaction between texts in the same image is the crucial factor to eliminate the impact of background noise, since different characters of the same word may contain strong similarities, such as the backgrounds and text styles. Using Transformer [48] can learn rich interactions between text instances. For example, Yu et al. [62] use transformer to make texts interact with each other at semantic level. Fang et al. [7] and Wang et al. [57] further adopt transformer to model the visual relationship between texts. Second, the interactions between detection and recognition is not enough by sharing backbone because neither the recognition loss optimizes the detector nor the recognizer utilizes the detection features. To jointly improve detection and recognition, a character segmentation map is designed by Mask TextSpotter [21], which simultaneously optimizes the detection and recognition results in the same branch; ABCNet v2 [30] proposes Adaptive End-to-End Training (AET) strategy using the detection results to extract recognition features instead of only using the ground truths; ARTS [67] improves the performance of the end-to-end text spotting by back-propagating the loss from the recognition branch to the detection branch using a differentiable Spatial Transform Network (STN) [15]. However, these three methods assume the detector proposes text features structurally, *e.g.* in the reading order. The overall performance of the text spotting is thereafter bounded by the detector.

We propose *SwinTextSpotter*, an end-to-end trainable Transformer-based framework, stepping toward better synergy between the text detection and recognition. To better distinguish the densely scattered text instances in crowded scenes, we use Transformer and a two-level self-attention mechanism in SwinTextSpotter, stimulating the interactions between the text instances. Addressing the challenge in arbitrarily-shaped scene text spotting, inspired by [13, 45], we regard text detection task as a set-prediction problem and thus adopt a query-based text detector. We further propose *Recognition Conversion (RC)*, which implicitly guides the recognition head through incorporating the detection features. RC can back-propagate recognition information to the detector and suppress the background noise in the features for recognition, leading to the joint optimization of the detector and recognizer. Empowered by the proposed RC, SwinTextSpotter has a concise framework without the character-level annotation and rectification module used in previous works to improve the recognizer. SwinTextSpotter has superior performance in both the detection and the recognition. As illustrated in Figure 1(b), the detector of SwinTextSpotter can accurately localize difficult samples. On the other hand, more accurate detection features can improve the recognizer and result in faster convergence and better performance, as shown in Figure 1(c).

We conduct extensive experiments on six benchmarks, including multi-oriented dataset RoIC13 [22] and ICDAR 2015 [18], arbitrarily-shaped dataset Total-Text [6] and SCUT-CTW1500 [29], and multilingual dataset ReCTS (Chinese) [66] and VinText (Vietnamese) [36]. The results demonstrate the superior performance of the SwinTextSpotter: (1) SwinTextSpotter achieves 88.0% F-measure for the detection task on SCUT-CTW1500 and Total-Text, exceeding previous methods by a large margin; (2) SwinTextSpotter significantly outperforms ABCNet v2 [30] by 9.8% in terms of 1-NED for the text spotting task in ReCTs dataset. Additionally, without using character-level annotation on ReCTs, SwinTextSpotter outperforms previous state-of-the-art methods MaskTextSpotter [21] and AE TextSpotter [54] that use such annotation; (3) SwinTextSpotter shows better robustness for the extremely rotated instances on RoIC13 dataset compared to MaskTextSpotter v3 [21]. The main contributions of this work are summarized as follows.

- SwinTextSpotter groundbreakingly shows that Transformer and the set-prediction scheme are effective in end-to-end scene text spotting.
- SwinTextSpotter adopts the *Recognition Conversion* to exploit the synergy of text detection and recognition.
- SwinTextSpotter is a concise framework that does not require character-level annotation as well as specifically designed rectification module for recognizing arbitrarily-shaped text.
- SwinTextSpotter achieves state-of-the-art performance on multiple public scene text benchmarks.

2. Related Work

Separate Scene Text Spotting. In past decades, the emergence of deep learning approaches greatly promote the development of scene text spotting. Wang et al. [52] use a sliding-window-based detector to detect characters and then classify each character. Bissacco et al. [2] combine DNN and HOG features and build a text extraction system by using characters classification. Liao et al. [24] propose the TextBoxes that incorporates the single-shot detector and a text recognizer [43] in two-stage manner. However, the aforementioned methods treat detection and recognition as

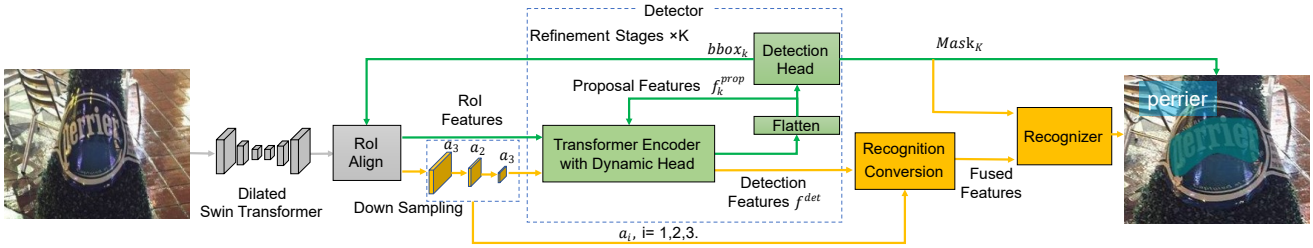


Figure 2. The framework of the proposed SwinTextSpotter. The gray arrows denote the feature extraction from images. The green arrows and orange arrows represent the detection stage and the recognition stage, respectively. The outputs of detection head are refined in K stages. The output detection in the K^{th} stage serves as the input to the recognition stage.

separate tasks without exchange of information between the two tasks.

End-to-End Text Spotting. Recently, researchers try to combine detection and recognition into one system. Li et al. [20] unify the detection and recognition into an end-to-end trainable scene text spotting framework. FOTS [27] uses a one stage detector to generate rotated boxes and adopts RoIRotate to sample the oriented text feature into horizontal grid for connecting the detection and recognition. He et al. [12] propose a similar framework using an attention-based recognizer.

For the task of arbitrarily-shaped scene text spotting, Mask TextSpotter series [21, 22, 32] solve the problem without explicit rectification by using character segmentation branch to improve the performance of the recognizer. TextDragon [8] combines the two tasks by RoISlide, a technique that transforms the predicting segments of the text instances into horizontal features. Wang et al. [49] adopt Thin-Plate-Spline [3] transformation to rectify the features. ABCNet [28] and its improved version ABCNet v2 [30] use the BezierAlign to transform the arbitrary-shape texts into regular ones. These methods achieve great progress by using rectification module to unify detection and recognition into end-to-end trainable systems. Qin et al. [39] propose RoI Masking to extract the feature for arbitrarily-shaped text recognition. Similar to [39], PAN++ [55] is based on a faster detector [56]. AE TextSpotter [54] uses the results of recognition to guide detection through language model. Though achieve significantly improvement on the performance of text spotting by sharing backbone, the aforementioned methods neither back-propagate recognition loss to the detector nor use detection features in the recognizer. The detector and the recognizer thus are still relatively independent to each other without joint optimization. Recently, Zhong et al. [67] propose ARTS which passes the gradient of recognition loss to the detector using Spatial Transform Network (STN) [15], demonstrating the power of synergy between the detection and recognition in text spotting.

3. Methodology

The overall architecture of SwinTextSpotter is presented in Figure 2, which consists of four components: (1) a backbone based on Swin-Transformer [31]; (2) a query-based text detector; (3) a *Recognition Conversion* module to bridge the text detector and recognizer; and (4) an attention-based recognizer.

As illustrated in the green arrows of Figure 2, in the first stage of detection, we first randomly initialize trainable parameters to be the boxes $bbox_0$ and proposal features f_0^{prop}. To make the proposal features contain global information, we use global average pooling to extract the image features and add them into f_0^{prop}. We then extract the RoI features using $bbox_0$. The RoI features and f_0^{prop} are fed into the Transformer encoder with dynamic head. The output of the Transformer encoder is flattened and forms the proposal features f_1^{prop}, which will be fed into the detection head to output the detection result. The box $bbox_{k-1}$ and proposal feature f_{k-1}^{prop} will serve as the input to later k^{th} stage of detection. The proposal feature f_k^{prop} recurrently updates itself by fusing the RoI features with previous f_{k-1}^{prop}, which makes proposal features preserve the information from previous stages. We repeat such refinement for totally K stages, resembling the iterative structure in the query-based detector [4, 13, 45, 68]. Such design allows more robust detection in sizes and aspect ratios [45]. More details of the detector are explained in Section 3.2.

Since the recognition stage (orange arrows) requires higher rate of resolution than detection, we use the final detection stage output box $bbox_K$ to obtain the RoI features whose resolution is four times as much as that in the detection stage. In order to keep the resolution of features consistent with the detector when fused with proposal features, we down-sample the RoI features to get three feature maps of descending sizes, denoting by $\{a_1, a_2, a_3\}$. Then we obtain detection features f^{det} by fusing the smallest a_3 and the proposal features f_K^{prop}. The detection features f^{det} in recognition stage contain all previous detection information. Finally the $\{a_1, a_2, a_3\}$ and the detection features f^{det}

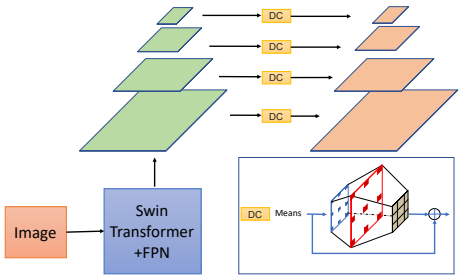


Figure 3. Illustration of the designed Dilated Swin-Transformer. The DC refers to two dilated convolution layers, one vanilla convolution layer and one residual structure.

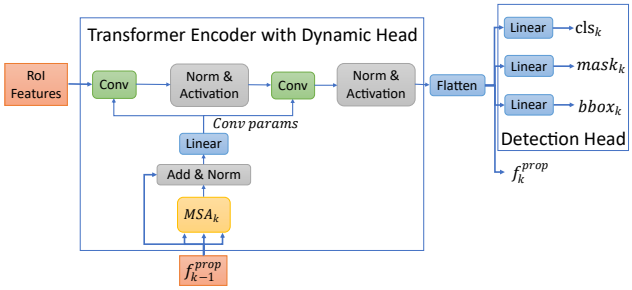


Figure 4. Illustration of k^{th} stage in detection. f_{k-1}^{prop} is the proposal features output by previous stage. MSA_k refers to the multi-head-attention in k^{th} stage. f_k^{prop} will serve as the input to next stage.

are sent into *Recognition Conversion* and recognizer for generating the recognition result. More details of *Recognition Conversion* and recognizer are explained in Section 3.3 and Section 3.4, respectively.

3.1. Dilated Swin-Transformer

Vanilla convolutions operate locally at fixed size (e.g. 3×3), which causes low efficacy in connecting remote features. For text spotting, however, modeling the relationships between different texts is critical since scene texts from the same image share strong similarities, such as their backgrounds and text styles. Considering the global modeling capability and computational efficiency, we choose Swin-Transformer [31] with a Feature Pyramid Network (FPN) [25] to build our backbone. Given the blanks existing between words in a line of text, the receptive field should be large enough to help distinguish whether adjacent texts belong to the same text line. To achieve such receptive field, as illustrated in Figure 3, we incorporate two dilated convolution layers [63], one vanilla convolution layer and one residual structure into the original Swin-Transformer, which also introduce the properties of CNN to Transformer [59].

3.2. A Query Based Detector

We use a query based detector to detect the text. Based on Sparse R-CNN [45], the query based detector is built on ISTR [13] which treats detection as a set-prediction problem. Our detector uses a set of learnable proposal boxes, alternative to replace massive candidates from the RPN [40], and a set of learnable proposal features, representing high-level semantic vectors of objects. The detector is empirically designed to have six query stages. With the Transformer encoder with dynamic head, latter stages can access the information in former stages stored in the proposal features [17, 45, 47]. Through multiple stages of refinement, the detector can be applied to text at any scale.

The architecture of the detection head in k^{th} stage is illustrated in Figure 4. The proposal features in $k-1$ stage is represented by $f_{k-1}^{prop} \in \mathbb{R}^{N,d}$. At the stage k, the proposal features f_{k-1}^{prop} produced in previous stage is fed into a self-attention module [48] MSA_k to model the relationships and generate two sets of convolutional parameters. The detection information in previous stages is therefore embedded into the convolutions. The convolutions conditioned on the previous proposal features is used to encode the RoI features. The RoI features are extracted by $bbox_{k-1}$, the detection result in previous stage, using RoIAlign [11]. The output features of the convolutions is fed into a linear projection layer to produce the f_k^{prop} for next stage. The f_k^{prop} is subsequently fed the into prediction head to generate $bbox_k$ and $mask_k$. To reduce computation, the 2D mask is transformed into 1D mask vector by the Principal Component Analysis [58] so the $mask_k$ is a one-dimensional vector.

When $k = 1$, the $bbox_0$ and f_0^{prop} are randomly initialized parameters, which is the input of the first stage. During training, these parameters are updated via back propagation and learn the inductive bias of the high-level semantic features of text.

We view the text detection task as a set-prediction problem. Formally, we use the bipartite match to match the predictions and ground truths [4, 13, 44, 45]. The matching cost becomes:

$$L_{match} = \lambda_{cls} \cdot L_{cls} + \lambda_{L1} \cdot L_{L1} + \lambda_{giou} \cdot L_{giou} + \lambda_{mask} \cdot L_{mask}, \quad (1)$$

where λ is the hyper-parameter used to balance the loss. L_{cls} is the focal loss [26]. The losses for regressing the bounding boxes are L1 loss L_{L1} and generalized IoU loss L_{giou} [41]. We compute the mask loss L_{mask} following [13], which calculates the cosine similarity between the prediction mask and ground truth. The detection loss is similar to the matching cost but we use the L2 loss and dice loss [33] to replace the cosine similarity as in [13].

3.3. Recognition Conversion

To better coordinate the detection and recognition, we propose *Recognition Conversion (RC)* to spatially inject the

features from detection head into the recognition stage, as illustrated in Figure 5. The RC consists of the Transformer encoder [48] and four up-sampling structures. The input of RC are the detection features f^{det} and three down-sampling features $\{a_1, a_2, a_3\}$.

The detection features are sent to the Transformer encoder $TrE()$, making the information of previous detection stages further fused with a_3. Then through a stack of up-sampling operation $E_u()$ and Sigmoid function $\phi()$, three masks $\{M_1, M_2, M_3\}$ for text regions are generated:

$$d_1 = TrE(f^{det}), \tag{2}$$
$$d_2 = (E_u(d_1) + a_2), \tag{3}$$
$$d_3 = (E_u(d_2) + a_1), \tag{4}$$
$$M_i = \phi(d_i), i = 1, 2, 3. \tag{5}$$

With the masks $\{M_1, M_2, M_3\}$ and the input features $\{a_1, a_2, a_3\}$, we further integrate these features effectively under the following pipeline:

$$r_1 = M_1 \cdot a_3, \tag{6}$$
$$r_2 = M_2 \cdot (E_u(r_1) + a_2), \tag{7}$$
$$r_3 = M_3 \cdot (E_u(r_2) + a_1), \tag{8}$$

where $\{r_1, r_2, r_3\}$ denote the recognition features. The r_3 is the fused features in Figure 5, which is finally sent to the recognizer at the highest resolution. As shown in the blue dashed lines in Figure 5, the gradient of the recognition loss L_{reg} can be back-propagated to the detection features, enabling RC to implicitly improve the detection head through the recognition supervision.

Generally, to suppress the background, the fused features will be multiplied by a $mask_K$ predicted by detection head (with the supervision of L_{mask}). However, the background noise still remains in the feature maps as the detection box is not tight enough. Such issue can be alleviated by the proposed RC since RC uses the detection features to generate tight masks to suppress the background noise, which is supervised by the recognition loss apart from the detection loss. As shown in the upper right corner of Figure 5, M_3 suppresses more background noise than $mask_K$, where M_3 has higher activation in texts region and lower in the background. Therefore the masks $\{M_1, M_2, M_3\}$ produced by RC, which will be applied to the recognition features $\{r_1, r_2, r_3\}$, makes recognizer easier to concentrate on the text regions.

With RC, the gradient of recognition loss not only flows back to the backbone network, but also to the proposal features. Optimized by both detection supervision and recognition supervision, the proposal features can better encode the high-level semantic information of the texts. Therefore, the proposed RC can incentivize the coordination between detection and recognition.

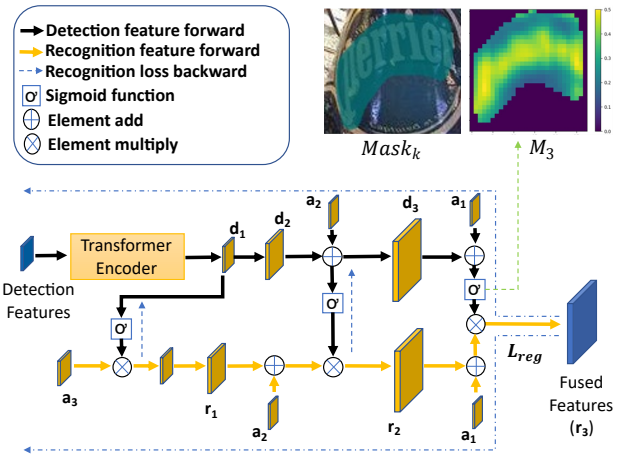


Figure 5. Detailed structure of *Recognition Conversion*.

3.4. Recognizer

After applying RC on the feature map, background noise is effectively suppressed and thus the text regions can be bounded more precisely. This enables us to merely use a sequential recognition network to obtain promising recognition results without rectification modules such as TPS [3], RoISlide [8], Bezier-Align [28] or character-level segmentation branch used in MaskTextSpotter [21]. To enhance the fine-grained feature extraction and sequence modeling, we adopt a bi-level self-attention mechanism, inspired by [61], as the recognition encoder. The two-level self-attention mechanism (TLSAM) contains both fine-grained and coarse-grained self-attention mechanisms for local neighborhood regions and global regions, respectively. Therefore, it can effectively extract fine-grained features while maintaining global modeling capability. As for the decoder, we simply follow MaskTextSpotter by using the Spatial Attention Module (SAM) [22]. The recognition loss is as follow:

$$L_{reg} = -\frac{1}{T}\sum_{k=1}^{T} log\ p(y_i), \tag{9}$$

wherein T is the max length of the sequence and $p(y_i)$ is the probability of sequence.

4. Experiments

We conduct experiments on various scene text benchmarks, including multi-oriented scene text benchmarks RoIC13 [22] and ICDAR 2015 [18], multilingual datasets ReCTS [66] and Vintext [36], and two arbitrarily-shaped scene text benchmarks Total-Text [6] and SCUT-CTW 1500 [29]. The ablation studies are conducted on Total-Text to verify each component of our proposed method.

Method	ICDAR 2015 End-to-End		
	S	W	G
FOTS [27]	81.1	75.9	60.8
Mask TextSpotter [21]	83.0	77.7	73.5
CharNet [60]	83.1	**79.2**	69.1
TextDragon [8]	82.5	78.3	65.2
Mask TextSpotter v3 [22]	83.3	78.1	**74.2**
MANGO [37]	81.8	78.9	67.3
PAN++ [55]	82.7	78.2	69.2
ABCNet v2 [30]	82.7	78.5	73.0
SwinTextSpotter	**83.9**	77.3	70.5

Table 1. End-to-end recognition result on ICDAR 2015. "S", "W", and "G" represent recognition with "Strong", "Weak", and "Generic" lexicon, respectively.

Method	Detection			1-NED
	R	P	H	
FOTS [27]	82.5	78.3	80.31	50.8
MaskTextSpotter [21]	88.8	89.3	89.0	67.8
AE TextSpotter [54]	**91.0**	92.6	**91.8**	71.8
ABCNet v2 [30]	87.5	93.6	90.4	62.7
SwinTextSpotter	87.1	**94.1**	90.4	**72.5**

Table 2. End-to-end text spotting result and detection result on ReCTS.

4.1. Implementation Details

We follow the training strategy in [37]. First, the model is pretrained on the Curved SynthText [28], ICDAR-MLT [34], and the corresponding dataset for 450K iterations. The initialized learning rate is 2.5×10^{-5}, which reduces to 2.5×10^{-6} at $380K^{th}$ iteration and 2.5×10^{-7} at $420K^{th}$ iteration. Then we jointly train the pretrained model for 80K iterations on the Total-Text, ICDAR 2013, and ICDAR-MLT, which decays to a tenth at 60K. Finally, we fine-tune the jointly trained model on the corresponding datasets. We also follow the training strategies in [30] and [36] to train the model on Chinese and Vietnamese.

We extract 4 feature maps with 1/4, 1/8, 1/16, 1/32 resolution of the input image for text detection and recognition. We train our model with image batch size of 8. The following data augmentation strategies are used: (1) random scaling; (2) random rotation; and (3) random crop. Other strategies such as random brightness, contrast, and saturation are also applied during training.

4.2. Datasets

We use the following datasets: **Curved SynthText** [28] is a synthesized dataset for arbitrarily-shaped scene text. It contains 94,723 images with multi-oriented text and 54,327 images with curved text. **ICDAR 2013** [19] is a scene text dataset proposed in 2013. It contains 229 training images and 233 test images. **ICDAR 2015 [18]** is built in 2015. It contains 1,000 training images and 500 test images. **ICDAR 2017** [34] is a multi-lingual text dataset. It contains 7,200 training images, 1,800 validation images. We only select the images containing English texts for training. **ICDAR19 ArT** [5] is a dataset for arbitrarily shaped text. It contains 5,603 training images. **ICDAR19 LSVT** [46] is a large number of Chinese datasets which contains 30,000 for training images. **Total-Text** [6] is the benchmark for arbitrarily-shaped scene text. It consists of 1,255 training images and 300 testing images. The word-level polygon boxes are provided as the annotations. **SCUT-CTW1500** [29] is a text-line level arbitrarily-shaped scene text dataset. It consists of 1,000 training images and 500 testing images. Compared to Total-Text, this dataset contains denser and longer text. **ReCTS** [66] consists of 20,000 training images and 5,000 testing images. It also provides character-level bounding boxes, which are not used in our method. **VinText** [36] is a recently proposed Vietnamese text dataset. It consists of 1,200 training images and 500 testing images.

4.3. Comparisons with State-of-the-Art methods

Except in special cases, all values in the table are in percentage.

Multi-oriented and Multilingual datasets. We first conduct experiments on ICDAR 2015, showing the superiority of SwinTextSpotter on oriented scene text. Table 1 shows that SwinTextSpotter achieves the best strong lexicon results on ICDAR 2015, without using the character-level annotations which were used by ABCNet v2 and Mask-TextSpotter v3. We also conduct experiments on RoIC13 dataset proposed in [22] to verify the rotation robustness of SwinTextSpotter. The end-to-end recognition results are shown in Table 3. Both in Rotation Angle $45°$ and in Rotation Angle $60°$ datasets, SwinTextSpotter can achieve the state-of-the-art in terms of the H-mean metric. Our method significantly outperforms the Mask TextSpotter v3 by 1.5% in terms of H-mean on Rotation Angle $45°$ and 1.3% on Rotation Angle $60°$. In addition to English, we also conduct experiment on Chinese dataset ReCTS and Vietnamese dataset VinText to verify the generality of SwinTextSpotter. As shown in Table 2, for ReCTS, our method surpasses ABCNet v2, which only works on word-level annotation, by 9.8% in 1-NED. SwinTextSpotter has 0.7% higher 1-NED than AE-TextSpotter, the SOTA method requiring additional character-level annotation. For VinText, the end-to-end results are presented in Table 4, "D" means using dictionary for the training of recognizer. SwinTextSpotter can also outperform previous methods on VinText, showing the generalization of our method. It is worth noting that for the above tasks, we do not use dictionary for the training of recognizer as ABCNet+D and Mask TextSpotter v3+D.

Method	Rotation Angle 45°			Rotation Angle 60°		
	R	P	H	R	P	H
CharNet [60]	35.5	34.2	33.9	8.4	10.3	9.3
Mask TextSpotter [21]	45.8	66.4	54.2	48.3	68.2	56.6
Mask TextSpotter v3 [22]	66.8	**88.5**	76.1	67.6	**88.5**	76.6
SwinTextSpotter	**72.5**	83.4	**77.6**	**72.1**	84.6	**77.9**

Table 3. End-to-end recognition result on RoIC13. P, R, H represent precision, recall and Hmean, respectively.

Method	H-mean
ABCNet [28]	54.2
ABCNet+D [36]	57.4
Mask Textspotter v3 [36]	53.4
Mask Textspotter v3+D [36]	68.5
SwinTextSpotter	**71.1**

Table 4. End-to-end text spotting result on VinText. ABCNet+D means adding the methods proposed in [36] to ABCNet. The same to Mask Textspotter v3+D.

Method	Detection	End-to-End	
	H-mean	None	Full
CharNet [60]	85.6	66.6	-
ABCNet [28]	-	64.2	75.7
PGNet [53]	86.1	63.1	-
Mask TextSpotter [21]	85.2	65.3	77.4
Qin et al. [39]	-	67.8	-
Mask TextSpotter v3 [22]	-	71.2	78.4
MANGO [37]	-	72.9	83.6
ABCNet v2 [30]	87.0	70.4	78.1
PAN++ [55]	-	68.6	78.6
SwinTextSpotter-Res	87.2	72.4	83.0
SwinTextSpotter	**88.0**	**74.3**	**84.1**

Table 5. End-to-end text spotting result and detection result on Total-Text. SwinTextSpotter-Res means using the ResNet50 with FPN as backbone. "None" represents lexicon-free. "Full" represents that we use all the words appeared in the test set.

Method	Detection	End-to-End		1-NED
	H-mean	None	Full	
TextDragon [8]	83.6	39.7	72.4	-
ABCNet [28]	81.4	45.2	74.1	-
MANGO [37]	-	**58.9**	**78.7**	-
ABCNet v2 [30]	84.7	57.5	77.2	46.9
SwinTextSpotter	**88.0**	51.8	77.0	45.7

Table 6. End-to-end text spotting result and detection result on SCUT-CTW1500. "None" represents lexicon-free. "Full" represents that we use all the words appeared in the test set.

The qualitative results of these three benchmarks are shown in Figure 6 (c)(d)(e)(f).

Irregular text. We conduct experiments on two arbitrarily-shaped scene text datasets (Total-Text and SCUT-CTW1500) for both detection and end-to-end scene text spotting tasks. In text detection task, the results in Table 5 and 6 demonstrate that SwinTextSpotter can achieve 88% H-mean on both datasets, which outperforms previous state-of-the-art methods 1.0% and 3.3% for Total-Text and SCUT-CTW1500, respectively. As for end-to-end scene text spotting task, according to Table 5, SwinTextSpotter significantly outperforms previous methods on Total-Text with 74.3% F-measure, 3.9% higher than ABCNet v2 and 1.4% higher than MANGO. Besides, for fair comparison to previous methods, we replace our Dilated Swin-Transformer backbone with ResNet-50 and the performance is still comparable to the best result (72.4% in our method and 72.9% in MANGO). On SCUT-CTW1500, however, as presented in Table 6, though our method can achieve the best performance on text detection, the end-to-end text spotting result still contains a gap. We discuss and analyze such phenomenon in Section 4.5. Some qualitative results are shown in Figure 6(a)(b).

4.4. Ablation Studies

To evaluate the effectiveness of the proposed components, we conduct ablation studies on Total-Text. ResNet-50 is used as the baseline backbone. In ablation studies, we only train different variants of SwinTextSpotter on the Curved SynthText, ICDAR-MLT and the corresponding dataset as the first stage described in Section 4.1.

Recognition Conversion. As shown in Table 7, with RC, the results can be improved by 3.0% and 6.9% for detection and end-to-end scene text spotting, respectively. RC greatly improve the performance of the detector and recognizer. This is mainly because the RC can generate more discriminative features for text regions to boost the performance of text recognition so as to benefit the text detector.

Dilated Swin-Transformer. We also compare the performance of different backbones. The model with Swin-Transformer can achieve 2.9% improvement on end-to-end results over the ResNet-50, but there is no improvement for detection. Incorporating dilated convolution into Swin-Transformer can further boost detection by 0.7% and 0.5% in end-to-end results.

Two-level self-attention mechanisms. In Table 7, we

Method	Recognition Conversion	Swin-Transformer	TLSAM	Dilated convolution	Total-Text	
					Det-Hmean	E2E-Hmean
Baseline					78.9	55.7
Baseline+	✓				81.9	62.6
Baseline+	✓	✓			81.7	65.5
Baseline+	✓		✓		81.7	62.5
Baseline+	✓	✓		✓	82.4	66.0
SwinTextSpotter	✓	✓	✓	✓	**83.2**	**66.9**

Table 7. Ablation studies on Total-Text without finetuning. ResNet-50 is used as the baseline backbone. TLSAM stands for the two-level self-attention mechanism.

(a) Total-Text (b) CTW1500 (c) VinText

(d) ReCTS (e) ICDAR2015 (f) RoIC13

Figure 6. Visualization results of our method. White text represents the correct results; Red text represents the wrong results; Blue text represents that the GT of the text instance is marked as "do not care". Best view in screen.

further conduct experiments to explore the influence of fine-grained features. Dilated Swin-Transformer performs poorly in capturing fine-grained features, while two-level self-attention mechanisms can effectively make up for this shortcoming. The enhancement of fine-grained features from two-level self-attention mechanism can result in 0.8% and 0.9% improvement on text detection and end-to-end text spotting results, respectively.

4.5. Limitation and Discussion

Long Arbitrarily-Shaped Text. We know that the long arbitrarily-shaped text requires a high resolution feature map to be recognized. When the feature map becomes larger, attention map in the recognizer will also expand. Large attention map may result in mismatch of the recognizer, which leads to the low end-to-end text spotting performance on SCUT-CTW1500. The amount of long arbitrarily-shaped data is limited. Our recognition decoder needs more training data than the 1D-Attention [1] and CTC [10] and so it is not well trained yet. However, the 1-NED result and "Full" result shown in Table 6 narrow the gaps between our method and ABCNet v2, which suggests that the errors mainly occur in individual characters.

5. Conclusion

In this paper, we propose SwinTextSpotter. To the best of our knowledge, SwinTextSpotter is the first successful attempt using Transformer-based method and set-prediction scheme for end-to-end scene text spotting. With the core idea to make text recognition as a part of detection, the proposed model tightly couples the detection and recognition instead of only sharing information in the backbone. The proposed *Recognition Conversion* enables the suppression of background noise in the recognition features by making the detection results differentiable with respect to the recognition loss. Such design greatly simplifies the text spotter framework by removing the rectification module and enables the joint optimization of both the detection and the recognition module toward better spotting performance without character-level annotation. Extensive experiments on public benchmarks demonstrate that SwinTextSpotter can achieve superior performance in end-to-end scene text spotting on arbitrarily-shaped text and multi-lingual text.

Acknowledgement This research is supported in part by NSFC (Grant No.: 61936003) and GD-NSF (no.2017A030312006, No.2021A1515011870).

References

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014. 8

[2] Alessandro Bissacco, Mark Cummins, Yuval Netzer, and Hartmut Neven. Photoocr: Reading text in uncontrolled conditions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 785–792, 2013. 2

[3] Fred L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):567–585, 1989. 3, 5

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020. 3, 4

[5] 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 *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1571–1576. IEEE, 2019. 6

[6] Chee-Kheng Ch'ng, Chee Seng Chan, and Cheng-Lin Liu. Total-text: toward orientation robustness in scene text detection. *International Journal on Document Analysis and Recognition (IJDAR)*, 23(1):31–52, 2020. 2, 5, 6

[7] Shancheng Fang, Hongtao Xie, Yuxin Wang, Zhendong Mao, and Yongdong Zhang. Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7098–7107, 2021. 2

[8] Wei Feng, Wenhao He, Fei Yin, Xu-Yao Zhang, and Cheng-Lin Liu. Textdragon: An end-to-end framework for arbitrary shaped text spotting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9076–9085, 2019. 2, 3, 5, 6, 7

[9] Lluís Gómez and Dimosthenis Karatzas. Textproposals: a text-specific selective search algorithm for word spotting in the wild. *Pattern Recognition*, 70:60–74, 2017. 1

[10] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 369–376, 2006. 8

[11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2961–2969, 2017. 4

[12] Tong He, Zhi Tian, Weilin Huang, Chunhua Shen, Yu Qiao, and Changming Sun. An end-to-end textspotter with explicit alignment and attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5020–5029, 2018. 1, 3

[13] Jie Hu, Liujuan Cao, Yao Lu, ShengChuan Zhang, Yan Wang, Ke Li, Feiyue Huang, Ling Shao, and Rongrong Ji. Istr: End-to-end instance segmentation with transformers. *arXiv preprint arXiv:2105.00637*, 2021. 2, 3, 4

[14] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Reading text in the wild with convolutional neural networks. *International Journal of Computer Vision*, 116(1):1–20, 2016. 1

[15] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. *MIT Press*, 2015. 2, 3

[16] Max Jaderberg, Andrea Vedaldi, and Andrew Zisserman. Deep features for text spotting. In *European Conference on Computer Vision*, pages 512–528. Springer, 2014. 1

[17] Xu Jia, Bert De Brabandere, Tinne Tuytelaars, and Luc V Gool. Dynamic filter networks. *Advances in Neural Information Processing Systems*, 29:667–675, 2016. 4

[18] Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, et al. Icdar 2015 competition on robust reading. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 1156–1160. IEEE, 2015. 2, 5, 6

[19] Dimosthenis Karatzas, Faisal Shafait, Seiichi Uchida, Masakazu Iwamura, Lluis Gomez i Bigorda, Sergi Robles Mestre, Joan Mas, David Fernandez Mota, Jon Almazan Almazan, and Lluis Pere De Las Heras. Icdar 2013 robust reading competition. In *2013 12th International Conference on Document Analysis and Recognition*, pages 1484–1493. IEEE, 2013. 6

[20] Hui Li, Peng Wang, and Chunhua Shen. Towards end-to-end text spotting with convolutional recurrent neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5238–5246, 2017. 1, 3

[21] M. Liao, P. Lyu, M. He, C. Yao, W. Wu, and X. Bai. 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, 2021. 1, 2, 3, 5, 6, 7

[22] Minghui Liao, Guan Pang, Jing Huang, Tal Hassner, and Xiang Bai. 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*, pages 706–722. Springer, 2020. 2, 3, 5, 6, 7

[23] Minghui Liao, Baoguang Shi, and Xiang Bai. Textboxes++: A single-shot oriented scene text detector. *IEEE Transactions on Image Processing*, 27(8):3676–3690, 2018. 1

[24] Minghui Liao, Baoguang Shi, Xiang Bai, Xinggang Wang, and Wenyu Liu. Textboxes: A fast text detector with a single deep neural network. In *Thirty-first AAAI Conference on Artificial Intelligence*, 2017. 2

[25] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2117–2125, 2017. 4

[26] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2980–2988, 2017. 4

[27] Xuebo Liu, Ding Liang, Shi Yan, Dagui Chen, Yu Qiao, and Junjie Yan. Fots: Fast oriented text spotting with a unified network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5676–5685, 2018. 1, 2, 3, 6

[28] Yuliang Liu, Hao Chen, Chunhua Shen, Tong He, Lianwen Jin, and Liangwei Wang. Abcnet: Real-time scene text spotting with adaptive bezier-curve network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9809–9818, 2020. 2, 3, 5, 6, 7

[29] 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. 2, 5, 6

[30] Yuliang Liu, Chunhua Shen, Lianwen Jin, Tong He, Peng Chen, Chongyu Liu, and Hao Chen. Abcnet v2: Adaptive bezier-curve network for real-time end-to-end text spotting. *arXiv preprint arXiv:2105.03620*, 2021. 2, 3, 6, 7

[31] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *International Conference on Computer Vision (ICCV)*, 2021. 3, 4

[32] Pengyuan Lyu, Minghui Liao, Cong Yao, Wenhao Wu, and Xiang Bai. Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 67–83, 2018. 1, 2, 3

[33] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571. IEEE, 2016. 4

[34] Nibal Nayef, Fei Yin, Imen Bizid, Hyunsoo Choi, Yuan Feng, Dimosthenis Karatzas, Zhenbo Luo, Umapada Pal, Christophe Rigaud, Joseph Chazalon, et al. 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 1, pages 1454–1459. IEEE, 2017. 6

[35] Lukáš Neumann and Jiří Matas. Real-time lexicon-free scene text localization and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(9):1872–1885, 2015. 1

[36] Nguyen Nguyen, Thu Nguyen, Vinh Tran, Triet Tran, Thanh Ngo, Thien Nguyen, and Minh Hoai. Dictionary-guided scene text recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2, 5, 6, 7

[37] Liang Qiao, Ying Chen, Zhanzhan Cheng, Yunlu Xu, Yi Niu, Shiliang Pu, and Fei Wu. Mango: A mask attention guided one-stage scene text spotter. In *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI)*, pages 2467–2476, 2021. 6, 7

[38] Liang Qiao, Sanli Tang, Zhanzhan Cheng, Yunlu Xu, Yi Niu, Shiliang Pu, and Fei Wu. Text perceptron: Towards end-to-end arbitrary-shaped text spotting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11899–11907, 2020. 2

[39] Siyang Qin, Alessandro Bissacco, Michalis Raptis, Yasuhisa Fujii, and Ying Xiao. Towards unconstrained end-to-end text spotting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4704–4714, 2019. 2, 3, 7

[40] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 28:91–99, 2015. 4

[41] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 658–666, 2019. 4

[42] Xuejian Rong, Bing Li, J Pablo Munoz, Jizhong Xiao, Aries Arditi, and Yingli Tian. Guided text spotting for assistive blind navigation in unfamiliar indoor environments. In *International Symposium on Visual Computing*, pages 11–22. Springer, 2016. 1

[43] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(11):2298–2304, 2016. 2

[44] Russell Stewart, Mykhaylo Andriluka, and Andrew Y Ng. End-to-end people detection in crowded scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2325–2333, 2016. 4

[45] Peize Sun, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka, Lei Li, Zehuan Yuan, Changhu Wang, et al. Sparse r-cnn: End-to-end object detection with learnable proposals. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14454–14463, 2021. 2, 3, 4

[46] Yipeng Sun, Zihan Ni, Chee-Kheng Chng, Yuliang Liu, Canjie Luo, Chun Chet Ng, Junyu Han, Errui Ding, Jingtuo Liu, Dimosthenis Karatzas, et al. Icdar 2019 competition on large-scale street view text with partial labeling-rrc-lsvt. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1557–1562. IEEE, 2019. 6

[47] Zhi Tian, Chunhua Shen, and Hao Chen. Conditional convolutions for instance segmentation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 282–298. Springer, 2020. 4

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017. 2, 4, 5

[49] Hao Wang, Pu Lu, Hui Zhang, Mingkun Yang, Xiang Bai, Yongchao Xu, Mengchao He, Yongpan Wang, and Wenyu

Liu. All you need is boundary: Toward arbitrary-shaped text spotting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12160–12167, 2020. 2, 3

[50] Hsueh-Cheng Wang, Chelsea Finn, Liam Paull, Michael Kaess, Ruth Rosenholtz, Seth Teller, and John Leonard. Bridging text spotting and slam with junction features. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3701–3708. IEEE, 2015. 1

[51] Jiapeng Wang, Chongyu Liu, Lianwen Jin, Guozhi Tang, Jiaxin Zhang, Shuaitao Zhang, Qianying Wang, Yaqiang Wu, and Mingxiang Cai. Towards robust visual information extraction in real world: New dataset and novel solution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2738–2745, 2021. 1

[52] Kai Wang, Boris Babenko, and Serge Belongie. End-to-end scene text recognition. In *2011 International Conference on Computer Vision*, pages 1457–1464. IEEE, 2011. 2

[53] Pengfei Wang, Chengquan Zhang, Fei Qi, Shanshan Liu, Xiaoqiang Zhang, Pengyuan Lyu, Junyu Han, Jingtuo Liu, Errui Ding, and Guangming Shi. Pgnet: Real-time arbitrarily-shaped text spotting with point gathering network. *arXiv preprint arXiv:2104.05458*, 2021. 7

[54] Wenhai Wang, Xuebo Liu, Xiaozhong Ji, Enze Xie, Ding Liang, ZhiBo Yang, Tong Lu, Chunhua Shen, and Ping Luo. Ae textspotter: Learning visual and linguistic representation for ambiguous text spotting. In *European Conference on Computer Vision*, pages 457–473. Springer, 2020. 2, 3, 6

[55] Wenhai Wang, Enze Xie, Xiang Li, Xuebo Liu, Ding Liang, Yang Zhibo, Tong Lu, and Chunhua Shen. Pan++: Towards efficient and accurate end-to-end spotting of arbitrarily-shaped text. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 2, 3, 6, 7

[56] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8440–8449, 2019. 3

[57] Yuxin Wang, Hongtao Xie, Shancheng Fang, Jing Wang, Shenggao Zhu, and Yongdong Zhang. From two to one: A new scene text recognizer with visual language modeling network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14194–14203, 2021. 2

[58] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 2(1-3):37–52, 1987. 4

[59] Haiping Wu, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. Cvt: Introducing convolutions to vision transformers. *arXiv preprint arXiv:2103.15808*, 2021. 4

[60] Linjie Xing, Zhi Tian, Weilin Huang, and Matthew R Scott. Convolutional character networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9126–9136, 2019. 6, 7

[61] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Xiyang Dai, Bin Xiao, Lu Yuan, and Jianfeng Gao. Focal self-attention for local-global interactions in vision transformers. *arXiv preprint arXiv:2107.00641*, 2021. 5

[62] Deli Yu, Xuan Li, Chengquan Zhang, Tao Liu, Junyu Han, Jingtuo Liu, and Errui Ding. Towards accurate scene text recognition with semantic reasoning networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12113–12122, 2020. 2

[63] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016. 4

[64] Chongsheng Zhang, Yuefeng Tao, Kai Du, Weiping Ding, Bin Wang, Ji Liu, and Wei Wang. Character-level street view text spotting based on deep multi-segmentation network for smarter autonomous driving. *IEEE Transactions on Artificial Intelligence*, pages 1–1, 2021. 1

[65] Peng Zhang, Yunlu Xu, Zhanzhan Cheng, Shiliang Pu, Jing Lu, Liang Qiao, Yi Niu, and Fei Wu. Trie: End-to-end text reading and information extraction for document understanding. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1413–1422, 2020. 1

[66] Rui Zhang, Yongsheng Zhou, Qianyi Jiang, Qi Song, Nan Li, Kai Zhou, Lei Wang, Dong Wang, Minghui Liao, Mingkun Yang, et al. Icdar 2019 robust reading challenge on reading chinese text on signboard. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1577–1581. IEEE, 2019. 2, 5, 6

[67] Humen Zhong, Jun Tang, Wenhai Wang, Zhibo Yang, Cong Yao, and Tong Lu. Arts: Eliminating inconsistency between text detection and recognition with auto-rectification text spotter. *arXiv preprint arXiv:2110.10405*, 2021. 2, 3

[68] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 3