A Bilingual, Open World Video Text Dataset and End-to-end Video Text Spotter with Transformer

Weijia Wu∗†
Zhejiang University
weijiawu@zju.edu.cn

Yuanqiang Cai∗
Beijing University of Posts and Telecommunications
caiyuanqiang@bupt.edu.cn

Debing Zhang
Kuaishou Technology

Sibo Wang
Kuaishou Technology

Zhuang Li
Kuaishou Technology

Jiahong Li
Kuaishou Technology

Yejun Tang
Kuaishou Technology

Hong Zhou‡
Zhejiang University

Abstract

Most existing video text spotting benchmarks focus on evaluating a single language and scenario with limited data. In this work, we introduce a large-scale, Bilingual, Open World Video text benchmark dataset (BOVText). There are four features for BOVText. Firstly, we provide 1,850+ videos with more than 1,650,000+ frames, 25 times larger than the existing largest dataset with incidental text in videos. Secondly, our dataset covers 30+ open categories with a wide selection of various scenarios, e.g., Life Vlog, Driving, Movie, etc. Thirdly, abundant text types annotation (i.e., title, caption or scene text) are provided for the different representational meanings in video. Fourthly, the BOVText provides bilingual text annotation to promote multiple cultures’ live and communication.

Besides, we propose an end-to-end video text spotting framework with Transformer, termed TransVTSpotter, which solves the multi-orient text spotting in video with a simple, but efficient attention-based query-key mechanism. It applies object features from the previous frame as a tracking query for the current frame and introduces a rotation angle prediction to fit the multi-orient text instance. On ICDAR2015(video), TransVTSpotter achieves the state-of-the-art performance with 44.2% MOTA, 13 fps. The dataset and code of TransVTSpotter can be found at [github.com/weijiawu/BOVText] and [github.com/weijiawu/TransVTSpotter] respectively.

1 Introduction

Text spotting [26, 16] has received increasing attention due to its numerous applications in computer vision, e.g., document analysis, image-based translation, image retrieval [41, 33], etc. With the advent of deep learning and abundance in digital data, reading text from images has made extraordinary progress in recent years with a lot of great public datasets [12, 17, 7] and algorithms [53, 70, 29, 25]. By contrast, video text spotting almost remains at a standstill for the lack of large-scale multidimensional practical datasets, which limited numerous applications of video text, e.g., video understanding [46], video retrieval [9], video text translation, and license plate recognition [1], etc.

∗Equal contribution.
†This work was done when Weijia Wu were interns in MMU, Kuaishou Technology, Beijing, China.
‡Corresponding author.
Video text spotting (VTS) is the task that requires simultaneously classifying, detecting, tracking and recognizing text instances in a video sequence. There have been a few previous works [61, 58] and datasets [35, 18] in the community for attempting to develop video text spotting. ICDAR2015 (Text in Videos) [17] was introduced during the ICDAR Robust Reading Competition in 2015 and mainly includes a training set of 25 videos (13k frames) and a test set of 24 videos (14k frames). The videos were categorized into seven scenarios: walking outdoors, searching for a shop in a shopping street, etc. YouTube Video Text (YVT) [35] dataset harvested from YouTube, contains 30 videos with 13k frames, 15 for training, and 15 for testing. The text content in the dataset mainly includes overlay scene text (e.g., street signs, business signs, words on shirt). RoadText-1K [36] are sampled from BDD100K [64], includes 700 videos (210k frames) for training and 300 videos for testing. The texts are all obtained from driving videos and match for driver assistance and self-driving systems. LSVTD [6] includes 100 text videos, 13 indoor (e.g., bookstore, shopping mall) and 9 outdoor (e.g., highway, city road) scenarios. However, as shown in Figure 1(a), most existing video text datasets are limited by the amount of training data (less than 300k frames), single video scenarios, and a single language. There are only a few outdoor scene text videos with 13k frames in ICDAR2015 (video). Similar situation for YVT, RoadText-1k, and LSVTD, the training set is limited and the dataset scenarios are single. This makes it difficult to evaluate the effectiveness of more advanced deep learning models for more open scenarios, such as game, sport and news report.

In this work, we contribute a large-scale, bilingual open-world benchmark dataset (BOVText) to the community for developing and testing video text spotting that can fare in a realistic setting. Our benchmark dataset has several advantages. Firstly, the large training set (i.e., 1,850+video and 1,650,000+ video frames) from Kuaishou and YouTube enables the development of deep design specific for video text spotting. Secondly, unlike the existing datasets, BOVText support 30+ open scenarios, including many new scenarios such as Sportscast (NBA, FIFA World Cup...), Life Vlog, Game, etc, as shown in Figure 1(a). These data is collected from the worldwide user of YouTube and Kuaishou. Thirdly, BOVText is the first benchmark for supporting abundant text types annotation. Caption, title, and scene text are separately tagged for the different representational meanings in the video. This made our BOVText has the potential to promote other video-and-language tasks, such as video understanding. Fourthly, bilingual text annotation (i.e., Chinese, English) is provided in BOVText to promote multiple cultures’ live and communication.

Except for the promising benchmark, we also proposed a simple, but effective video text spotter with transformer (TransVTSpotter). As shown in Figure 1(b), unlike previous methods that involve multiple steps, such as proposal generation, text aggregation, and post-processing (NMS), TransVTSpotter only requires two steps. 1) Text tracking: for each consecutive frame image, we obtain the multi-orient boxes tracking trajectories of text by boxes IoU matching between the predicted detection boxes [5] and the predicted tracking boxes [47], where the detection boxes are obtained by taking an object query as input, just like DETR [5]. And features from previously detected objects to form another
“track query” to discover associated objects (i.e., the predicted tracking boxes) on the current frames. Besides, an additional angle loss of multi-orient box and Hungarian angle cost are design to obtain the angle of multi-orient. 2) Text recognition: recognizing the tracked texts with attention-based text recognizer [28]. Without bells and whistles, TransVTSpotter achieves state-of-the-art performance on ICDAR2015 with 44.2% MOTA, 13 fps. The main contributions of this work are three folds:

(1) We propose a large-scale, bilingual and open world video text spotting benchmark named BOVText. The proposed dataset provides 1,850+ videos, 1,650,000 frames, open videos scenarios (e.g., Indoor, Outdoor, Game, Sport), abundant text types (i.e., title, caption or scene text), multi-stage tasks and is 25 times the existing largest dataset with incidental text.

(2) We first propose a new video text spotting framework with Transformer, termed TransVTSpotter, which solves the video multi-orient text spotting with a simple, but effective pipeline based on the tracking query-key mechanism and rotated boxes angle prediction.

(3) We evaluate and compare TransVTSpotter and other techniques for scene text detection, recognition, text tracking, and end-to-end video text spotting on BOVText and other existing datasets. Besides, a thorough analysis of performance on the proposed dataset is provided.

2 Related Work

2.1 End-to-End Text Spotting

For image-level text spotting, various methods [21, 13, 29] based on deep learning have been proposed and have improved the performance considerably. Li et al. [21] proposed the first end-to-end trainable scene text spotting method. The method successfully uses a RoI Pooling [37] to joint detection and recognition features. Liao et al. [29] propose a Mask TextSpotter which subtly refines Mask R-CNN and uses character-level supervision to detect and recognize characters simultaneously. Compared to text spotting in a static image, video text spotting methods are rare. Yin et al. [53] provides a detailed survey, summarizes text detection, tracking and recognition methods in video. Wang et al. [55] introduced an end-to-end video text recognition method through associations of texts in the current frame and several previous frames to obtain final results. Cheng et al. [6] propose a video text spotting framework by only recognizing the localized text one-time. Nguyen et al. [35] improves detection and recognition performance by temporal redundancy and linearly interpolate to recover missing detection results. Rong et al. [40] tracked video text using tracking-by-detection. An MSER detector was used to locate scene text character, which was used as a constraint to optimize the trajectory search. To promote video text spotting, we attempt to establish a standardized benchmark (BOVText), covering various open scenarios and bilingual text annotation.

2.2 Text Spotting Datasets for Images and Videos

The various and practical benchmark datasets [17, 39, 18, 7, 18] contribute to the huge success of scene text detection and recognition at the image level. ICDAR2015 [17] was provided from the ICDAR2015 Robust Reading Competition. Google glasses capture these images without taking care of position, so text in the scene can be in arbitrary orientations. ICDAR2017MLT [34] is a large-scale bilingual text dataset, which is composed of complete scene images which come from 9 languages. The COCO-Text dataset [49] is currently the largest dataset for scene text detection and recognition. It contains 50,000+ images for training and testing.

The development of video text spotting is limited in recent years due to the lack of efficient data sets. ICDAR 2015 Video [13] consists of 28 videos lasting from 10 seconds to 1 minute in indoors or outdoors scenarios. Limited videos (i.e., 13 videos) used for training and 15 for testing. Minetto Dataset [32] consists of 5 videos in outdoor scenes. The frame size is 640 x 480 and all videos are used for testing. YYT [35] contains 30 videos, 15 for training and 15 for testing. Different from the above two datasets, it contains web videos except for scene videos. USTB-VidTEXT [61] with only five videos mostly contain born-digital text sourced from Youtube. RoadText-1K [36] provides a driving videos dataset with 1000 videos. The 10-second long video clips in the dataset are sampled from BDD100K [64]. As shown in Table.1, the existing datasets contain a limited training set and single video scenarios. To promote the development of video text spotting, we create a large-scale, bilingual open-world benchmark dataset.
Figure 2: Distributions of BOVText. (a) The benchmark dataset covers a wide and open range of life scenes (32 open-domain categories). (b) Caption text (blue box) and scene text (red box) are distinguished in BOVText, which is favorable for downstream tasks.

2.3 Transformers in Vision

Transformer is first proposed in [48] as a new paradigm for machine translation. But recently, there is a popularity of using transformer architecture in vision tasks, such as detection [5, 72], segmentation [69], 3D data processing [68], video object tracking [52] and even backbone construction [10]. Lately, some works [57, 67, 52] show using a transformer in processing sequential visual data also make remarkable shots. MOTR [67] introduce the concept of track query and the contiguous query passing mechanism for multiple-object Tracking. VisTR [57] solves instance segmentation by learning the pixel-level similarity and instance tracking is to learn the similarity between instances. But for video text spotting or multi-orient text tracking, to the best of our knowledge, there are still no transformer-based solutions while it is intuitive for its good capacity in temporal processing. Here, we propose the TransVTSpotter method and provide an affirmative answer to that, which shows convincingly high performance on the popular benchmark.

3 BOVText Benchmark

3.1 Data Collection and Annotation

Data Collection. To obtain abundant videos with various text types, we first start by acquiring a large list of different scenarios with text (e.g., game scenario, travel scenario) using YouTube and KuaiShou - an online resource that contains billions of videos with various scene text from cartoon movies to human relation. Then, we choose 31 open-domain categories with 1 unknown category, i.e., Game, Home, Fashion, and Technology, as shown in Figure 2(a). With each raw video category, we first choose the video clips with text, then make two rounds of manual screening to remove the ordinary videos without scene text and caption text. As a result, we obtain 1,620,305 video frames, as shown in Table 1. Finally, to fair evaluation, we divide the dataset into two parts: the training set with 1,124,213 frames from 1,297 videos, and the testing set with 496,091 frames from 555 videos. As shown in Figure 2, different from the existing data sets, our dataset not only cares about scene text spotting in the real world, but also focuses on caption texts in the video. For the most part, caption text represents more global information than scene text, which is quite favorable for some downstream tasks, e.g., video understanding, video caption translation, etc.

Data Annotation. We invite a professional annotation team to label each video text with four kinds of description information: the rotated bounding box describing the location information, judging the tracking identification(ID) of the same text, identifying the content of the text information, and distinguishing the category of the caption, title or scene text. To save the annotation cost, we first sample the videos, annotate each sampled video frame, and then transform the annotation information from the sampled video frame to the unlabeled video frame by interpolation. Finally, we invite an audit team to carry out another round of annotation checks, and re-label part video frames with unqualified annotation. For video sampling, we use uniform sampling with a sampling
Table 1: Statistical Comparison. ‘D’, ‘R’, ‘T’, ‘S’ and ‘BI’ denotes the Detection, Recognition, Tracking, Spotting and bi-lingual text, respectively. ‘Incidental’ denotes indoor and outdoor scenarios in daily life (e.g., walking outdoors, driving). ‘Open’ refers to any scenarios, e.g., Game, Sport(NBA).

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Category</th>
<th>BI</th>
<th>Task</th>
<th>Video</th>
<th>Frames</th>
<th>Texts</th>
<th>Supported Scenario</th>
</tr>
</thead>
<tbody>
<tr>
<td>AcTV2-D[25]</td>
<td>Caption</td>
<td>D</td>
<td>D</td>
<td>8</td>
<td>1,843</td>
<td>5,133</td>
<td>News video</td>
</tr>
<tr>
<td>USTB-VidTEXT[28]</td>
<td>Caption</td>
<td>D</td>
<td>D&amp;S</td>
<td>5</td>
<td>27,670</td>
<td>41,932</td>
<td>Web video</td>
</tr>
<tr>
<td>YTVI[3]</td>
<td>Scene,</td>
<td>D&amp;R&amp;T&amp;S</td>
<td>30</td>
<td>15,500</td>
<td>16,620</td>
<td>Incidental: Cartoon, Outdoors(supermarket, shopping street, driving)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Caption</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ICDAR2015 V1[71]</td>
<td>Scene</td>
<td>D&amp;R&amp;T&amp;S</td>
<td>51</td>
<td>27,824</td>
<td>143,588</td>
<td>Incidental: Indoor(walking, driving, supermarket, shopping street)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Scene</td>
<td>✓</td>
<td>D&amp;R&amp;T&amp;S</td>
<td>100</td>
<td>66,700</td>
<td>569,300</td>
<td>Incidental: Indoor(school, mall, supermarket, hotel, etc.) Outdoor(driving)</td>
</tr>
<tr>
<td>RoadText-1k[36]</td>
<td>Scene</td>
<td>D&amp;R&amp;T&amp;S</td>
<td>1,000</td>
<td>300,000</td>
<td>1,280,613</td>
<td>Driving</td>
<td></td>
</tr>
<tr>
<td>BOVText(ours)</td>
<td>Scene,</td>
<td>D&amp;R&amp;T&amp;S</td>
<td>1,852</td>
<td>1,670,305</td>
<td>292,261</td>
<td>Open: Cartoon, Vlog(suppermarket, shopping street, driving), Travel(indoor and outdoor), Game(PUBG mobile...), Sport(NBA, world cup...), News, TV program, Education(campus, classroom, book...), Technology(introduction video, scientific propaganda,...)</td>
<td></td>
</tr>
</tbody>
</table>

In green refers to these scenarios only supported by BOVText.

frequency of 3 to sample all the videos in the dataset, and obtain the sampled video frame set. For sampling video frame annotation, each text instance is labeled in the same quadrilateral way as in the ICDAR2015[71]. In addition, the text instance also will be marked with two description information: the category of the caption, title or scene, the recognition content, and the tracking ID. For interpolation on unlabeled video frames, each text instance is marked with tracking ID and recognition content, so we can judge whether the texts in adjacent sampling frames are the same text with the same ID. For the same text instance, we first determine whether the text annotation of the sampled video frame is the starting or end frame of the text instance. If not, we look forward and backward for the starting and end position of the text instance and label it. Then we use the linear interpolation way to calculate the position of the text object in the middle of the unmarked video frame, and give tracking ID, recognition content, and category. For check and re-label bad cases, the linear interpolation shows a dissatisfied performance in some cases, e.g., the new text appears on starting frame, text suddenly disappears on ending frame, which are difficult to capture. Therefore, we invite an audit team to carry out another round of annotation checks. Around 150,000 video frames with unqualified annotation from 1,670,305 video frames are selected to refine, taking 20 men in three weeks. As a labor-intensive job, the whole labeling process takes 30 men in three months, i.e., 21,600 man-hours, to complete about 600,000 sampled video frame annotations.

3.2 Dataset Analysis

The statistic comparison between BOVText and other datasets are visualized in Figure[1](a). YouTube and summarized in Table[1]. Besides, we provide more detailed information in appendix, such as ‘data distribution for 32 open scenarios’, ‘text language and category distribution’.

To provide the community with unified text-level quantitative descriptions, we will compare our dataset with the previous datasets from four aspects, i.e., text description, video scene, dataset size, and supported tasks. For text description attribute (i.e., Category, MLingual), our BOVText supports four types of text annotations(caption, title, scene, and other text) of video text with multi-language, which obviously has more extensive description ability than the existing dataset. For video scene attribute (i.e., Scenario), unlike the existing datasets, BOVText provides various open-world scenarios, including many new scenarios such as Sportscast(NBA, FIFA World Cup...), Life Vlog, Game, etc. We present the 31 open scenarios and an "Unknown" scenarios distribution on BOVText in three levels, i.e., video, video frames, and text instances, as shown in Table[2]. BOVText spans various video domains with these scenarios in the existing datasets (e.g., Driving for RoadText-1k[36], Vlog(suppermarket, shopping street, indoor), Travel(hotel, railway station) for LSVTD[6]) and more open domains that are not yet supported (e.g., Game(PUBG mobile, Honor of Kings...), Sport(NBA, world cup...), News). For the size of the dataset (i.e., Videos, Frames, Texts), BOVText is 25 times larger than the existing largest dataset (i.e., LSVTD[6]) with various scenarios(1,620,305 v.s 66,700 video frames). RoadText1k[36] contains 300k videos frames, but the supported scenario is too single for only supports driving video scenarios. For the supported tasks, the proposed BOVText supports all video text tasks: detection, recognition, video text tracking, end to end video text spotting. For comprehensive research, we not only focus the scale, location, recognition content, and tracking ID, but also additionally collect and annotate the category of caption, title, scene or other texts for
Table 2: The Data Distribution for 32 Open Scenarios. In green refers to these scenarios only supported by BOVText. "%" denotes the percentage of each scenario data for whole data.

<table>
<thead>
<tr>
<th>Scenarios</th>
<th>Video</th>
<th>Video Frames</th>
<th>Text Instances</th>
<th>Scenarios</th>
<th>Video</th>
<th>Video Frames</th>
<th>Text Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cartoon</td>
<td>61(3.2%)</td>
<td>60,395(3.6%)</td>
<td>123,191(1.2%)</td>
<td>Sport</td>
<td>91(4.8%)</td>
<td>72,090(4.3%)</td>
<td>266,996(4.6%)</td>
</tr>
<tr>
<td>Vlog</td>
<td>81(4.2%)</td>
<td>76,056(4.5%)</td>
<td>214,910(3.7%)</td>
<td>News Report</td>
<td>79(4.1%)</td>
<td>48,869(3.0%)</td>
<td>178,000(3.1%)</td>
</tr>
<tr>
<td>Driving</td>
<td>72(3.8%)</td>
<td>61,656(3.7%)</td>
<td>151,994(2.6%)</td>
<td>Celebrity</td>
<td>46(2.4%)</td>
<td>39,440(2.3%)</td>
<td>121,235(2.1%)</td>
</tr>
<tr>
<td>Advertising</td>
<td>33(1.7%)</td>
<td>29,129(1.6%)</td>
<td>91,090(1.6%)</td>
<td>Technology</td>
<td>59(3.1%)</td>
<td>52,072(3.1%)</td>
<td>140,172(2.4%)</td>
</tr>
<tr>
<td>Activity</td>
<td>31(1.6%)</td>
<td>23,585(1.4%)</td>
<td>67,879(1.2%)</td>
<td>Program</td>
<td>43(2.3%)</td>
<td>40,762(2.4%)</td>
<td>214,561(3.7%)</td>
</tr>
<tr>
<td>Comedy</td>
<td>86(4.5%)</td>
<td>79,404(4.7%)</td>
<td>317,865(5.5%)</td>
<td>Game</td>
<td>81(4.1%)</td>
<td>12,055(1.0%)</td>
<td>84,106(1.5%)</td>
</tr>
<tr>
<td>Interview</td>
<td>24(1.3%)</td>
<td>18,229(1.1%)</td>
<td>63,616(1.1%)</td>
<td>Livestreaming</td>
<td>59(3.1%)</td>
<td>60,494(3.6%)</td>
<td>211,569(3.6%)</td>
</tr>
<tr>
<td>Government</td>
<td>47(2.5%)</td>
<td>32,283(1.9%)</td>
<td>93,874(1.6%)</td>
<td>Speech</td>
<td>59(3.1%)</td>
<td>52,465(3.1%)</td>
<td>175,119(3.0%)</td>
</tr>
<tr>
<td>Travel</td>
<td>83(4.3%)</td>
<td>75,266(4.5%)</td>
<td>280,446(4.8%)</td>
<td>Movie</td>
<td>107(5.6%)</td>
<td>105,949(6.3%)</td>
<td>299,760(5.2%)</td>
</tr>
<tr>
<td>Campus</td>
<td>44(2.3%)</td>
<td>37,556(2.2%)</td>
<td>139,760(2.4%)</td>
<td>Photography</td>
<td>53(2.8%)</td>
<td>52,771(3.1%)</td>
<td>173,832(3.0%)</td>
</tr>
<tr>
<td>International</td>
<td>60(3.1%)</td>
<td>60,486(3.6%)</td>
<td>132,117(2.3%)</td>
<td>Education</td>
<td>73(3.8%)</td>
<td>60,315(3.6%)</td>
<td>360,774(4.2%)</td>
</tr>
<tr>
<td>Short Video</td>
<td>84(4.4%)</td>
<td>79,148(4.7%)</td>
<td>326,938(5.6%)</td>
<td>Dance</td>
<td>37(1.9%)</td>
<td>22,254(1.3%)</td>
<td>71,740(1.2%)</td>
</tr>
<tr>
<td>Makeup</td>
<td>60(3.1%)</td>
<td>52,449(3.1%)</td>
<td>111,814(1.9%)</td>
<td>Fishery</td>
<td>86(4.5%)</td>
<td>79,750(4.7%)</td>
<td>230,085(4.0%)</td>
</tr>
<tr>
<td>Talent</td>
<td>79(4.1%)</td>
<td>66,038(3.9%)</td>
<td>339,382(5.9%)</td>
<td>Fashion</td>
<td>53(2.8%)</td>
<td>42,868(2.6%)</td>
<td>98,942(1.7%)</td>
</tr>
<tr>
<td>Beauty Industry</td>
<td>37(1.9%)</td>
<td>35,852(2.1%)</td>
<td>132,025(2.3%)</td>
<td>Introduction</td>
<td>73(3.8%)</td>
<td>42,086(2.4%)</td>
<td>236,721(4.1%)</td>
</tr>
<tr>
<td>Eating</td>
<td>52(2.7%)</td>
<td>65,609(3.9%)</td>
<td>191,035(3.3%)</td>
<td>Unknown</td>
<td>48(2.5%)</td>
<td>29,212(1.7%)</td>
<td>150,721(2.6%)</td>
</tr>
</tbody>
</table>

Each text instance. As shown in Figure 2 (b), in a video, different types of text instances may exist simultaneously, and they are helpful to understand videos synergistically. Caption text can directly show the dialogue between people in video scenes and represent the time or topic of the video scenes, scene text can unambiguously define the object and can identify important localization and road paths in video scenes. Therefore, the text category annotation is favoring downstream tasks (e.g., video text translation, video understanding, and video retrieval), more details in the appendix.

3.3 Supported Tasks and Metrics

The proposed BOVText supports four tasks: (1) Text Detection. (2) Text Recognition. (3) Video Text Tracking. (4) End to End Text Spotting in Videos. Following ICDAR2015 [71], the evaluation protocols [51] are used for text detection and recognition task. For video text tracking and spotting task, the existing video text datasets such as ICDAR2015 (video) [18] and RoadText-1k [36] all adopted the MOT metrics (i.e., Multiple Object Tracking Accuracy (MOTA) and Multiple Object Tracking Precision (MOTP)). However, there are two sets of measures for Multiple Object Tracking: the MOT metrics (MOTA, MOTP) [2] and ID metrics (IDF1) [22, 39]. The CVPR19 MOTChallenge evaluation framework [3] presents that different measures serve different purposes. Event-based measures like MOT help pinpoint the source of some errors and are thereby informative for the designer of certain system components. Identity-based measure(IDF1) is more favorable for evaluating how well computed identities conform to true identities. Except for using MOTA, MOTP, Identity-based measures(IDF1), as a new metric is adopted firstly for video text spotting task. More detailed information for metric can be obtained in the appendix.

3.4 Our Method: TransVTSpotter

Two ingredients are essential for direct text spotting for TransVTSpotter: (1) A set prediction loss that forces unique matching between predicted and ground truth multi-orient boxes. (2) An architecture that predicts a set of objects and associates the same objects during different frames.

Multi-orient Box Matching. Compare to DETR [5], the difference is that we propose an angle prediction and corresponding loss while only horizontal boxes prediction for DETR. Let us denote the ground truth set of objects by \( y_i \) and \( \hat{y}_i \) the set of \( N \) predictions. \( y \) is as a set of size \( N \) padded with \( \emptyset \) (no object). To find a bipartite matching between these two sets we search for a permutation of \( N \) elements \( \sigma \in \mathcal{S}_N \) with the lowest cost:

\[
\sigma = \arg \min_{\sigma \in \mathcal{S}_N} \sum_{i=1}^{N} \mathcal{L}_{\text{match}}(y_i, \hat{y}_{\sigma(i)}),
\]

where \( \mathcal{L}_{\text{match}}(y_i, \hat{y}_{\sigma(i)}) \) is a pair-wise matching cost between ground truth \( y_i \) and a prediction with index \( \sigma(i) \). The matching cost takes into account the class prediction, boxes prediction and the boxes rotated angle prediction. Each element \( i \) of the ground truth set can be seen as \( y_i = (c_i, b_i, \alpha_i) \) where \( c_i \) is the target class label, \( b_i \in [0, 1]^d \) is a vector that defines ground truth box center coordinates and its height and width relative to the image size, and \( \alpha_i \) is rotation angle between the longest edge of ground truth multi-orient box and horizontal line (x-axis). For the prediction with index \( \sigma(i) \) we define
probability of class $c_i$ as $\tilde{p}_{\sigma(i)}(c_i)$, the predicted box as $\hat{b}_{\sigma(i)}$, and the predicted angle as $\hat{a}_{\sigma(i)}$. Thus we define $L_{\text{match}}(y_i, \tilde{y}_{\sigma(i)})$ as $\mathbf{1}_{\{c_i \neq \emptyset\}} \tilde{p}_{\sigma(i)}(c_i) + \mathbf{1}_{\{c_i \neq \emptyset\}} L_{\text{box}}(b_i, \hat{b}_{\sigma(i)}) + \mathbf{1}_{\{c_i \neq \emptyset\}} L_{\text{angle}}(a_i, \hat{a}_{\sigma(i)})$.

Finally, we could compute the loss function with all pairs matched:

$$L_{\text{Hungarian}}(y, \hat{y}) = \sum_{i=1}^{N} \{ - \log \tilde{p}_{\sigma(i)}(c_i) + \mathbf{1}_{\{c_i \neq \emptyset\}} L_{\text{box}}(b_i, \hat{b}_{\sigma(i)}) + \mathbf{1}_{\{c_i \neq \emptyset\}} L_{\text{angle}}(a_i, \hat{a}_{\sigma(i)}) \} ,$$

where $L_{\text{box}}(\cdot)$ a linear combination of the $\ell_1$ loss and the generalized IoU loss [58]. And $L_{\text{angle}}(\cdot)$ refers to a cosine embedding loss with $1 - \cos(\hat{a}_{\sigma(i)} - a_i)$.

**TransVTSpotter architecture.** The overall TransVTSpotter architecture is surprisingly simple and depicted in Figure. It contains four main components: 1) A transformer backbone [53] extracts feature representation of multiple images; 2) A transformer encoder models the relations of pixel-level features; 3) Two transformer decoders (shared weight) decode the instance-level features; 4) Attention-based recognizer [28] recognizes each text instance.

**Text Tracking with Transformer.** Text Tracking with Transformer includes three components: backbone, transformer encoder and transformer decoder, as shown in Figure. Backbone. Starting from the two consecutive frames $x_t \in \mathbb{R}^{3 \times H_0 \times W_0}$ and $x_{t+1} \in \mathbb{R}^{3 \times H_0 \times W_0}$, a transformer backbone [53] generates a lower-resolution activation map for the two frames ($f_t \in \mathbb{R}^{C \times H \times W}$ and $f_{t+1} \in \mathbb{R}^{C \times H \times W}$), then a new feature sequence $f_t^* \in \mathbb{R}^{C \times H \times W}$ can be obtained by simple concatenating $f_t$ and $f_{t+1}$. The extracted features $f_t$ of the current frame are temporarily saved and then re-used for the next frame. Transformer Encoder. We adopted deformable transformer encoder [72] to model the similarities among all the pixel level features for the extracted features $f_t$, Transformer Decoder. Two parallel decoders [47] are employed. The object decoder takes learned object query [5] as input and predicts detection rotated boxes. The tracking decoder takes the object feature from previous frames as input and predicts the corresponding tracking rotated boxes. Finally, with detection rotated boxes and tracking rotated boxes, TransTT obtains the final tracking result by box IoU matching and the Kuhn-Munkres(KM) algorithm [19].

**Recognizer.** Following MASTER [28] is utilized to predict output sequence with 2D-attention.

### 4 Experimental

In this section, we mainly conduct experiments on our BOVText. More experiments, such as the performance of TransVTSpotter in other datasets, would be provided in the appendix.

#### 4.1 Implementation Details

Except for the TransVTSpotter, we also adopt CRNN [42], RARE [43] as the recognition baseline and PSENet [54], EAST [70], DB [23] as the detection baseline to evaluate our BOVText. Detection: we train detectors with pre-trained model on COCOText [49]. Recognition: the network is pre-trained on the chinese oc4 [8] and MJSynth [15], then fine-tuned on our BOVText. All of our experiments are conducted on 8 V100 GPUs. AdamW [27] as the optimizer and the batch size is set to be 16. The

---

https://github.com/YCG09/chinese_ocr
Table 3: Attribute Experiments for Scenarios. In green are the gaps compare to training with RoadText-1k [36] and LSVTD [6].

<table>
<thead>
<tr>
<th>Method</th>
<th>Training Set</th>
<th>Detection (F-score/%) on BOVText</th>
<th>Tracking (IDF1%) on BOVText</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Cartoon</td>
<td>Travel</td>
</tr>
<tr>
<td>TransVTSpotter</td>
<td>LSVTD</td>
<td>54.3</td>
<td>42.7</td>
</tr>
<tr>
<td></td>
<td>LSVTD&amp;RoadText</td>
<td>56.2</td>
<td>45.1</td>
</tr>
<tr>
<td></td>
<td>BOVText</td>
<td>93.9</td>
<td>65.4</td>
</tr>
</tbody>
</table>

Figure 4: Attribute Experiments of BOVText. (a) Tracking performance (i.e., \(IDF_1\)) with TransVTSpotter in different scenarios, ‘LSVT&D’ and ‘BOVText’ denotes training on LSVTD and BOVText, respectively. (b) End to end video text spotting performance (i.e., \(IDF_1\)) with TransVTSpotter in different scenarios. (c) Recognition accuracy of different models in different languages. (d) Detection of different model in caption or scene text. ‘CH’, ‘AN’, ‘EN’ and ‘ALL’ refer to ‘Chinese’, ‘Alphanumeric’, ‘English’ and ‘All Characters’, respectively.

initial learning rate is 2e-4 for the transformer and 2e-5 for the backbone. The weight decay is 1e-4. All transformer weights are initialized with Xavier-init [11]. The data augmentation includes random horizontal, scale augmentation, resizing the input images whose shorter side is by 480-800 pixels while the longer side is by at most 1333 pixels. The model is first pre-trained on COCOText [49] and then fine-tuned on other video text training sets. For each iteration, two adjacent frames are randomly selected from one video from training set to train our model.

4.2 Attribute Experiments Analysis for BOVText

New Scenarios, New Challenge for Video Text Tasks. Figure 4(a) and Table 3 gives the tracking performance \(IDF_1\) of TransVTSpotter in different scenarios of BOVText. Two new insights can be present from the figure and table: 1) The existing benchmark datasets cannot effectively test the effectiveness of advancing algorithm on some novel scenarios (e.g., Game, Cartoon) for first proposed in BOVText. LSVTD [6] and RoadText-1k [36], as the two largest data sets on the existing video text datasets are used to compare with our BOVText, TransVTSpotter achieves a tracking performance \(IDF_1\) of 61.2% on Game scenario with BOVText training set, 26 percent point improvement than training with LSVTD [6] and RoadText-1k [36]. We argue that the main cause is that there existing a mass of texts in Game scenario, but LSVTD and RoadText almost no such dense text scenario, which is a new challenge for algorithms. A visualization example in YouTube is used to support the idea. Besides, training with only RoadText-1k obtains a low performance no matter which scenarios. There are main two causes for this. Firstly, the location annotation of RoadText-1k is an upright bounding box (two points), but the counterpart of BOVText is multi-orient boxes (four points). Secondly, the data domain is entirely different for the two datasets. Compare with various scenarios (e.g., Game, Sports) and text types (e.g., long caption text, big text), the scenario of RoadText-1k only contains small and low-resolution road signs, plate number on driving scenarios. 2) Huge performance gap existing during different scenarios. As shown in Figure 4(a), the model achieves the best performance with a \(IDF_1\) of 90.2% in Cartoon videos, since the conspicuous text instances and simple background are designed in cartoon videos. By comparison, several scene categories obtain extremely dissatisfied performance due to complex background, various text appearance, and unstable camera movements, such as Campus of 40.2% and Travel of 24.1%.
Table 4: Detection and Recognition Performance on BOVText.

<table>
<thead>
<tr>
<th>Method</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
<th>Method</th>
<th>Prediction</th>
<th>Recognition</th>
</tr>
</thead>
<tbody>
<tr>
<td>EAST [70]</td>
<td>52.2</td>
<td>38.1</td>
<td>44.1</td>
<td>CRNN [42]</td>
<td>26.0</td>
<td>31.2</td>
</tr>
<tr>
<td>PSENet [54]</td>
<td>74.3</td>
<td>73.2</td>
<td>73.5</td>
<td>RARE [43]</td>
<td>25.2</td>
<td>34.2</td>
</tr>
<tr>
<td>DB [23]</td>
<td>77.2</td>
<td>74.5</td>
<td>75.3</td>
<td>GRCCNN [50]</td>
<td>23.1</td>
<td>39.8</td>
</tr>
<tr>
<td>TransVTSpotter(ours)</td>
<td>80.2</td>
<td>76.4</td>
<td>78.5</td>
<td></td>
<td>-</td>
<td>26.2</td>
</tr>
</tbody>
</table>

Table 5: Text Tracking and End to End Video Text Spotting Performance on BOVText.

<table>
<thead>
<tr>
<th>Method</th>
<th>Text Tracking on BOVText</th>
<th>End to End Text Spotting on BOVText</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detection</td>
<td>Recognition</td>
<td>Detection</td>
</tr>
<tr>
<td>EAST [70]</td>
<td>CRNN [42]</td>
<td>23.5</td>
</tr>
<tr>
<td>PSENet [54]</td>
<td>CRNN [42]</td>
<td>34.7</td>
</tr>
<tr>
<td>DB [23]</td>
<td>RARE [43]</td>
<td>33.7</td>
</tr>
<tr>
<td>TransVTSpotter(ours)</td>
<td>RARE [43]</td>
<td>65.0</td>
</tr>
</tbody>
</table>

Bilingual Recognition, New Challenge. As shown in Figure 4(c), the text recognition results for different languages are provided. In summary, the alphanumerical recognition result (about 47%) is better than the Chinese recognition result (about 35%), regardless of the models. The final results (about 40%) for all characters are satisfactory, can not meet the requirement of the application. Unlike English, Chinese contains thousands of characters, thus we conduct several experiments for each language.

Long Caption Text, New Challenge. As shown in Figure 4(d), for DB [23], PSENet [54] and our TransVTSpotter, the performance of caption text is better than the counterpart of scene text (around 80% vs. 60%) due to the more clear and bigger caption text. But for EAST [70], long caption text show a low performance with 35% F-score. The prime reason is that caption texts are all long text (average width-height ratio: 6.8 for caption vs. 2.3 for scene text on BOVText), a different case for EAST [70], as shown in YouTubeDemo. However, the existing video text datasets hardly contain long caption texts, our BOVText can fill out the gap for a more comprehensive evaluation of text types.

4.3 Text Detection, Recognition, Tracking and Spotting on BOVText

Video Text Detection and Recognition. As shown in Table 4, image-based text detection on BOVText is not unsatisfactory, with lower results than these methods report on existing text datasets. For example, EAST obtains an F-score of 44.1% compared to the F-score of 80.7% on icdar2015 [71], but our TransVTSpotter obtain an F-score of 78.5% on BOVText, at least 3% improvement compared to the image-based detectors (i.e., DB, PSENet and EAST). For text recognition, CRNN [42] based on CTC loss, RARE [43] with attention mechanism and GRCCNN [50] as the base text recognizers to test our BOVText. The text annotation in our BOVText covers two languages (i.e., English and Chinese), thus we conduct several experiments for each language. The recognition model only yields about 40% accuracy on our dataset, but the same model reports at least 90% on most benchmark datasets [18] for text recognition. The main reasons have two points: (1) The proposed BOVText is bilingual, and the category number of Chinese characters in real-world is much larger than those of Latin languages. (2) The video texts are quite blurred, out-of-focus, and the distribution of characters is relatively smaller than the static image counterparts, which presents more challenges.

Video Text Tracking. As shown in Table 5, IDP1 (58.2%) of our TransVTSpotter achieves the best performance, at least 20% improvement than other methods. Besides, without NMS and other post-processing, TransVTSpotter presents 13 fps no matter which dataset. More details and analysis concerning TransVTSpotter can be obtained in the appendix. And EAST shows the worst performance with a IDF1 of 23.2%. The IDF1 of EAST [70] is lower with 6.7% gap than that of PSENet [54]. The main reason is that MOVText contains a mass of long text instances, but regression-based EAST can not deal with the long text cases well. The performance of DB is similar to that of PSENet for both all are the segmentation-based methods.

End to End Text Spotting in Video. Detection and text tracking tasks are paving the way for the recognition task. Table 5 shows the performance of text spotting on BOVText. Similar to the text tracking performance, our TransVTSpotter achieves the state-of-the-art performance with at least...
6.2% $ID_{F1}$ improvement compared to the other methods. Besides, the MOTP of TransVTSpotter achieves 0.781%, two percent points improvement than the counterpart of using DB and RARE. The great performance for 6.2% $ID_{F1}$ and 0.781% MOTP present satisfactory tracking and recognition trajectory and detection results, respectively. The corresponding performance using EAST [70] as the detector in video text spotting is still not satisfied with around 5% $ID_{F1}$ and −0.8 MOTA. Without TransVTSpotter, the combination of DB [23] and RARE [43] achieves the best performance with a 17.3% $ID_{F1}$, but there is at least 6.2% gap compare to our method.

4.4 Potential Negative Societal Impacts and Solution

We argue that there mainly exits slight potential negative societal impacts for personal privacy. Although much personal information, e.g., names, identifying information, human faces, have been blurred to protect privacy, there still might exist a little risk. **How to blur human faces.** We blur the human faces in BOVText with four steps. Firstly, human faces in each frame would be detected by face_recognition[9] - a powerful, simple, and easy-to-use face recognition open source project with complete development documents and application cases. Secondly, with the location box from the previous step, we extract face ROI from the original image. Thirdly, we blur the face ROI with Gaussian Blur operation in OpenCV[10]. Finally, we store the blurred face in the original image and recover to video.

4.5 Link to Other Video-and-Language Applications

In this section, we show that the practicability of the proposed BOVText, not a toy benchmark, which can promote other video-and-text application research. Text spotting in static images has numerous application scenarios: (1) Automatic data entry. SF-Express[11] utilizes OCR techniques to accelerate the data entry process. NEBO[12] performs instant transcription as the user writes down notes. (2) Autonomous vehicle [31, 30]. Text-embedded panels carry important information, e.g., geo-location, current traffic condition, navigation, and etc. (3) Text-based reading comprehension. TextCaps [44] and text-based VQA [45, 3] show the new vision-and-language tasks, which need to recognize text, relate it to its visual context, semantic, and visual reasoning between multiple text tokens and visual entities, such as objects. Similarly, there are many application demands for video text understanding across various industries and in our daily lives. We list the most outstanding ones that significantly impact, improving our productivity and life quality. **Firstly**, automatically describing video with natural language [60, 56] can bridge video and language. **Secondly**, video text automatic translation[13] can be extremely helpful as people travel, and help video-sharing websites [14] to cut down language barriers. **Finally**, text-based video retrieval [20, 24] is irreplaceable business for many companies, such as Google and YouTube. More details and analyses for application scenarios concerning BOVText in the appendix.

5 Conclusion and Future Work

In this paper, we establish a large-scale, bilingual open-world benchmark dataset for video text tracking and spotting, termed BOVText, with four description information, i.e., , bounding box, tracking ID, recognition content, and text category label. Compare with the existing benchmarks, the proposed BOVText mainly contains four advantages: large-scale training set (i.e., 1,850+video), 32 open real scenarios (Sportscast, Life Vlog, Game), bilingual annotation, and abundant text types annotation(Caption, title, and scene text). Besides, we first propose an end-to-end video text spotting framework with Transformer, termed TransVTSpotter, which presents a simple, but efficient attention-based query-key pipeline. On ICDAR2015(video), TransVTSpotter achieves the state-of-the-art performance. In general, we hope the proposed BOVText and TransVTSpotter would provide a standard benchmark to facilitate the advance of video-and-text research.

9https://github.com/ageitgey/face_recognition
10https://www.tutorialspoint.com/opencv/opencv_gaussian_blur.htm
11https://www.sf-express.com/cn/sc/
12https://www.myscript.com/nebo/
13https://translate.google.com/intl/en/about/
14https://www.youtube.com/
References


