Sequential Transformer for End-to-End Video Text Detection

Jun-Bo Zhang, Meng-Biao Zhao, Fei Yin, Cheng-Lin Liu
SKL of MAIS, Institute of Automation of Chinese Academy of Sciences,
and School of Artificial Intelligence, University of Chinese Academy of Sciences
Beijing, China
{zhangjunbo2020,zhaomengbiao2017}@ia.ac.cn, {fyin,liucl}@nlpr.ia.ac.cn

Abstract

In existing methods of video text detection, the detection and tracking branches are usually independent of each other, and although they jointly optimize the backbone network, the tracking-by-detection paradigm still needs to be used during the inference stage. To address this issue, we propose a novel video text detection framework based on sequential transformer, which decodes detection and tracking tasks in parallel, without explicitly setting up a tracking branch. To achieve this, we first introduce the concept of instance query, which learns long-term context information in the video sequence. Then, based on the instance query, the transformer decoder is used to predict the entire box and mask sequence of the text instance in one pass. As a result, the tracking task is realized naturally. In addition, the proposed method can be applied to the scene text detection task seamlessly, without modifying any modules. To the best of our knowledge, this is the first framework to unify the tasks of scene text detection and video text detection. Our model achieves state-of-the-art performance on four video text datasets (YVT, RT-1K, BOVText, and BiRViT-1K), and competitive results on three scene text datasets (CTW1500, MSRA-TD500, and Total-Text). The code is available at https://github.com/zjb-1/SeqVideoText.

1. Introduction

Video text detection aims to localize the text instance in the image frames of video and construct a trajectory for each text instance. As a fundamental task in computer vision, it has been widely applied to video content analysis, video retrieval, and scene understanding. Compared to static scenes, video scenes contain temporal information and richer content, making video text more complex. In addition to the problems of text in static scenes, video text also faces problems such as motion blur, lighting changes, and occlusions, which make video text processing more challenging.

Most early methods [36, 47, 56] treat video text detection as a two-stage task: first performing single-frame text detection, and then applying tracking techniques [1, 15, 17] to associate detection results. However, these methods ignore the mutual supervision between detection and tracking tasks as well as the temporal information in videos. Recently, some methods [10, 11, 48, 49] takes a leap forward towards a unified end-to-end architecture by sharing a convolutional neural network (CNN) backbone and employing a feature cropping mechanism to extract the relevant area of interest for the tracking head. Although these methods have the advantage of task collaboration and improve the performance of the model, the detection and tracking tasks in the framework are still independent except for jointly training the backbone network. Specifically, the tracking head is usually trained using the detection ground-truth, and thus it is not optimized for the prediction of the detection head. Furthermore, the tracking-by-detection paradigm is still employed in the inference stage, which might lead to error accumulation. Recently, Wu *et al.* [45] try to simplify the video detection process by using a transformer [39] based framework, which only utilizes temporal information from adjacent two frames, and the tracking task still relies on IoU-based matching rules, however. Thus, constructing a concise and effective end-to-end video text detection framework remains a challenge.

Here, we conduct an in-depth analysis of the video text detection task. Videos contain richer temporal information than single frame image, and are highly context-dependent, which could provide useful cues for text detection and tracking. Essentially, both text detection and text tracking are about similarity learning between samples: the former focuses on learning similarity between pixels in the image, while the latter learns similarity of text instances between adjacent frames. Therefore, combining these two tasks into one framework is desired. Recently, transformer based models have made significant progress in computer vision tasks due to their ability to model long-term dependencies, with their core mechanism, self-attention, learning

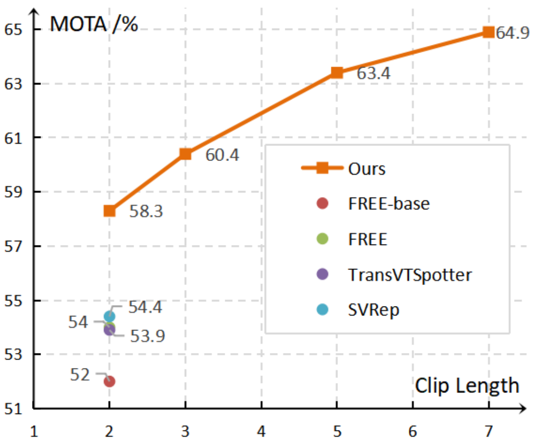

Figure 1. The tracking performance comparisons of video text detection methods on YVT dataset. Our method significantly outperforms the previous methods with the same clip length. After using longer video clips, the tracking performance is further improved. This is not possible with the other methods, because they only support two input frames at a time.

similarity between global features. Therefore, we suggest that it can be applied to video text detection for handling temporal information in multiple frames.

In this paper, we propose a video text detection framework based on sequential transformer, which decodes detection and tracking tasks in parallel via sequence prediction. To achieve this, we introduce the concept of *instance query* to represent the sequence feature of each text instance. In the iteration process of the transformer decoder, the shared *instance query* is decomposed into *object queries* at the frame level, which are used to continuously refine the specific information of the same text instance in different frames. These object queries are kept on each frame and used to predict the bounding box sequence. At the same time, the instance query aggregates the temporal information of the text instance from object queries and predicts the mask sequence of each text instance. Since the text sequences are directly generated from the decoder, it naturally realizes text matching across frames, eliminating the need for post-processing operations.

In experiments on four video text datasets (YVT [30], RT-1K [32], BOVText [45], and BiRViT-1K), the proposed method has achieved state-of-the-art performance. Fig. 1 compares the tracking performance of our method on the YVT dataset with previous methods. With the same input sequence length, our method achieves the best performance, outperforming the previous methods by 3.9%. By increasing the length of the input sequence, the performance can be been further improved.

Our framework can be seamlessly applied to scene text detection task without modifying any modules, as there is no explicit tracking process. We also demonstrate its performance on scene text datasets.

In summary, our contributions are in three folds:

(1) We propose a novel end-to-end video text detection framework based on sequential transformer. The model can effectively capture the temporal contextual information of video sequence, and decodes the text detection and tracking tasks in parallel. It eliminates the dependency on separate tracking branch and other manual components (such as NMS) and is therefore more concise than previous methods.

(2) Benefiting from the implicit tracking process, the proposed method can accomplish both scene text detection and video text detection tasks without modifying any components, unifying the two tasks for the first time.

(3) The proposed method is demonstrated effective for both detection and tracking, yielding state-of-the-art performance on four video text datasets and competitive performance on three scene text datasets.

2. Related Work

Video text detection task is an extension of scene text detection task, requiring not only single frame text detection, but also text tracking. Therefore, we review related works of scene text detection and video text detection.

2.1. Scene Text Detection

Scene text detection methods based on deep neural networks can be divided into two categories: regression-based methods and segmentation-based methods. Regression based methods [14, 20, 27, 37, 54] adopt similar ideas to generic object detection with some text-specific modifications. For example, RRPN [27] detects multi-oriented text by using rotated anchors. EAST [54] applies pixel-level regression with angle prediction for multi-oriented text instances. The detected results of such methods are generally quadrilaterals or rotated rectangles. To detect arbitrarily shaped texts, segmentation-based methods have been proposed [4, 5, 8, 22, 40, 41]. As examples, PixelLink [8] classifies text at the pixel-level and predicts the connection relationship between pixels to aggregate text regions, PSENet [40] proposed a new post-processing algorithm to segment text instances which are close to each other.

Recently, transformer [39] based models have made great achievements in computer vision tasks. For example, DETR [6] presents a novel transformer-based framework for object detection. It eliminates hand-designed components such as the proposal anchors and NMS, making the pipeline very succinct. Transformer-based scene text detection and recognition methods [16, 35, 53] have been proposed. Our model is proposed for video text detection, and can directly degenerate into a transformer-based scene text detector.

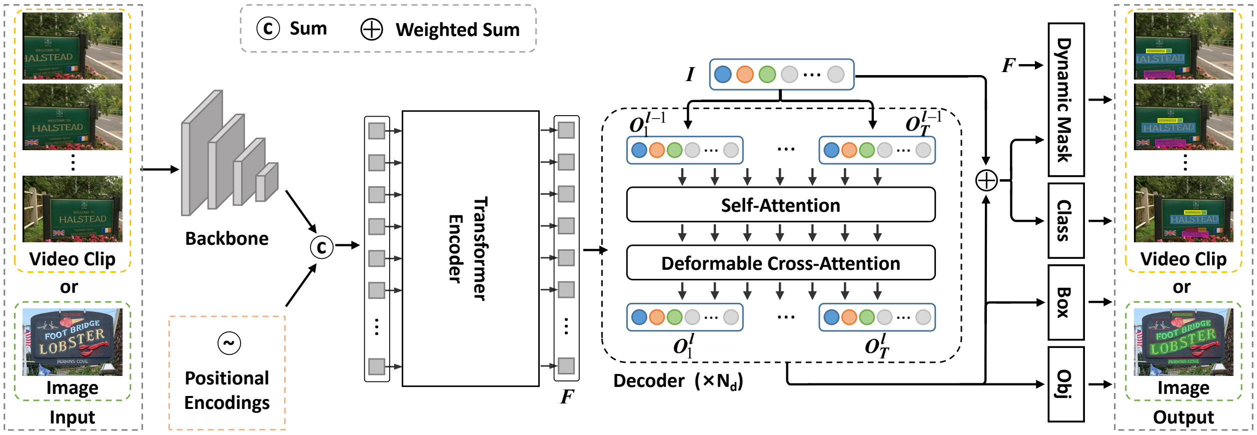

Figure 2. An overview of the proposed framework. Given a video clip, the model decodes the text detection and tracking tasks in parallel through sequence prediction, simultaneously predicting the box sequence and mask sequence of a text instance in one pass. "Dynamic Mask", "Class", "Box", and "Obj" denote the dynamic convolution mask head, class head, box head, and object head, respectively.

2.2. Video Text Detection

Early video text detectors adopted a two-stage approach separating detection and tracking. The method of Tian *et al.* [36] first detects scene texts in individual frames, then integrates the detection results into the tracking trajectory by dynamic programming. The method of Yang *et al.* [47] tracks proposals in adjacent frames with a motion-based method. However, these methods ignore the temporal contexts of video and the supervision information between detection and tracking.

Recently, some methods based on tracking-by-detection paradigm have been proposed, which integrate detection and tracking into a unified framework. Yu *et al.* [48] proposed the first end-to-end video text detection model with online tracking, in which the detection and tracking tasks are bridged together by feature descriptor. Feng *et al.* [10] proposed a semantic feature descriptor to improve the robustness of detection and tracking. Gao *et al.* [11] leverage a spatiotemporal Siamese complementary module to suppress the missed detection of text instances and use a text similarity learning network to integrate the visual and semantic cues of the text instance into a unified representation. The method of Wu *et al.* [45] builds a transformer based text detector, adds a tracking decoder to predict text positions, and then obtain tracking results through IoU match. Although these methods have made great progress, they only utilize the context information from the adjacent two frames, ignoring the long-term temporal information in video, and the pipelines are complex, requiring multiple processing steps to complete tracking. The proposed method models multi-frame information at one time, and adopts sequence prediction to decode detection and tracking results in parallel, thus greatly simplifies the process.

3. Methodology

The proposed sequential transformer based method treats video text detection as a direct sequence prediction problem. It takes a video clip as input, and outputs the masks and bounding boxes sequence of each text instance in the video in order.

3.1. Network Architecture

As shown in Fig. 2, the proposed video text detection framework contains four main components: a backbone network for feature extraction, a transformer encoder to extract feature representations of each frame independently, a transformer decoder to model frame-level text features and sequence-level instance features, and four output heads to predict text instance sequence masks, sequence categories, text bounding boxes and confidence scores, respectively.

Backbone Network. We adopt a CNN based backbone for visual feature extraction. It takes T frames or images of $H_0 \times W_0$ as input, denoted as $\boldsymbol{x}_c \in \mathbb{R}^{T\times 3\times H_0\times W_0}$. The output features are denoted as $\{\boldsymbol{f}_t\}_{t=1}^{T}$ ($\boldsymbol{f}_t \in \mathbb{R}^{d\times H\times W}$).

Transformer Encoder. First, a 1×1 convolution is used on the feature maps to reduce the channels of the $\boldsymbol{f}_t$ to $C = 256$. After adding the positional encoding [6], we adopt deformable transformer encoder [55] to model the similarity among pixels in each frame, and get the output feature maps $\{\boldsymbol{F}_t\}_{t=1}^{T}$.

Transformer Decoder. Considering the rich temporal information and strong correlation between adjacent frames in video, the same text instance in different frames can be considered as a whole. Motivated by SeqFormer [43], we introduce Instance Query $\mathbf{I}_q \in \mathbb{R}^C$ to represent each text instance sequence, which is the learnable embedding. Since the appearance and position of the same text instance may change in different frames, we decompose the instance query into T frame-level object queries $\mathbf{O} = \{\mathbf{O}_t\}_{t=1}^T$ ($\mathbf{O}_t \in \mathbb{R}^C$), which correspond to each frame to learn accurate text feature representation.

During the iteration process of the decoder layers, the object query $\mathbf{O}_t$ continuously perceives text features from the frame feature map $\boldsymbol{F}_t$ in a coarse-to-fine manner:

$$\mathbf{O}_t^l = \begin{cases} \mathbf{I}_q, & l = 0 \\ \mathrm{DeformAtten}(\mathbf{O}_t^{l-1}, \boldsymbol{F}_t), & l \geq 1 \end{cases} \tag{1}$$

where $\mathbf{O}_t^l$ is the object query on frame t from the l-th decoder layer, and DeformAtten represents the deformable attention module, which reduces computational complexity by assigning a small fixed number of key points for each query. At the same time, the instance query weights the object queries in the time dimension to aggregate the temporal features of text instance:

$$\mathbf{I}_q^l = \mathbf{I}_q^{l-1} + \sum_{t=1}^{T} \mathrm{Softmax}\left(\boldsymbol{W} \cdot \mathbf{O}_t^l\right) \mathbf{O}_t^l, \tag{2}$$

where $\boldsymbol{W} \in \mathbb{R}^C$ is the learnable weights. Set the number of predicted targets in each frame to N, after N_d decoder layers, we will get N instance embeddings with text sequence information and T object embeddings $\{\mathbf{OE}_\mathrm{t}\}_{t=1}^T$ ($\mathbf{OE} \in \mathbb{R}^{\mathrm{N}\times\mathrm{C}}$) with specific text position information for each frame.

Output Heads. We add mask head, class head, box head and object head on the top of the decoder outputs. Class head is a linear mapping layer used to predict the category of each instance query. Given an instance embedding with index $\sigma(i)$, class head predicts that it is class c_i with probability $\hat{p}_{\sigma(i)}(c_i)$.

Box head is a 3-layer perceptron with ReLU activation function. For each object query, the box head predicts the box center coordinates and its height and width relative to the image size. For the instance with index $\sigma(i)$, we denote the predicted boxes sequence as $\hat{b}_{\sigma(i)} = \{\hat{b}_{(\sigma(i),1)}, \ldots, \hat{b}_{(\sigma(i),T)}\}$.

Object head is a linear mapping layer used to predict the confidence score of each object query, which is a fine-grained target discrimination operation compared with the class head. In cases of text occlusion, the object head can assist in identifying the text that disappears in the text sequence. For the instance with index $\sigma(i)$, we denote the predicted object sequence as $\hat{o}_{\sigma(i)} = \{\hat{o}_{(\sigma(i),1)}, \ldots, \hat{o}_{(\sigma(i),T)}\}$.

Mask head is a dynamic convolution network used to predict the text masks sequence. Considering the instance embedding aggregates the long-temporal information of text instance, which is richer representation of text feature, we use it to generate the masks sequence. Following [38], a 3-layer feed forward network encodes the instance embedding into parameters ω_i of mask head, which has three 8-channel 1×1 convolution layers. And an FPN-like module is used to fuse multi-scale feature maps from transformer encoder and generate feature maps sequence $\{\hat{\boldsymbol{F}}_{mask}^1, \ldots, \hat{\boldsymbol{F}}_{mask}^T\}$, where $\hat{\boldsymbol{F}}_{mask}^t \in \mathbb{R}^{(8\times\frac{H}{8}\times\frac{W}{8})}$. Moreover, $\hat{\boldsymbol{F}}_{mask}^t$ is combined with a map of the relative coordinates from the center of $\hat{b}_{(\sigma(i),t)}$ to provide a strong location hint for predicting the text mask. The new feature maps sequence $\{\tilde{\boldsymbol{F}}_{mask}^t\}_{t=1}^T$, $\tilde{\boldsymbol{F}}_{mask}^t \in \mathbb{R}^{(10\times\frac{H}{8}\times\frac{W}{8})}$, is sent to the mask head to predict the masks sequence:

$$\{\hat{m}_{i,t}\}_{t=1}^T = \{\mathrm{MaskHead}(\tilde{\boldsymbol{F}}_{mask}^t, \omega_i)\}_{t=1}^T. \tag{3}$$

3.2. Text Sequence Matching

Our model infers N fixed-size prediction sequences in a single pass through the decoder, where each sequence contains T objects. Let us denote by $\hat{\boldsymbol{y}}_i = \{\hat{\boldsymbol{y}}_i\}_{i=1}^N$ the predicted text instance sequences, and $\boldsymbol{y}$ the ground truth of text instance sequences. Each element i of the ground truth set is denoted as $\boldsymbol{y}_i = \{c_i, (b_{i,1}, \ldots, b_{i,T}), (o_{i,1}, \ldots, o_{i,T})\}$, where c_i is the target class label including ϕ, $b_{i,t} \in [0,1]^4$ is a vector that defines ground truth box center coordinates and its relative height and width in the frame t, and $o_{i,t}$ is the object indication in the frame t, 1 if there is a text instance, 0 otherwise. In the training process, bipartite matching between the ground truth and the prediction is conducted in the sequence-level, by searching for a permutation of N elements $\sigma \in S_N$ with the lowest cost by Hungarian algorithm [18]:

$$\hat{\sigma} = \underset{\sigma \in S_N}{\mathrm{argmin}} \sum_{i}^{N} \mathcal{L}_{\mathrm{match}}(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_{\sigma(i)}), \tag{4}$$

where $\mathcal{L}_{match}$ is a pair-wise matching cost between ground truth sequence y_i and text prediction sequence with index $\sigma(i)$:

$$\begin{aligned} \mathcal{L}_{\mathrm{match}}(\boldsymbol{y}_i, \hat{\boldsymbol{y}}_{\sigma(i)}) = -\hat{p}_{\sigma(i)}(c_i) + \mathcal{L}_{\mathrm{box}}(b_i, \hat{b}_{\sigma(i)}) \\ + \mathcal{L}_{\mathrm{object}}(o_i, \hat{o}_{\sigma(i)}), \end{aligned} \tag{5}$$

where $c_i \neq \phi$.

Given the optimal assignment $\hat{\sigma}$, we use the Hungarian loss to compute the loss for all matched pairs:

$$\begin{aligned} \mathcal{L}_{\mathrm{Hung}}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \sum_{i=1}^{N} \big[-\log\hat{p}_{\hat{\sigma}(i)}(c_i) + \mathcal{L}_{\mathrm{object}}(o_i, \hat{o}_{\hat{\sigma}(i)}) \\ + \mathbb{1}_{\{c_i \neq \phi\}} \mathcal{L}_{\mathrm{box}}(b_i, \hat{b}_{\hat{\sigma}(i)}) \\ + \mathbb{1}_{\{c_i \neq \phi\}} \mathcal{L}_{\mathrm{mask}}(m_i, \hat{m}_{\hat{\sigma}(i)}) \big]. \end{aligned} \tag{6}$$

The object loss $\mathcal{L}_{\mathrm{object}}$ is defined as Focal loss. The box loss $\mathcal{L}_{\mathrm{box}}$ is defined as a combination of the $\mathcal{L}_1$ loss and the generalized IoU loss [33]. And the $\mathcal{L}_{\mathrm{mask}}$ is a linear combination of the Dice [28] and Focal loss. These losses are normalized by the length of the input video clip.

3.3. Inference

Our method detects text instances by sequence prediction, which can directly obtain the text trajectory from the output of the model, without using a separate tracking branch for processing. However, the long time duration in video complicates the computation. To overcome this, we process a video clip containing T frames each time. In order to get the complete text trajectory on the whole video, the results of adjacent video clips need to be integrated. The process is as follows:

(1) **Intra-Clip:** After filtering out the sequences with text class probability less than θ_1, which do not contain text, we can get M text sequences. Further, we use the object confidence to perform fine discrimination in each frame, and text objects with confidence less than θ_2 will be ignored. Finally, the quadrilateral boxes of text can be obtained from the box sequences, and the polygon boxes can be generated from the mask sequences. In our experiments, θ_1 and θ_2 are set to 0.5.

(2) **Inter-Clip:** To correlate text instances in adjacent video clips, we overlap them by n frames. Assuming that the j-th video clip has M text sequences, and the $(j+1)$-th video clip has L text sequences, we calculate the pairwise mask matching cost between them, and use Hungarian algorithm to obtain the optimal matching results:

$$\tilde{\sigma} = \underset{\sigma \in \delta_M}{\operatorname{argmin}} \sum_{i}^{M} \sum_{t}^{n} \mathcal{L}'_{\text{mask}} \left(\hat{m}^{j}_{i,t}, \hat{m}^{j+1}_{(\sigma(i), T-n+t)} \right), \quad (7)$$

where $\mathcal{L}'_{\text{mask}}$ is defined as Dice loss. For the matched text sequence in the $(j+1)$-th video clip, we keep its trajectory ID in the previous video clip, otherwise we create a new ID for it.

3.4. Application to Scene Text Detection

Our model can be seamlessly applied to the scene text detection on single frame images. To do this, it only needs to modify the length of the input sequence to 1. In the inference process, we only need to keep the objects whose predicted text category probability greater than θ, which is set as 0.5 in all experiments.

Post Process	F1↑	IDsw↓	MOTA↑	MOTP↑
w/o CA	79.3	1386	46.8	79.4
w/ CA	79.3	52	64.9	79.4

Table 1. Ablation studies for clip association (CA) in post processing. "IDsw" denotes the number of ID Switches.

4. Experiments

4.1. Datasets and Metrics

We evaluate the text detection and tracking performance on four video text datasets, and three scene image datasets.
Video Text Datasets: YVT is harvested from YouTube, consists of 15 training videos and 15 testing videos. It contains web videos besides scene videos. **RT-1K** contains 1,000 English videos of road scenes, including 700 for training and 300 for testing. The text instances are annotated with rectangular boxes at line-level. **BOVText** is a large-scale, bilingual, open world video text dataset, which was collected from worldwide users of YouTube and KuaiShou. It contains 2,000+ videos, including 1,750,000 frames and 30+ open scenarios. **BiRViT-1K** is a large bilingual road scene video text dataset collected by ourselves. It includes 1000 videos, consisting of 300 Chinese videos, 300 English videos and 400 bilingual videos. These videos are split into training set and test set at 7:3 ratio. The text instances are annotated at line-level with quadrilateral boxes. The dataset is available at http://www.nlpr.ia.ac.cn/databases/CASIA-BiRViT1K/.
Scene Text Datasets: SynthText-150K [24] is a synthesized dataset for arbitrarily shaped scene text, which contains 94,723 images with multi-oriented texts and 54,327 images with curved texts. **CTW1500** is a line-level arbitrarily shaped scene text dataset, containing 1,000 training images and 500 testing images. **MSRA-TD500** is a multilingual text dataset in Chinese and English, containing 300 training images and 200 testing images. The text instances are annotated at line-level. **Total-Text** is a arbitrarily shaped scene text dataset, containing 1,255 training images and 300 testing images, annotated as polygon boxes at word-level.

Text detection task is evaluated by the metrics of precision (P), recall (R) and F1-score (F1), which follow the ICDAR competition. Text tracking task is evaluated by the metrics of the CLEAR-MOT [3], including multiple object tracking accuracy (MOTA) and multiple object tracking precision (MOTP). MOTA comprehensively evaluates the detection error and tracking error of the tracker, and MOTP evaluates the positioning ability of the tracker.

4.2. Implementation Details

Model settings. We use the ResNet-50 [12] pretrained on ImageNet dataset [9] as the backbone. The encoder and decoder of transformer follow DeformableDETR [55] and both contain 6 layers with a hidden dimension of 256. We set sampled key numbers K=4 and 8 attention heads for attention modules. And the number of instance queries N is set to 300.
Training. The model is implemented with PyTorch and optimized by AdamW [26] with an initial learning rate of 1e-4, the learning rates of the backbone and linear projections

#Clip Length	#Overlap Frames	Text Detection (%)			Text Tracking (%)		FPS
		P	R	F1	MOTA	MOTP	
3	2	77.6	74.9	76.2	60.4	78.4	5.9
5	2	78.9	77.0	77.9	63.4	79.1	8.4
7	2	**80.4**	**78.3**	**79.3**	64.9	**79.4**	10.1
7	4	80.4	78.1	79.2	**65.2**	79.2	5.4
7	6	80.4	78.3	79.3	65.1	79.3	2.5
9	2	79.7	77.9	78.8	64.3	79.3	**10.7**

Table 2. Ablation studies for input video clip length and overlapping frames length. **Note that "FPS" is the average processing speed of each video in the dataset.**

Method	P	R	F1	MOTA	MOTP
w/o object head	77.9	78.5	78.2	63.5	79.1
w/ object head	80.4	78.3	79.3	64.9	79.4

Table 3. Ablation studies for object head.

used for deformable attention modules are multiplied by a factor of 0.1. All experiments were run on a workstation with NVIDIA RTX A6000.

For scene text detection, following SwinTextSpotter [16], the model is first pretrained on a unified set of SynthText-150K, ICDAR 2013, ICDAR 2015 and Total-Text for 12 epochs, and the learning rate is decayed at the 6-th epoch by a factor of 0.1. Then we fine-tune the model on the corresponding datasets for 40 epochs. All the models were run on 4 GPUs with a batch size of 8.

For video text detection, we first re-train the pretrained model of scene text datasets for 12 epochs on a unified set of YVT, RT-1K, BOVText and BiRViT-1K. Then the model is fine-tuned on the corresponding video datasets for 12 epochs. The learning rates are both decayed at the 6-th epoch by a factor of 0.1. The input sequence length T is set to 7 by default. All models were run on 4 GPUs with a batch size of 4.

In the training process, the images are randomly rescaled, by resizing the shorter side to 608-800 pixels and the longer side to at most 1333 pixels.

Inference. We re-scale the shorter side of the images to 800 pixels while maintaining the aspect ratio. For video text detection task, we set the number of overlapping frames n of adjacent video clips as 2 by default.

4.3. Ablation Studies

We performed extensive experiments on the YVT dataset for analyzing the impact of different settings. Unless mentioned, the default sequence length and the number of overlapping frames are set as 7 and 2, respectively.

Clip Association. Because of the long duration of the video data, we adopted the segmented approach as described in Sec. 3.3. Therefore, we first analyze the impact of association between clips. As shown in Tab. 1, when clip association is used, the tracking metric MOTA is significantly improved, which is 18% (64.9% vs 46.8%) higher than that without association. This is because the association operation effectively matches the text trajectories between adjacent clips, thereby greatly reducing the number of ID Switches (1386 vs 52), improving the tracking stability.

Overlapping Frames Length. The impacts of video clip length and overlapping length are shown in Tab. 2. The results show that with a fixed input sequence length 7, the detection and tracking performance is hardly affected as the number of overlapping frames increases. However, the inference speed drops significantly with increasing overlapping length, because the number of clips that need to be calculated also increases, leading to a large number of repeated calculations.

Video Clip Length. To evaluate the importance of the long-term temporal information to our method, we experiment with models trained with different input video clip length. As shown in Tab. 2, with a fixed overlapping frames length 2, the detection and tracking performance gradually improves as the length of video clip increases, and the optimal performance is achieved when the length is 7. Compared with the length of 3, the F1-score increases by 3.1% and the MOTA increases by 4.5%. This indicates that long-term contextual information can help the model obtain stronger feature representations, which leads to better discovery and association of text instances.

In addition, as the length of the clip increases, the inference speed of the model on the video also gradually increases, reaching 10.1 fps at the length of 7, and then degrades. So we use the clip length of 7 and the number of overlapping frames of 2 as the default values in the experiments to tradeoff between the performance and speed.

Object Head. The object head can be used to accurately determine where text instances appear in the video sequence. To evaluate its effect, we trained two different models with or without it. As shown in Tab. 3, when the object head is added, the detection and tracking performance both improve, with F1-score and MOTA increased by 1.1% and 1.4%, respectively. This is because the object head can filter out false positives in the text instance sequence, improve the

Dataset	Method	Text Detection (%)			Text Tracking (%)	
		P	R	F1	MOTA	MOTP
RT-1K	EAST [54]	42.0	30.0	35.0	–	–
	FOTS [23]	45.0	36.0	40.0	–	–
	CTPN [37]	44.0	41.0	42.0	-29.8	17.0
	Reddy *et al.* [32]	44.0	41.0	42.0	-11.0	7.0
	FREE [7]	63.0	43.4	51.4	3.0	71.0
	Feng *et al.* [10]	**76.0**	43.0	54.9	–	–
	Ours	64.0	**58.9**	**61.3**	**41.0**	**74.8**
BOVText	EAST [54]	55.4	40.8	47.0	-21.6	75.8
	PSENet [40]	78.3	75.7	77.0	52.1	77.5
	DB [22]	84.3	**77.6**	80.8	53.2	78.3
	TransVTSpotter [45]	86.2	77.4	81.7	68.2	82.1
	Ours	**90.2**	76.5	**82.8**	**75.9**	**84.4**
BiRViT-1K	EAST* [54]	53.8	40.6	46.3	–	–
	PSENet* [40]	65.5	60.3	62.8	–	–
	TransVTSpotter* [45]	72.4	66.0	69.0	53.7	75.8
	Ours	**77.6**	**66.5**	**71.6**	**62.4**	**77.3**
YVT	Mosleh *et al.* [29]	79.0	72.0	75.0	–	–
	Shivakumara *et al.* [34]	79.0	73.0	76.0	–	–
	Wu *et al.* [44]	81.0	73.0	77.0	–	–
	Yu *et al.* [48]	89.3	71.1	79.2	–	–
	FREE [7]	**90.3**	**81.6**	**85.7**	54.0	78.0
	TransVTSpotter [45]	–	–	–	53.9	75.9
	SVRep [19]	–	–	–	54.4	74.2
	Ours	80.4	78.3	79.3	**64.9**	**79.4**

Table 4. Text detection and tracking results on four video text datasets. "*" denotes results produced by our implementation. The best result of each dataset is in **bold**.

Method	CTW1500			MSRA-TD500			Total-Text		
	P	R	F1	P	R	F1	P	R	F1
TextField [46]	83.0	79.8	81.4	87.4	75.9	81.3	84.3	83.9	84.1
PSENet [40]	84.8	79.7	82.2	–	–	–	84.0	78.0	80.9
LOMO [50]	89.2	69.6	78.4	–	–	–	88.6	75.7	81.6
CRAFT [2]	86.0	81.1	83.5	88.2	78.2	82.9	87.6	79.9	83.6
PAN [41]	86.4	81.2	83.7	84.4	83.8	84.1	89.3	81.0	85.0
DB [21]	86.9	80.2	83.4	91.5	79.2	84.9	87.1	82.5	84.7
ContourNet [42]	84.1	83.7	83.9	–	–	–	86.9	83.9	85.4
DRRG [51]	85.9	83.0	84.5	88.1	82.3	85.1	86.5	84.9	85.7
ABCNet V2 [25]	85.6	83.8	84.7	89.4	81.3	85.2	90.2	84.1	87.0
MOST [13]	–	–	–	90.4	82.7	86.4	–	–	–
TextBPN [52]	86.5	83.6	85.0	86.6	84.5	85.6	90.7	85.2	87.9
Raisi *et al.* † [31]	–	–	–	90.9	83.8	87.2	–	–	–
TESTR† [35]	**92.0**	82.6	87.1	–	–	–	**93.4**	81.4	86.9
Tian *et al.* † [35]	88.1	82.4	85.2	91.6	**84.8**	**88.1**	90.7	**85.7**	**88.1**
SwinTextSpotter† [16]	–	–	**88.0**	–	–	–	–	–	87.2
Ours†	89.0	**85.8**	87.4	**93.3**	81.8	87.2	88.4	85.1	86.7

Table 5. Text detection results on three scene text datasets. "†" indicates that the method is based on transformer.

precision and enhance the stability of the tracking process.

4.4. Comparison with the State of the Art

4.4.1 Video Text Detection and Tracking

We conducted experiments on four video text datasets RT-1K, BOVText, BiRViT-1K and YVT, and the text detection and tracking results are shown in Tab. 4. Our method achieves the best detection performance on RT-1K, BOVText and BiRViT-1K with F1-score of 61.3%, 82.8% and 71.6%, respectively, which are 6.4%, 1.1% and 2.6% higher than the previous state-of-the-art methods. These results validate the effectiveness of the transformer-based architec-

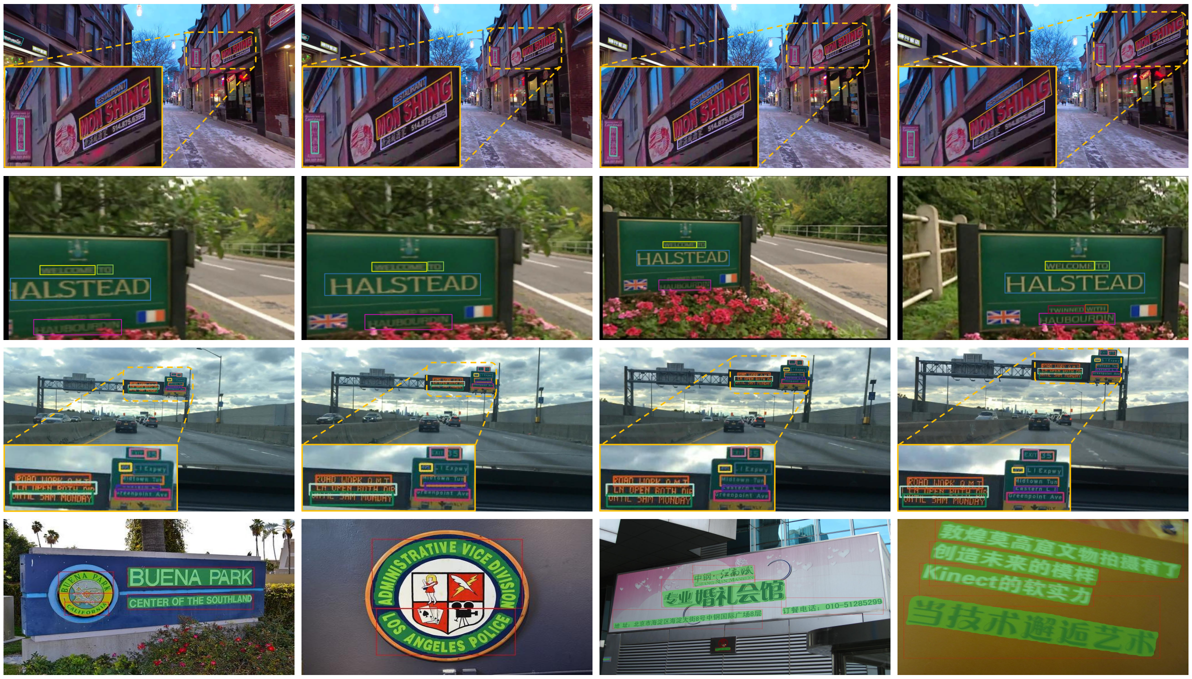

Figure 3. Examples of text detection and tracking results. First three rows: video text detection. Boxes with the same color belong to the same trajectory. Last row: scene text detection.

ture for text detection. In addition, our method achieves optimal tracking performance on all four datasets, significantly improving the MOTA and MOTP metrics. Specifically, the MOTA on four datasets are 41.0%, 75.9%, 62.4% and 64.9% respectively. These results verify the effectiveness of the proposed method based on sequence prediction in text tracking task, which does not require a separate tracking branch and is more concise than the previous methods. Fig. 3 shows some detection and tracking results on different datasets.

4.4.2 Scene Text Detection

Our model does not have an explicit tracking branch, making it easily to transfer to scene text detection task by simply setting the video clip length as 1. Therefore, we evaluate the single frame detection performance of the model on three scene text datasets CTW1500, MSRA-TD500 and Total-Text. As shown in Tab. 5, our detection model outperforms most previous CNN-based methods and achieves competitive performance compared with the recent scene text detectors based on transformer. Our model achieves F1-score of 87.4%, 87.2% and 86.7% on the three datasets, respectively. Some qualitative examples are shown in the last row of Fig. 3.

The above results demonstrate that our model can effectively unify scene text detection task and video text detection task in the same framework. We believe that our concise, elegant and effective framework will serve as a strong baseline to promote research in related fields such as video text detection, tracking and recognition.

5. Conclusion

In this paper, we proposed a novel end-to-end video text detection framework based on sequential transformer, which leverages the instance query to aggregate the long-term contextual information from the input video sequence and directly generates the entire masks sequence and boxes sequence of each text instance in one pass. Compared with the previous methods, the proposed method does not need to set explicit tracking branch, making the framework more concise. Notably, our method can be applied to scene text detection (from single frame images) without modifying any modules, thus unify scene text detection and video text detection tasks in the same framework. Our method achieves state-of-the-art results on four video text datasets and competitive results on three scene text datasets. We hope that our simple and effective framework can promote the research and applications in related fields in the future.

Acknowledgements

This work has been supported by the National Key Research and Development Program Grant 2020AAA0108003 and the National Natural Science Foundation of China (NSFC) Grant 61721004.

References

[1] Lukežič Alan, Tomáš Vojíř, Luka Čehovin, Jiří Matas, and Matej Kristan. Discriminative correlation filter tracker with channel and spatial reliability. *International Journal of Computer Vision*, 126(7):671–688, 2018. 1

[2] Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. Character region awareness for text detection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 9365–9374, 2019. 7

[3] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008:1–10, 2008. 5

[4] Y. Cai, C. Liu, P. Cheng, D. Du, L. Zhang, W. Wang, and Q. Ye. Scale-residual learning network for scene tex detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(7):2725–2738, 2021. 2

[5] Meng Cao, Can Zhang, Dongming Yang, and Yuexian Zou. All you need is a second look: Towards arbitrary-shaped text detection. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(2):758–767, 2022. 2

[6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision*, pages 213–229, 2020. 2, 3

[7] Zhanzhan Cheng, Jing Lu, Baorui Zou, Liang Qiao, Yunlu Xu, Shiliang Pu, Yi Niu, Fei Wu, and Shuigeng Zhou. Free: A fast and robust end-to-end video text spotter. *IEEE Transactions on Image Processing*, 30:822–837, 2020. 7

[8] Dan Deng, Haifeng Liu, Xuelong Li, and Deng Cai. Pixellink: Detecting scene text via instance segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 2

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 5

[10] Wei Feng, Fei Yin, Xu-Yao Zhang, and Cheng-Lin Liu. Semantic-aware video text detection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 1695–1705, 2021. 1, 3, 7

[11] Yuzhe Gao, Xing Li, Jiajian Zhang, Yu Zhou, Dian Jin, Jing Wang, Shenggao Zhu, and Xiang Bai. Video text tracking with a spatio-temporal complementary model. *IEEE Transactions on Image Processing*, 30:9321–9331, 2021. 1, 3

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 5

[13] 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 *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 8813–8822, 2021. 7

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

[15] Joao F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista. Exploiting the circulant structure of tracking-by-detection with kernels. In *Proceedings of the European Conference on Computer Vision*, pages 702–715, 2012. 1

[16] Mingxin Huang, Yuliang Liu, Zhenghao Peng, Chongyu Liu, Dahua Lin, Shenggao Zhu, Nicholas Yuan, Kai Ding, and Lianwen Jin. Swintextspotter: Scene text spotting via better synergy between text detection and text recognition. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 4593–4603, 2022. 2, 6, 7

[17] Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. Forward-backward error: Automatic detection of tracking failures. In *Proceedings of the International Conference on Pattern Recognition*, pages 2756–2759, 2010. 1

[18] Harold W Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955. 4

[19] Zhuang Li, Weijia Wu, Mike Zheng Shou, Jiahong Li, Size Li, Zhongyuan Wang, and Hong Zhou. Contrastive learning of semantic and visual representations for text tracking. *arXiv preprint arXiv:2112.14976*, 2021. 7

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

[21] Minghui Liao, Zhaoyi Wan, Cong Yao, Kai Chen, and Xiang Bai. Real-time scene text detection with differentiable binarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11474–11481, 2020. 7

[22] Minghui Liao, Zhisheng Zou, Zhaoyi Wan, Cong Yao, and Xiang Bai. Real-time scene text detection with differentiable binarization and adaptive scale fusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(1):919–931, 2022. 2, 7

[23] 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 Conference on Computer Vision and Pattern Recognition*, pages 5676–5685, 2018. 7

[24] 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 Conference on Computer Vision and Pattern Recognition*, pages 9809–9818, 2020. 5

[25] 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. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):8048–8064, 2021. 7

[26] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5

[27] Jianqi Ma, Weiyuan Shao, Hao Ye, Li Wang, Hong Wang, Yingbin Zheng, and Xiangyang Xue. Arbitrary-oriented scene text detection via rotation proposals. *IEEE Transactions on Multimedia*, 20(11):3111–3122, 2018. 2

[28] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *Proceedings of the International Conference on 3D vision*, pages 565–571, 2016. 4

[29] Ali Mosleh, Nizar Bouguila, and Abdessamad Ben Hamza. Automatic inpainting scheme for video text detection and removal. *IEEE Transactions on Image processing*, 22(11):4460–4472, 2013. 7

[30] Phuc Xuan Nguyen, Kai Wang, and Serge Belongie. Video text detection and recognition: Dataset and benchmark. In *Proceedings of the Winter Conference on Applications of Computer Vision*, pages 776–783, 2014. 2

[31] Zobeir Raisi, Mohamed A Naiel, Georges Younes, Steven Wardell, and John S Zelek. Transformer-based text detection in the wild. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 3162–3171, 2021. 7

[32] Sangeeth Reddy, Minesh Mathew, Lluis Gomez, Marçal Rusinol, Dimosthenis Karatzas, and CV Jawahar. Roadtext-1k: Text detection & recognition dataset for driving videos. In *Proceedings of the International Conference on Robotics and Automation*, pages 11074–11080, 2020. 2, 7

[33] 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 Conference on Computer Vision and Pattern Recognition*, pages 658–666, 2019. 4

[34] Palaiahnakote Shivakumara, Liang Wu, Tong Lu, Chew Lim Tan, Michael Blumenstein, and Basavaraj S Anami. Fractals based multi-oriented text detection system for recognition in mobile video images. *Pattern Recognition*, 68:158–174, 2017. 7

[35] Jingqun Tang, Wenqing Zhang, Hongye Liu, MingKun Yang, Bo Jiang, Guanglong Hu, and Xiang Bai. Few could be better than all: Feature sampling and grouping for scene text detection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 4563–4572, 2022. 2, 7

[36] Shu Tian, Wei-Yi Pei, Ze-Yu Zuo, and Xu-Cheng Yin. Scene text detection in video by learning locally and globally. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 2647–2653, 2016. 1, 3

[37] Zhi Tian, Weilin Huang, Tong He, Pan He, and Yu Qiao. Detecting text in natural image with connectionist text proposal network. In *Proceedings of the European Conference on Computer Vision*, pages 56–72, 2016. 2, 7

[38] Zhi Tian, Chunhua Shen, and Hao Chen. Conditional convolutions for instance segmentation. In *Proceedings of the European Conference on Computer Vision*, pages 282–298, 2020. 4

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017. 1, 2

[40] Wenhai Wang, Enze Xie, Xiang Li, Wenbo Hou, Tong Lu, Gang Yu, and Shuai Shao. Shape robust text detection with progressive scale expansion network. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 9336–9345, 2019. 2, 7

[41] 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 International Conference on Computer Vision*, pages 8440–8449, 2019. 2, 7

[42] Yuxin Wang, Hongtao Xie, Zheng-Jun Zha, Mengting Xing, Zilong Fu, and Yongdong Zhang. Contournet: Taking a further step toward accurate arbitrary-shaped scene text detection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 11753–11762, 2020. 7

[43] Junfeng Wu, Yi Jiang, Song Bai, Wenqing Zhang, and Xiang Bai. Seqformer: Sequential transformer for video instance segmentation. In *Proceedings of the European Conference on Computer Vision*, pages 553–569, 2022. 3

[44] Liang Wu, Palaiahnakote Shivakumara, Tong Lu, and Chew Lim Tan. A new technique for multi-oriented scene text line detection and tracking in video. *IEEE Transactions on Multimedia*, 17(8):1137–1152, 2015. 7

[45] Weijia Wu, Debing Zhang, Yuanqiang Cai, Sibo Wang, Jiahong Li, Zhuang Li, Yejun Tang, and Hong Zhou. A bilingual, openworld video text dataset and end-to-end video text spotter with transformer. *Advances in Neural Information Processing Systems*, 2021. 1, 2, 3, 7

[46] 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(11):5566–5579, 2019. 7

[47] Xue-Hang Yang, Wenhao He, Fei Yin, and Cheng-Lin Liu. A unified video text detection method with network flow. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 331–336, 2017. 1, 3

[48] Hongyuan Yu, Yan Huang, Lihong Pi, Chengquan Zhang, Xuan Li, and Liang Wang. End-to-end video text detection with online tracking. *Pattern Recognition*, 113:107791, 2021. 1, 3, 7

[49] Hongyuan Yu, Chengquan Zhang, Xuan Li, Junyu Han, Errui Ding, and Liang Wang. An end-to-end video text detector with online tracking. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 601–606, 2019. 1

[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 *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 10552–10561, 2019. 7

[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 *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 9699–9708, 2020. 7

[52] Shi-Xue Zhang, Xiaobin Zhu, Chun Yang, Hongfa Wang, and Xu-Cheng Yin. Adaptive boundary proposal network for arbitrary shape text detection. In *Proceedings of the International Conference on Computer Vision*, pages 1305–1314, 2021. 7

[53] Xiang Zhang, Yongwen Su, Subarna Tripathi, and Zhuowen Tu. Text spotting transformers. In *Proceedings of the Con-*

ference on Computer Vision and Pattern Recognition, pages 9519–9528, 2022. 2

[54] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. East: an efficient and accurate scene text detector. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 5551–5560, 2017. 2, 7

[55] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *Proceedings of the International Conference on Learning Representations*, 2021. 3, 5

[56] Ze-Yu Zuo, Shu Tian, Wei-yi Pei, and Xu-Cheng Yin. Multi-strategy tracking based text detection in scene videos. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 66–70, 2015. 1