TextFuseNet: Scene Text Detection with Richer Fused Features

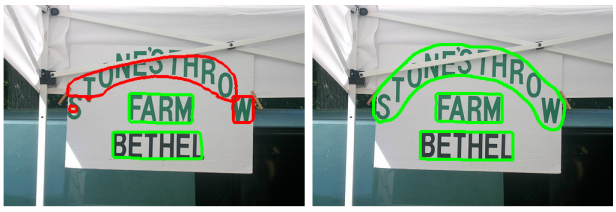
Jian Ye[1] , **Zhe Chen**[2] , **Juhua Liu**[3*] and **Bo Du**[1*]
[1]School of Computer Science, Institute of Artificial Intelligence, and National Engineering Research Center for Multimedia Software, Wuhan University, China
[2]UBTECH Sydney AI Centre, School of Computer Science, Faculty of Engineering, The University of Sydney, Australia
[3]School of Printing and Packaging, and Institute of Artificial Intelligence, Wuhan University, China
{leaf-yej, liujuhua, dubo}@whu.edu.cn, zhe.chen1@sydney.edu.au

Abstract

Arbitrary shape text detection in natural scenes is an extremely challenging task. Unlike existing text detection approaches that only perceive texts based on limited feature representations, we propose a novel framework, namely TextFuseNet, to exploit the use of richer features fused for text detection. More specifically, we propose to perceive texts from three levels of feature representations, *i.e.*, character-, word- and global-level, and then introduce a novel text representation fusion technique to help achieve robust arbitrary text detection. The multi-level feature representation can adequately describe texts by dissecting them into individual characters while still maintaining their general semantics. TextFuseNet then collects and merges the texts' features from different levels using a multi-path fusion architecture which can effectively align and fuse different representations. In practice, our proposed TextFuseNet can learn a more adequate description of arbitrary shapes texts, suppressing false positives and producing more accurate detection results. Our proposed framework can also be trained with weak supervision for those datasets that lack character-level annotations. Experiments on several datasets show that the proposed TextFuseNet achieves state-of-the-art performance. Specifically, we achieve an F-measure of 94.3% on ICDAR2013, 92.1% on ICDAR2015, 87.1% on Total-Text and 86.6% on CTW-1500, respectively.

1 Introduction

Scene text detection is attracting increasing attention in the computer vision community. Many progress has been made with the rapid development of deep learning[Wang *et al.*, 2019d][Wang *et al.*, 2017][Qiao *et al.*, 2019][Gao *et al.*, 2019]. However, this task remains challenging, since the texts commonly have diversified shapes and text detectors can be easily affected by the issues like complex backgrounds, irregular shapes, and texture interference.



(a) (b)

Figure 1: Illustrations of the results from commonly used instance-segmentation-based methods (a) and our proposed TextFuseNet (b). Green polygons represent true positives, while red polygons represent false positives.

Existing methods mainly have two types: character-based methods and word-based methods. Character-based methods regard the text as a combination of multiple characters. They first extract characters with well-designed character detectors and then group them into words. However, character-based methods are generally time-consuming due to the significantly large number of character candidates generated for text detection. Instead of character-based methods, word-based methods have been proposed to directly detect words based on generic object detection pipelines. Although they are much simpler and more efficient, these methods usually fail to effectively detect the texts with arbitrary shapes. To tackle this issue, some word-based methods further apply instance segmentation to conduct text detection. In these methods, foreground segmentation masks are estimated to help determine various text shapes. Despite promising results, existing instance-segmentation-based methods still have two major limitations. Firstly, these methods only detect texts based on a single region of interest (RoI) without considering global contexts, thus they tend to produce inaccurate detection results based on limited visual information. Secondly, prevailing methods do not model different levels of word semantics, running the risk of producing false positives for text detection. Figure 1 shows an example of these methods.

In this paper, we propose a novel scene text detection framework, namely TextFuseNet, to effectively detect texts with arbitrary shapes by utilizing richer fused features. In general, we follow Mask R-CNN [He *et al.*, 2017] and Mask TextSpotter [Lyu *et al.*, 2018] and formulate the text detection task as an instance segmentation task. Unlike these methods, we recast the original pipeline of Mask R-CNN to enable the

*Corresponding author

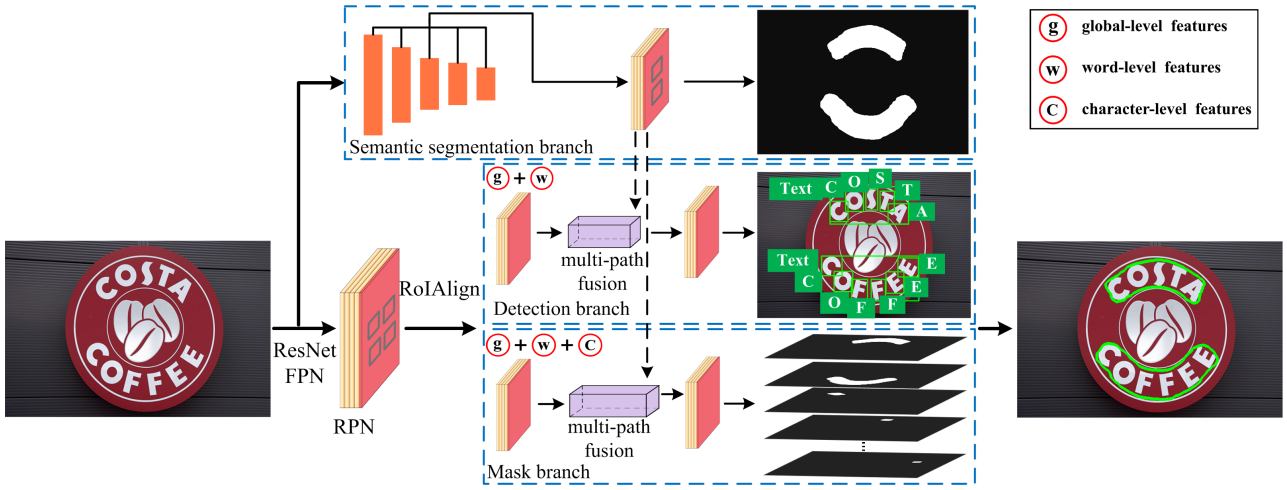


Figure 2: The overall pipeline of the proposed framework. We extract and utilize three levels of feature representations, *i.e.*, character-, word- and global-level features for texts. We also propose the multi-path fusion architecture to obtain richer fused features for text detection.

analysis and fusion of three levels of feature representations, *i.e.*, character-, word-, and global-level features, for text detection. In particular, we first introduce an additional semantic segmentation branch to the detection pipeline to help perceive and extract global-level representations. The global semantic features can later be used to guide the detection and mask branches of the detection pipeline. Next, we attempt to extract character- and word-level features within detection and mask branches in the Mask R-CNN pipeline. Different from the original Mask R-CNN, in the detection and mask branches, we detect and segment not only word instances but also character instances, delivering both character- and word-level representations. After the perception of three-level representations, we then introduce the multi-path feature fusion architecture, which fuses the character-, word- and global-level features through a multi-path fusion network, to facilitate the TextFuseNet to learn a more discriminative representation and produce more accurate text detection results. In practice, considering that some existing datasets lack annotations for characters, we further develop a weakly supervised learning scheme to generate character-level annotations by learning from word-level annotated datasets. Overall, the architecture of TextFuseNet is shown in Figure 2.

The contributions of this work are three-fold: (1) We propose a novel framework, namely TextFuseNet, which extracts character-, word- and global-level features and introduce a multi-path fusion architecture to fuse them for accurate text detection; (2) Based on the proposed framework, we introduce a weakly supervised learning scheme, which exploits word-level annotations to guide searching for character training samples, to achieve effective learning without annotations for character instances; (3) Our proposed framework achieves state-of-the-art performance on several famous benchmarks which contain texts of arbitrary shapes.

2 Related Work

As mentioned above, existing methods can be roughly classified into two main categories, *i.e.*, character-based methods and word-based methods.

Character-based methods usually first apply some complicated character detectors, such as SWT, MSER and FASText, to extract character candidates. These character candidates are filtered by a character/non-character classifier to remove false candidates. Finally, the remained characters are grouped into words according to either prior knowledge or some clustering/grouping models. However, most character-based methods require elaborate design and involve multiple stages of processing, which are very complicated and lead to error accumulation. Therefore, the performance of character-based methods is always time-consuming and suboptimal.

Word-based methods detect words directly, which are mainly inspired by general object detection methods. [Tian *et al.*, 2016] proposed a Connectionist Text Proposal Network (CTPN) which consists of CNN and RNN to detect whole text lines by linking a series of small text boxes. Inspired by SSD, [Liao *et al.*, 2018a] proposed TextBoxes and its extension TextBoxes++ by adding several text-box layers. [Shi *et al.*, 2017] proposed SegLink by employing Fully Convolutional Networks (FCN) to detect text segments and their link relationships. Text segments are linked as the final detection result according to their relationship. However, these methods are only suitable for horizontal or multi-oriented text.

In order to tackle the challenge of texts with arbitrary shapes, many **instance-segmentation-based methods** have been proposed to detect text with arbitrary shapes. [Deng *et al.*, 2018] conducts text/non-text prediction and link prediction through CNN, and connects positive sample pixels with positive links, which directly obtains text boxes without regression. [Xie *et al.*, 2019] proposed a Supervised Pyramid Context Network (SPCNet) to locate text regions based on Mask R-CNN. [Wang *et al.*, 2019a] proposed a Progressive Scale Expansion Network (PSENet) to detect text with arbitrary shapes. [Tian *et al.*, 2019] mapped pixels onto an embedding space and introduced a shape-aware loss to make training adaptively accommodate various aspect ratios of text instances. Compared with previous works, we analyze and

fuse more different levels of features to obtain richer fused features, which effectively improve the performance of text detection.

3 Methodology

In this section, we describe how the multi-level feature representations are extracted by semantic segmentation, detection, and mask branches, and how we fuse them using the multi-path fusion architecture. Meanwhile, we also explore the strategy of weakly supervised learning for generating character-level annotations.

3.1 Framework

Figure 2 depicts the overall architecture of TextFuseNet. In the TextFuseNet, we first extract multi-level feature representations and then perform multi-path fusion to conduct text detection. This framework is mainly implemented by five components: a feature pyramid network (FPN) as backbone for extracting multi-scale feature maps, a region proposal network (RPN) for generating text proposals, a semantic segmentation branch for exploiting global semantics, a detection branch for detecting words and characters, and a mask branch for instance segmentation of words and characters.

In TextFuseNet, we first follow Mask R-CNN and Mask TextSpotter and employ a ResNet as the backbone of FPN. Also, we use RPN to generate text proposals for the subsequent detection and mask branches. Then, to extract multi-level feature representations, we mainly propose to apply the following implementations. First, we introduce a new semantic segmentation branch to conduct semantic segmentation for the input image and help obtain global-level features. Then, in the detection branch that refines text proposals via predicting their categories and adopting bounding box regression, we extract and fuse word- and global-level features to detect both words and characters. This is different from existing methods only focus on detecting a single word or character for each proposal. For the mask branch that performs instance segmentation for objects detected from the detection branch, we extract and fuse all the character-, word-, and global-level features to fulfill the instance segmentation, as well as the final text detection task. The detailed network configurations to extract multi-level feature representations are presented in Section 3.2. After the multi-features are extracted, we then propose a multi-path fusion architecture to fuse the different features for detecting texts with arbitrary shapes. The multi-path fusion architecture can effectively align and merge the multi-level features to deliver robust text detection. The details of the implementation of the multi-path fusion architecture are described in Section 3.3.

3.2 Multi-level Feature Representation

In general, character- and word-level features can be easily obtained within the detection and mask branches of the detector. We can achieve this by detecting both words and characters appeared within the proposals. RoIAlign is applied here to extract different features and perform detection for both words and characters.

However, we need a novel network in the feature extraction stage to help obtain the global-level features. Therefore, we propose to further employ a semantic segmentation branch in the detector to extract global-level features. As presented in Figure 2, the semantic segmentation branch is constructed based on the output of FPN. We fuse features from all the levels into a unified representation and perform segmentation on this unified representation, thus obtaining globally segmented results for text detection. In practice, we apply a 1×1 convolution to align channel numbers of features from different levels and resize the feature maps into the same size for later unification.

3.3 Multi-path Fusion Architecture

After we have obtained multi-level features, we adopt multi-path fusion in both detection and mask branches. In the detection branch, based on the text proposals obtained from RPN, we extract global- and word-level features for text detection in different paths. We then fuse both types of features to deliver text detection in the form of words and characters. Note that we cannot extract and fuse character-level features in detection branch because the characters are not yet recognized before performing the detection. In practice, given a generated text proposal, we use RoIAlign to extract the global- and word-level features within the size of 7×7 from the output features of FPN. We fuse these features via by element-wise summation, and feed them into a 3×3 convolution layer and a 1×1 layer. The final fused features are used for classification and bounding box regression.

In the mask branch, for each word-level instance, we can then fuse the corresponding character-, word-, and global-level features within the multi-path fusion architecture for instance segmentation. Figure 3 shows a detailed illustration of the multi-path fusion architecture. In the proposed architecture, we extract multi-level features from different paths and fuse them to obtain richer features to help learn a more discriminative representation.

Formally, given an input word denoted by r_i, we first identify the character results C_i which belong to this word proposal based on its ratio of the intersection with a character over the character's area, meaning that the ratio is 1 if the word box fully covers the character, and 0 if not. We use c_j to denote the characters. Then the set of characters C_i belong to the word r_i can be collected based on:

$$C_i = \{c_i \mid \frac{b_i \cap b_j}{b_j} > T\} \tag{1}$$

where b_i and b_j are the bounding boxes of word r_i and character instance c_j, respectively, and T is the threshold. In our implementation, we set $T = 0.8$.

Since the number of characters is not fixed and may range from zero to hundreds, for a given detected word r_i, we fuse the features of characters in the set C_i into a unified representation. In particular, we first use RoIAlign to extract the corresponding features with the size of 14×14 for each character in C_i and then fuse these feature maps via element-wise summation. Through a 3×3 convolution layer and a 1×1 convolution layer, we can obtain the final character-level features.

By further applying RoIAlign to extract features of words and corresponding global semantic features, we fuse all these three levels features via element-wise summation, and feed

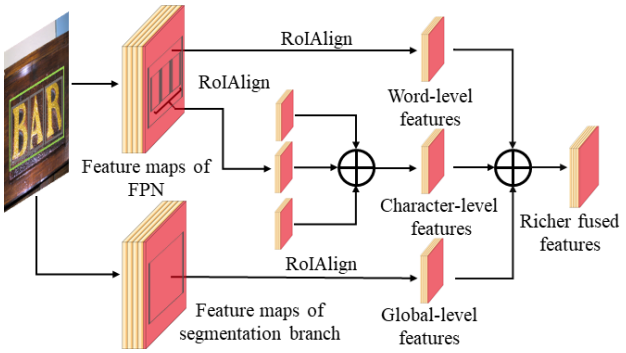


Figure 3: Illustration of the multi-path fusion architecture in the mask branch. For a word proposal, we fuse character-, word- and global-level features in different paths to obtain richer fused features.

them into a 3×3 convolution layer and a 1×1 layer to obtain richer features. The final fused features are used for instance segmentation. Note that the 3×3 convolution layers and 1×1 convolution layer following element-wise summation are used for further bridging the semantic gap among different features.

Overall Objective. Lastly, we formulate the overall objective of the proposed TextFuseNet for tackling text detection problem:

$$L = L_{rpn} + L_{seg} + L_{det} + L_{mask} \tag{2}$$

where L_{rpn}, L_{seg}, L_{det} and L_{mask} are the loss function of RPN, semantic segmentation branch, detection branch and mask branch respectively.

3.4 Weakly Supervised Learning

Since TextFuseNet is formulated to detect both words and characters, character-level annotations are needed to achieve effective training. However, as mentioned previously, some existing datasets do not provide character-level annotations to train the TextFuseNet. Instead of annotating characters which is a time-consuming and labor costly task, we are inspired by the idea of weakly supervised learning and propose a weak supervision-based learning scheme to help train the TextFuseNet. In the proposed scheme, we search for character-level training examples by learning from the weakly supervised data with a pre-trained model. The pre-trained model is trained based on our proposed framework on a fully annotated datasets which provide both character- and word-level annotations. Then, for a dataset *A* that only has word-level annotations, the goal of our developed weak supervised learning is to search for character training samples in *A* through the pre-trained model *M*.

More specifically, we first apply the pre-trained model *M* on the word-level annotated dataset *A*. For each image in dataset *A*, we can obtain a set of character candidate samples:

$$R = \{r_0(c_0, s_0, b_0, m_0), r_1(c_1, s_1, b_1, m_1), \cdots, r_i(c_i, s_i, b_i, m_i), \cdots\} \tag{3}$$

where c_i, s_i, b_i, and m_i represent the predicted category, confidence score, bounding box, and mask of the *i*-th character candidate sample r_i, respectively. Then we filter false positive samples in *R* based on the confidence score threshold and the weakly supervised word-level annotations and obtain positive character samples:

$$P = \{(c_i, s_i) \mid c_i \in C \text{ and } s_i > S \text{ and } \frac{(m_i \cap g_i)}{m_i} > T\} \tag{4}$$

where *C* denotes all character categories to be detected, *S* represents the confidence score threshold used to identify positive character samples, $\frac{(m_i \cap g_i)}{m_i}$ denotes the intersection overlap of candidate character sample r_i with its word-level ground truth g_j, and *T* is the threshold to determines whether the candidate character sample is inside the words or not. Due to the constraints provided by word-level annotations, the confidence score threshold *S* can be set to a relatively lower, which is also beneficial for keeping the diversity of character samples. In our implementation, *S* and *T* is set to 0.1 and 0.8 respectively. Finally, the identified positive character samples can be used as character-level annotations and be combined with the word-level annotations to train a more robust and accurate text detection model.

4 Experiments

In this section, we evaluate the performance of TextFuseNet on four challenging public benchmark datasets: ICDAR 2013, ICDAR 2015, Total-Text and CTW-1500, and compare with previous state-of-the-art methods.

4.1 Datasets

SynthText is a synthetically generated dataset and usually used for pre-training the text detection models. This dataset consists of 800,000 images with 8 million synthetic word instances with both word- and character-level annotations in the form of rotated rectangles.

ICDAR2013 is a typical horizontal text dataset and is proposed in Challenge 2 of the ICDAR 2013 Robust Reading Competition. It contains 229 training images and 233 test images. ICDAR 2013 also provides both character- and word-level annotations.

ICDAR2015 is a multi-orient text dataset and is proposed in Challenge 4 of the ICDAR 2015 Robust Reading Competition. It focuses on incidental scene text and contains 1000 training images and 500 test images. This dataset only provides word-level annotations labeled with quadrangles.

Total-Text is a comprehensive arbitrary shape text dataset for scene text reading. Total-Text contains 1255 training images and 300 test images. All images are annotated with polygons in word-level.

CTW-1500 also focuses on arbitrary shape text reading and contains 1000 training images and 500 test images. Different from Total-Text, the annotations in CTW-1500 are labeled with polygons in the text-line-level.

4.2 Implementation Details

We implemented our framework based on the Maskrcnn-benchmark, and all experiments are conducted on a high-performance server with NVidia Tesla V100 (16G) GPUs.

Figure 4: Example results of TextFuseNet on different datasets. Sample images from (a) to (d) are selected from ICDAR 2013, ICDAR 2015, Total-Text and CWT-1500, respectively.

The model is trained with 4 GPUs and evaluated with 1 GPU.

Training. The whole training process was divided into three stages: pre-training on SynthText, searching for character training samples under weak supervision, and fine-tuning on the real-world data. Since SynthText provides both word- and character-level annotations, we can obtain a pre-trained model with full supervision. After pre-training, for weakly supervised learning, we apply the pre-trained model on ICDAR 2015, Total-Text and CTW-1500 to search for character training samples of their corresponding word-level annotations. The identified character samples are then combined with their original word-level annotations to fine-tune the pre-trained models on the new datasets. To better analyze the capability of the proposed TextFuseNet, we adopt ResNet with two different depths of $\{50, 101\}$ as the backbone on each dataset. Moreover, in order to enhance network robustness, data augmentation strategies such as multi-scale training, rotating randomly, and random color adjusting are applied.

Stochastic gradient descent (SGD) is adopted to optimize our framework. The weight decay is set to 0.0001, momentum is set to 0.9, and batch size is set to 8. In the pre-training stage, we train the model on SynthText for 20 epochs. The learning rate is set to 0.01 in the first 10 epochs, divided by 10 in the last 10 epochs. In the fine-tuning stage, the training iterations on every dataset are set to 20K. The learning rate is set to 0.005 in the first 10K iterations, divided by 10 in the remains.

Inference. During inference, the shorter side of the test image was scaled to 1000 while keeping the aspect ratio unchanged. The global semantic features are extracted in the semantic segmentation branch. For RPN generated text proposals, we select the top 1,000 proposals for the detection branch. With the detection results obtained, we adopt Soft NMS to suppress the redundant bounding boxes. The instance segmentation is then performed upon suppressed detection results. We only keep the instance segmentation results of word instances as the final text detection results.

Method	ICDAR2015			Total-Text		
	R	P	F	R	P	F
Baseline	83.8	87.4	85.5	80.5	81.5	81.0
MFR	86.3	90.3	88.3	82.2	85.2	83.7
MFR+MFA	**88.9**	**91.3**	**90.1**	**83.2**	**87.5**	**85.3**

Table 1: Performance contribution of each module in TextFuseNet. "MFR" represents multi-level feature representation, while "MFA" means the multi-path fusion architecture. "P", "R" and "F" represent Precision, Recall and F-measure respectively.

4.3 Ablation Study

Compared with the original Mask R-CNN, we introduce two modules to improve the performance of text detection in our proposed TextFuseNet. The first module is to conduct multi-level feature representation (MFR). The other is introducing multi-path features fusion architecture (MFA) to obtain richer fused features for text detection. Therefore, we conducted an ablation study on ICDAR 2015 and Total-Text to evaluate how each module in TextFuseNet influences final performance. For each dataset of ICDAR 2015 and Total-Text, three models are trained and the comparison results with different models are shown in Table 1. "**Baseline**" refers to the model trained with the original Mark R-CNN. "**MFR**" represents the model trained with Mask R-CNN using multi-level feature representation, and "**MFR+MFA**" refers to the model with full implementation of TextFuseNet. The backbone network used in this ablation study is a FPN with ResNet-50.

As is shown in Table 1, multi-level feature representation alone improves both precision and recall significantly, and the final improvements of "MFR" are more than F-measure of 2% on both ICDAR 2015 and Total-Text. Moreover, the combination of "MFR" and "MFA" can further enhance the performance and improves the F-measure by 4.6% and 4.3% beyond the baseline on ICDAR 2015 and Total-Text respectively. These results validate that both multi-level feature representation and multi-path features fusion can help to obtain

Method	ICDAR2013				ICDAR2015				Total-Text				CTW-1500			
	R	P	F	FPS	R	P	F	FPS	R	P	F	FPS	R	P	F	FPS
CTPN [Tian *et al.*, 2016]	83.0	93.0	88.0	7.14	52.0	74.0	61.0	7.1	-	-	-	-	-	-	-	-
SegLink [Shi *et al.*, 2017]	83.0	87.7	85.3	**20.6**	76.8	73.1	75.0	-	-	-	-	-	-	-	-	-
TextSnake [Long *et al.*, 2018]	-	-	-	-	80.4	84.9	82.6	1.1	74.5	82.7	78.4	-	85.3	67.9	75.6	-
TextBoxes++* [Liao *et al.*, 2018a]	86.0	92.0	89.0	-	78.5	87.8	82.9	2.3	-	-	-	-	-	-	-	-
RRD* [Liao *et al.*, 2018b]	86.0	92.0	89.0	-	80.0	88.0	83.8	-	-	-	-	-	-	-	-	-
PixelLink* [Deng *et al.*, 2018]	87.5	88.6	88.1	-	82.0	85.5	83.7	7.3	-	-	-	-	-	-	-	-
Mask TextSpotter [Lyu *et al.*, 2018]	88.6	95.0	91.7	4.6	81.0	91.6	86.0	4.8	55.0	69.0	61.3	-	-	-	-	-
TextField [Xu *et al.*, 2019]	-	-	-	-	80.5	84.3	82.4	6.0	79.9	81.2	80.6	-	79.8	83.0	81.4	-
CTD [Liu *et al.*, 2019a]	-	-	-	-	-	-	-	-	71.0	74.0	73.0	-	69.8	74.3	73.4	13.3
CSE [Liu *et al.*, 2019b]	-	-	-	-	-	-	-	-	79.1	81.4	80.2	2.4	76.0	81.1	78.4	2.6
MSR* [Xue *et al.*, 2019]	-	-	-	-	78.4	86.6	82.3	4.3	74.8	83.8	79.0	-	78.3	85.0	81.5	-
PSENet [Wang *et al.*, 2019a]	-	-	-	-	84.5	86.9	85.7	1.6	78.0	84.0	80.9	3.9	79.7	84.8	82.2	3.9
PAN [Wang *et al.*, 2019b]	-	-	-	-	84.0	81.9	82.9	**26.1**	**89.3**	81.0	85.0	**39.6**	81.2	86.4	83.7	**39.8**
ATRR [Wang *et al.*, 2019c]	89.7	93.7	91.7	-	86.0	89.2	87.6	-	76.2	80.9	78.5	-	80.2	80.1	80.1	-
SAE [Tian *et al.*, 2019]	-	-	-	-	85.0	88.3	86.6	-	-	-	-	-	77.8	82.7	80.1	-
SPCNet [Xie *et al.*, 2019]	90.5	93.8	92.1	-	85.8	88.7	87.2	-	82.8	83.0	82.9	-	-	-	-	-
LOMO* [Zhang *et al.*, 2019]	-	-	-	-	87.6	87.8	87.7	-	79.3	87.6	83.3	-	76.5	85.7	80.8	-
DB (ResNet-50) [Liao *et al.*, 2020]	-	-	-	-	83.2	91.8	87.3	12.0	82.5	87.1	84.7	32.0	80.2	86.9	83.4	22.0
Our (ResNet-50)	89.5	95.1	92.2	7.7	88.9	91.3	90.1	8.3	83.2	87.5	85.3	7.1	85.0	85.8	85.4	7.3
Our (ResNet-101)	**92.3**	**96.5**	**94.3**	4.0	**89.7**	**94.7**	**92.1**	4.1	85.3	**89.0**	**87.1**	3.3	**85.4**	**87.8**	**86.6**	3.7

Table 2: Evaluation results on different datasets. "$*$" means multi-scale inference.

richer fused features and more discriminative representations which are beneficial for text detection.

4.4 Comparisons with State-of-the-Art Methods

Arbitrary Shape Text Detection. As mentioned above, CTW-1500 and Total-Text focus on text with arbitrary shapes, in which horizontal, multi-oriented and curved text exist simultaneously in most of images. Therefore, we use these two datasets to evaluate the effectiveness of TextFuseNet in detecting texts with arbitrary shapes. The last two columns of Table 2 lists the results of TextFuseNet compared to some previous methods on CTW-1500 and Total-Text, respectively. Note that FPS is only for reference, since different GPUs are adopted in different methods. As shown in Table 2, our proposed TextFuseNet using single-scale inference achieves state-of-the-art performance on both CTW-1500 and Total-Text. Specifically, in CTW-1500, TextFuseNet with the backbone of ResNet-50 achieves an F-measure of 85.4%, higher than the current best one by 1.7%. When the backbone is ResNet-101, a more compelling result can be achieved (F-measure: 86.6%), outperforming all the other competitors by at least 2.9%. Similarly, for Total-Text, our TextFuseNet with ResNet-50 already achieves state-of-the-art result and its ResNet-101 version outperforms other approaches by at least 2.1%. The above experimental results show that TextFuseNet can access the state-of-the-art performance on arbitrary shape text detection.

Multi-oriented Text Detection. We also evaluated the effectiveness of TextFuseNet in detecting multi-orient text on ICDAR 2015. Our results and comparison with previous works are shown in the third column of Table 2. As Table 2 shows, TextFuseNet with backbone of ResNet-50 and ResNet-101 achieve state-of-the-art performance, and their F-measure are 90.1% and 92.1% respectively. Compared with the current best one, our ResNet-50 and ResNet-101 version outperform it by 2.4% and 4.4% respectively. Moreover, to the best of our knowledge, our proposed framework is the first one on ICDAR 2015 with an F-measure over 90.0%.

Horizontal Text Detection. Finally, we evaluated the effectiveness of TextFuseNet in detecting horizontal text on ICDAR 2013. The results of TextFuseNet and comparison with previous works are presented in the second column of Table 2. TextFuseNet with the backbone of ResNet-50 and ResNet-101 both achieve very outstanding results, and the F-measure are 92.2% and 94.3% respectively, which outperform all previous works.

Therefore, from these experimental results on ICDAR 2013, ICDAR 2015, Total-Text and CTW-1500, our proposed TextFuseNet achieves state-of-the-art performance. In addition, for the speed, TextFuseNet can also inference at an appropriate speed, which has a degree of ascendancy compared to some previous methods. Some examples using TextFuseNet are shown in Figure 4.

5 Conclusion

In this paper, we propose a novel framework TextFuseNet for arbitrary shape text detection by investigating on three levels of features, *i.e.*, character-, word- and global-level features. Different level features are fully and finely explored to learn richer fused features, which are beneficial to text detection. Our experimental results show that TextFuseNet achieves state-of-the-art performance in detecting text with arbitrary shape.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grants (No.61822113, No.62041105), Australian Research Council Project (No.FL-170100117), the Natural Science Foundation of Hubei Province under Grants (No.2018CFA050) and the Science and Technology Major Project of Hubei Province (Next-Generation AI Technologies) under Grant (No.2019AEA170). The numerical calculations in this paper have been done on the supercomputing system in the Supercomputing Center of Wuhan University.

References

[Deng *et al.*, 2018] Dan Deng, Haifeng Liu, Xuelong Li, and Deng Cai. Pixellink: Detecting scene text via instance segmentation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[Gao *et al.*, 2019] Mingyu Gao, Yujie Du, Yuxiang Yang, and Jing Zhang. Adaptive anchor box mechanism to improve the accuracy in the object detection system. *Multimedia Tools and Applications*, 78(19):27383–27402, 2019.

[He *et al.*, 2017] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[Liao *et al.*, 2018a] Minghui Liao, Baoguang Shi, and Xiang Bai. Textboxes++: A single-shot oriented scene text detector. *IEEE transactions on image processing*, 27(8):3676–3690, 2018.

[Liao *et al.*, 2018b] Minghui Liao, Zhen Zhu, Baoguang Shi, Gui-song Xia, and Xiang Bai. Rotation-sensitive regression for oriented scene text detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5909–5918, 2018.

[Liao *et al.*, 2020] Minghui Liao, Zhaoyi Wan, Cong Yao, Kai Chen, and Xiang Bai. Real-time scene text detection with differentiable binarization. *In Thirty-Fourth AAAI Conference on Artificial Intel-ligence*, 2020.

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

[Liu *et al.*, 2019b] Zichuan Liu, Guosheng Lin, Sheng Yang, Fayao Liu, Weisi Lin, and Wang Ling Goh. Towards robust curve text detection with conditional spatial expansion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7269–7278, 2019.

[Long *et al.*, 2018] Shangbang Long, Jiaqiang Ruan, Wenjie Zhang, Xin He, Wenhao Wu, and Cong Yao. Textsnake: A flexible representation for detecting text of arbitrary shapes. In *Proceedings of ECCV*, pages 20–36, 2018.

[Lyu *et al.*, 2018] Pengyuan Lyu, Minghui Liao, Cong Yao, Wenhao Wu, and Xiang Bai. Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. In *Proceedings of ECCV*, pages 67–83, 2018.

[Qiao *et al.*, 2019] Tingting Qiao, Jing Zhang, Duanqing Xu, and Dacheng Tao. Mirrorgan: Learning text-to-image generation by redescription. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1505–1514, 2019.

[Shi *et al.*, 2017] Baoguang Shi, Xiang Bai, and Serge Belongie. Detecting oriented text in natural images by linking segments. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2550–2558, 2017.

[Tian *et al.*, 2016] Zhi Tian, Weilin Huang, Tong He, Pan He, and Yu Qiao. Detecting text in natural image with connectionist text proposal network. In *Proceedings of ECCV*, pages 56–72. Springer, 2016.

[Tian *et al.*, 2019] Zhuotao Tian, Michelle Shu, Pengyuan Lyu, Ruiyu Li, Chao Zhou, Xiaoyong Shen, and Jiaya Jia. Learning shape-aware embedding for scene text detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4234–4243, 2019.

[Wang *et al.*, 2017] Zheng Wang, Ruimin Hu, Chen Chen, Yi Yu, Junjun Jiang, Chao Liang, and Shin'ichi Satoh. Person reidentification via discrepancy matrix and matrix metric. *IEEE transactions on cybernetics*, 48(10):3006–3020, 2017.

[Wang *et al.*, 2019a] 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 IEEE International Conference on Computer Vision*, 2019.

[Wang *et al.*, 2019b] Wenhai Wang, Enze Xie, Xiaoge Song, Yuhang Zang, Wenjia Wang, Tong Lu, Gang Yu, and Chunhua Shen. Efficient and accurate arbitrary-shaped text detection with pixel aggregation network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8440–8449, 2019.

[Wang *et al.*, 2019c] Xiaobing Wang, Yingying Jiang, Zhenbo Luo, Cheng-Lin Liu, Hyunsoo Choi, and Sungjin Kim. Arbitrary shape scene text detection with adaptive text region representation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6449–6458, 2019.

[Wang *et al.*, 2019d] Zheng Wang, Junjun Jiang, Yang Wu, Mang Ye, Xiang Bai, and Shin'ichi Satoh. Learning sparse and identity-preserved hidden attributes for person re-identification. *IEEE Transactions on Image Processing*, 29(1):2013–2025, 2019.

[Xie *et al.*, 2019] Enze Xie, Yuhang Zang, Shuai Shao, Gang Yu, Cong Yao, and Guangyao Li. Scene text detection with supervised pyramid context network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9038–9045, 2019.

[Xu *et al.*, 2019] 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*, 2019.

[Xue *et al.*, 2019] Chuhui Xue, Shijian Lu, and Wei Zhang. Msr: Multi-scale shape regression for scene text detection. *In Proceedings of the 28th International Joint Confer-ence on Artificial Intelligence*, 2019.

[Zhang *et al.*, 2019] 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 IEEE Conference on Computer Vision and Pattern Recognition*, 2019.