Semi-Supervised Pixel-Level Scene Text Segmentation by Mutually Guided Network

Chuan Wang, Shan Zhao, Li Zhu, Kunming Luo, Yanwen Guo, *Senior Member, IEEE*, Jue Wang, *Senior Member, IEEE*, and Shuaicheng Liu, *Member, IEEE*

***Abstract*— In this paper we present a new data-driven method for pixel-level scene text segmentation from a single natural image. Although scene text detection, i.e. producing a text region mask, has been well studied in the past decade, pixel-level text segmentation is still an open problem due to the lack of massive pixel-level labeled data for supervised training. To tackle this issue, we incorporate text region mask as an auxiliary data into this task, considering acquiring large-scale of labeled text region mask is commonly less expensive and time-consuming. To be specific, we propose a mutually guided network which produces a polygon-level mask in one branch and a pixel-level text mask in the other. The two branches' outputs serve as guidance for each other and the whole network is trained via a semi-supervised learning strategy. Extensive experiments are conducted to demonstrate the effectiveness of our mutually guided network, and experimental results show our network outperforms the state-of-the-art in pixel-level scene text segmentation. We also demonstrate the mask produced by our network could improve the text recognition performance besides the trivial image editing application.**

***Index Terms*— Semi-supervised, scene text segmentation, mutually guided network.**

I. Introduction

SCENE text detection is the process of localizing text regions in a natural image, which has been widely studied in the past decade. With the increasing development of Convolutional Neural Networks (CNNs), a series of works [2]–[4] have been proposed in the research community. However, this task only produces coarse-grained masks of the texts, which limits its application to more fine-grained scenarios such as image editing (copying-and-pasting the text, changing the visual effects of text, etc.) or inpainting. As a result, pixel-level scene text segmentation has become an emerging topic recently, which is proven quite challenging. The difficulties result from that, in the wild, scene texts may differ in various colors, fonts and the aligned shapes. Early image processing based algorithm MSER [5] + SWT [6] simply detects the texts based on the prior knowledge of stroke width and connected regions in the image, lacking of enough learning based mechanism. Therefore, its performance is far from being applicable to the images in the wild. CNNs based methods, on the other hand, are proven effective for learning various cases, while its bottleneck lies in its high reliance on the large scale data, especially that manually labeled by human beings. For this reason, considering the existing available datasets TotalText [7] and ICDAR-2013 [8] only contain less than $3k$ samples, Bonechi *et al.* [9] propose two relatively large datasets COCO-TS and MLT-S, making the number around $20k$. However, as the datasets are labeled by the machine, the pixel-level mask ground truths contain much noise so as to be unsuitable for strong supervised training.

To this end, we propose to improve pixel-level scene text segmentation by using the available datasets as semi-supervision to train a neural network. We take advantages of the polygon-level masks (or bounding boxes) of the texts, which are easily obtained in the existing datasets for text detection, to serve as a guidance for the pixel-level text segmentation. Specifically, the polygon-level masks can benefit the pixel-level text segmentation in two folds. First, it provides a prior knowledge to localize the text regions, working as an attention map to guide the neurons where to pay more attention. Second, it also works as a posterior probability map to filter out the false positive regions in a segmented pixel-level mask. Actually, this type of guidance can also work in an opposite manner, i.e. making the pixel-level masks guide the prediction of the polygon-level masks, so that the segmentation of the two grained masks are mutually guided. For this reason, we design a dual-task mutually guided neural network, which contains a shared encoder but two decoders for the pixel-level and polygon-level masks separately. The output from each decoder is fed to the other one so as to form a recurrent loop as shown in Fig. 2(a). The shared encoder of the two branches extracts common feature maps from the input image, considering the similarity of the two tasks and the compactness of the network. This structure enables us to train the pixel-level text segmentation network without increasing the annotated pixel-level mask ground truths as training data but simply adding more polygon-level masks. Since the ground truth of each training sample may not contain two types of masks simultaneously, we train the

Manuscript received August 17, 2020; revised July 22, 2021 and August 14, 2021; accepted August 14, 2021. Date of publication September 21, 2021; date of current version September 30, 2021. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61872067 and Grant 61720106004 and in part by Sichuan Science and Technology Program under Grant 2019YFH0016. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Shuicheng Yan. *(Chuan Wang and Shan Zhao are co-first authors.) (Corresponding author: Shuaicheng Liu.)*

Chuan Wang, Shan Zhao, Kunming Luo, and Jue Wang are with Megvii Technology, Beijing 100190, China.

Li Zhu is with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China.

Yanwen Guo is with the National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China.

Shuaicheng Liu is with the School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: liushuaicheng@uestc.edu.cn).

Digital Object Identifier 10.1109/TIP.2021.3113157

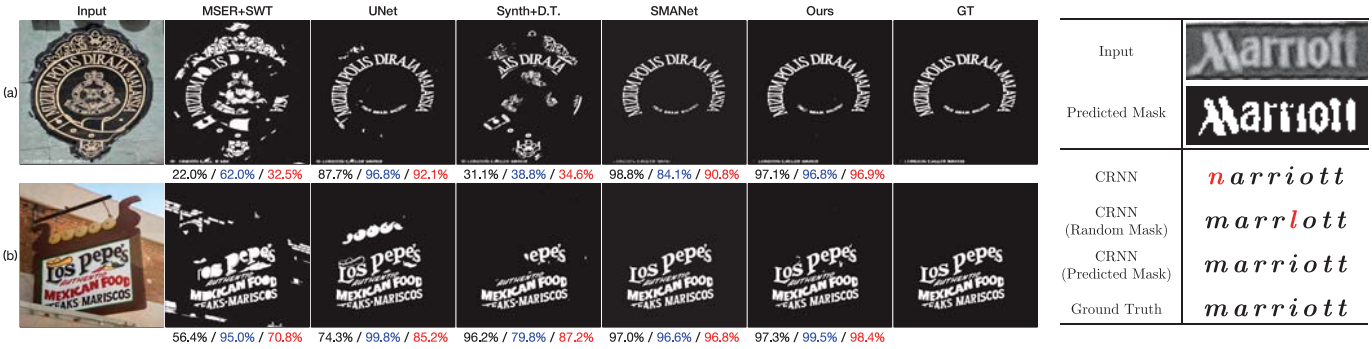


Fig. 1. Our pixel-level scene text segmentation results compared with the state-of-the-art. With the predicted masks by our method, text recognition accuracy of existing frameworks could be improved as seen in the rightmost column, taking CRNN [1] as an example.

network in a semi-supervised manner with a newly introduced loss and a corresponding training strategy. Experimental results demonstrate the effectiveness of all the newly involved techniques for our network, and quantitative evaluations also show that our network outperforms state-of-the-art methods as shown in Fig. 1, 6 and Table II for pixel-level scene text segmentation task. We also demonstrate that the pixel-level text mask could work as an attention map to improve the text recognition accuracy. To summarize, our main contributions are:

- A novel uniform dual-task mutually guided neural network for text segmentation in polygon-level and pixel-level simultaneously.
- A newly designed loss and customized training strategy within a semi-supervised fashion, which performs well in the case of missing ground truth information.
- Ablation studies to demonstrate the enhancement of the text recognition accuracy brought by the pixel-level mask, in addition to the involved techniques for our network.

II. Related Work

A. Text Detection

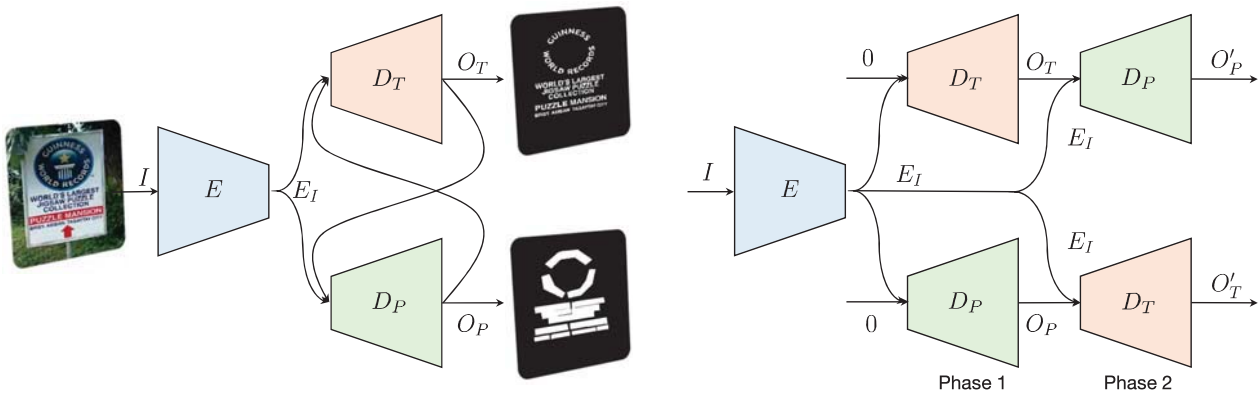
Text detection has been studied in the past decade, during which a large number of approaches were proposed. This task aims at localizing text regions by polygon or rectangle boxs instead of pixel-level masks. Existing methods can be grouped into two categories, i.e. detection-based and segmentation-based. The former ones mostly draw inspiration from the general object detection frameworks such as R-CNN [10], [11] or SSD [12]. For example, as the variants of the SSD [12], TextBoxes [2] proposes a well-designed reference box to deal with variation of aspect ratios of text instances. And RRPN [3] handles the orientation of scene text via rotating both anchors and RoIPooling in Faster R-CNN [10]. EAST [4] directly regresses the geometries of text instances. A common problem for this kind of method is that they are difficult to handle very long or arbitrarily shaped text. On the other hand, segmentation-based method converts the problem of text detection into semantic segmentation. Yao *et al.* [13] first produces dense maps via FCN [14], which implies the attributes of the scene text. Zhang *et al.* [15] utilizes FCN [14] and MSER [5] to extract text blocks and select character candidates. PixelLink [16] obtains different text instances by predicting the connection between pixels and predicting the classification of pixels. Although these methods solve the text detection by segmentation, they rarely obtain pixel-level masks for the texts so as not to be viewed as pure segmentation solutions.

B. Pixel-Level Text Segmentation

Recently, as CNNs have achieved the state-of-the-art performance in many computer vision tasks, the fine-grained text detection has drawn attention in the research community, and there appears several works focusing on pixel-level scene text segmentation [9], [17]. For example, realizing that training a CNN is unachievable due to the availability of less than $3k$ pixel-level annotated images in the only two public datasets, i.e. ICDAR-2013 [8] and TotalText [7], Bonechi *et al.* [9] propose a weakly supervised method to generate the pixel-level annotations for COCO-Text [18] dataset and MLT dataset [19], producing two new datasets COCO-TS and MLT-S. However, their method follows an explicit manner in three separate steps, i.e. first they train a background-foreground network (BFNet) on synthetic bounding-box-scaled images, then they apply this BFNet to dataset COCO-Text, generating a text probability map followed by a hard threshold clipping operation, obtaining the COCO-TS dataset. Finally they use COCO-TS to train a Segmentation Multiscale Attention Network (SMANet). The problem in this method lies in that the ground truth dataset COCO-TS is also machine-generated, its quality is far from that of human-annotation. Unlike this approach, our network utilizes the polygon-level masks as weak supervision in an implicit manner, designing customized semi-supervised loss for end-to-end training. As a result, our method is flexible to be generalized for more data and achieves a better performance.

C. Multi-Task CNN

Multi-task learning (MTL) has been widely used in many applications of computer vision. It aims to improve learning efficiency and prediction accuracy for each task. A MTL method is typically achieved by sharing some hidden layers among multiple tasks. For example, Kendall *et al.* [20] uses a shared encoder and three decoders to get depth prediction, semantic segmentation and instance segmentation separately. To reduce the learning difficulty, some works decompose a complex task into multiple related sub-tasks and then fuse



(a) Our mutually guided dual-task network (b) Our network decoupled in two phases

Fig. 2. Structure of our mutually guided dual-task network. (a) The original network with loop. (b) The 2 stages decoupled network for ease of analyzing.

together. For example, Wang *et al.* [21] utilizes a dual-task neural network that jointly learns spatial details and temporal coherence for a video inpainting task, and Cai *et al.* [22] proposes a two-stream structure to learn features on both the trimap adaptation and the alpha estimation jointly in an image matting task. Following the similar idea, we compose a dual-task network for polygon-level and pixel-level text segmentation simultaneously. Additionally, the output from the two tasks serve as guidance for each other, instead of simply sharing an encoder or feature maps. As a result, our network could be viewed as a variant of the classic dual-task network.

D. Application to Video Segmentation

As a semi-supervised method, our idea of using guidance to facilitate video object segmentation also has application potential to video segmentation task as [23], [24], [25], [26]. For example, [23] pre-trains a segmentation network using generic dataset and finetunes it with the labeled first frame. Similarly, with our idea, a branch could possibly be added to their backbone network to enable mutual guidance training, i.e. this branch producing the particular object mask and the original backbone doing the coarse-grained object mask, and both segmentation outputs are mutually guided. [24] applies embedding learning to foreground pixels then uses the embedding from the labeled first frame and previous frame to guide the segmentation of the current frame. [25] further extends this idea to foreground and background to improve the segmentation robustness. [26] proposes an interactive framework to perform segmentation under the guidance of a memory aggregation module that records the information from the previous interaction rounds. Though challenging, the mutual guidance idea also has the potential to be applied in these cases, for example a possible idea is reversely guiding the optimization of previous frame after current frame is done. We hope our method could inspire more works in the video segmentation area in the future.

III. Algorithm

Our method is built upon dual-task fully convolutional neural networks. It takes a single RGB image I as input, and produces probability maps of pixel-level text O_T and polygon-level text O_P as outputs. The network contains two decoders D_T, D_P in separate branches, for estimating the two probability maps respectively. The two decoders share a common encoder E, which extracts the feature map of I as E_I. The output O_T, together with E_I, are fed to the decoder of D_P in the other branch, and vice versa. For either decoder, the output from the other branch serves as a guidance for the task, making the dual tasks mutually guided. The network structure is illustrated at Fig. 2(a).

A. Mutually Guided Network

Our network is a mutually guided dual-task network, which can be viewed as a recurrent network due to the signal loop in the structure. For ease of analysis, we decoupled the original network into two phases with no loop as shown in Fig. 2(b). To distinguish the symbols in the two phases, for any symbol appearing in Phase 1, we add a prime notation $'$ on it to represent the symbol in Phase 2. For example, we use G and G', as well as O and O' to represent guidances and outputs in Phase 1 and 2 respectively. And for briefness, we use $\kappa \in \{T, P\}$ to represent a *pixel-level* or *polygon-level* module or variable. A set of $\{X_\kappa\}$ represents both X_T and X_P where X is any applicable symbol in our model. For example, $\{D_\kappa\}$ means D_T and D_P, representing the pixel-level and polygon-level decoders (detailed in later sections).

In Phase 1, we feed decoders $\{D_\kappa\}$ with guidances $\{G_\kappa\}$ and produce outputs $\{O_\kappa\}$ as intermediate results. Then we feed the decoders with $\{G'_\kappa\}$ in Phase 2 and produce the final outputs $\{O'_\kappa\}$. For two phases, the input I and the weights in E and $\{D_\kappa\}$ are identical, while guidances in two phases are commonly various, i.e. $G_\kappa \neq G'_\kappa$ for $\kappa \in \{T, P\}$. That is because in Phase 1, normally we have very limited or even no information to provide, while in Phase 2 we have the initial results $\{O_\kappa\}$ estimated to serve as guidances. Moreover, here we design a shared encoder E instead of two independent ones, not only for reducing redundancy, but also for extracting the common feature map E_I for the successive two tasks. To summarize, the entire network can be written as follows,

$$\text{Phase 1} \begin{cases} G_T = 0, & G_P = 0 \\ O_T = D_T(E_I, G_T), & O_P = D_P(E_I, G_P) \end{cases}$$

TABLE I
LAYER CONFIGURATION FOR ENCODER E AND DECODERS D_T AND D_P

	Type	Kernel	Stride	Channel
1)	*conv*	3	1	64
2)	*conv*	3	1	64
3)	*maxpool*	2	2	64
4)	*conv*	3	1	128
5)	*conv*	3	1	128
6)	*maxpool*	2	2	128
7)	*conv*	3	1	256
8)	*conv*	3	1	256
9)	*maxpool*	2	2	256
10)	*conv*	3	1	512
11)	*conv*	3	1	512
12)	*maxpool*	2	2	512
13)	*conv*	3	1	512
14)	*conv*	3	1	512

(a) Encoder E

	Type	Kernel	Stride	Channel
1)	*upsample*	2	/	512
2)	*conv*	3	1	256
3)	*conv*	3	1	256
4)	*upsample*	2	/	256
5)	*conv*	3	1	128
6)	*conv*	3	1	128
7)	*upsample*	2	/	128
8)	*conv*	3	1	64
9)	*conv*	3	1	64
10)	*upsample*	2	/	64
11)	*conv*	3	1	64
12)	*conv*	3	1	64

(b) Decoder D_T and D_P

$$\text{Phase 2} \begin{cases} G'_T = O_P, & G'_P = O_T \\ O'_T = D_T(E_I, G'_T), & O'_P = D_P(E_I, G'_P) \end{cases}$$

For the structures of E and $\{D_\kappa\}$, we adopted a FCN including 4 downsampling blocks in E and 4 upsamping blocks in D_κ. The size of input I is $512^2 \times 3$ so that the feature maps between E and D_κ, i.e. E_I is of size $32^2 \times 512$. We also applied an encoder of the same structure as E to the guidance $\{G_\kappa\}$, to ensure its extracted feature can be well concatenated to E_I, after they are fed to D_κ. For each fully convolutional layer, the kernel size is set to 3×3, and is followed by a BatchNorm and a ReLU layer. We list the layer configurations in Table I.

B. Learning Algorithm

Training a pixel-level text segmentation network is not a straight-forward task. Its main challenge results from the very limited human-annotated training data. Despite the fact that synthesizing the training data is not expensive, the trained model is unable to applied to real data due to domain gap existence. We realized that due to recent development for text detection task, it is relatively easy to obtain the polygon-level mask for an image. So in our problem, each data pair contains ground truth pixel-level or polygon-level mask only, noted as M_T and M_P. Since there is few training data containing both M_T and M_P, our task naturally becomes semi-supervised, which is achieved by a customized loss as follows.

1) Semi-Supervised Loss: Our customized semi-supervised loss is composed of three parts, including strongly-supervised and weakly-supervised ones. The former one is the SoftIoU loss between the output and the ground truth; and the latter ones include a so-called subset loss and a Conditional Random Field (CRF) loss.

a) SoftIoU loss: As aforementioned, the training data provided to our problem, is a data pair $\langle I, M_\kappa \rangle$ with M_κ being either pixel-level or polygon-level ground truth mask, i.e. $\kappa = T$ or $\kappa = P$. If $\kappa = T$, we compute the SoftIoU losses between M_T and its outputs O_T, O'_T respectively, making them strong supervision to the pixel-level text segmentation task. Similarly, it also applies to data sample if $\kappa = P$, so that we produce a sum of four terms of SoftIoU losses:

$$\mathcal{L}_{IoU} = \sum_{\substack{\kappa \in \{T,P\}, \\ X \in \{O_\kappa, O'_\kappa\}}} l_\kappa \cdot \frac{-\sum_i X_i M_{\kappa,i}}{\sum_i (X_i + M_{\kappa,i} - X_i M_{\kappa,i})} \tag{1}$$

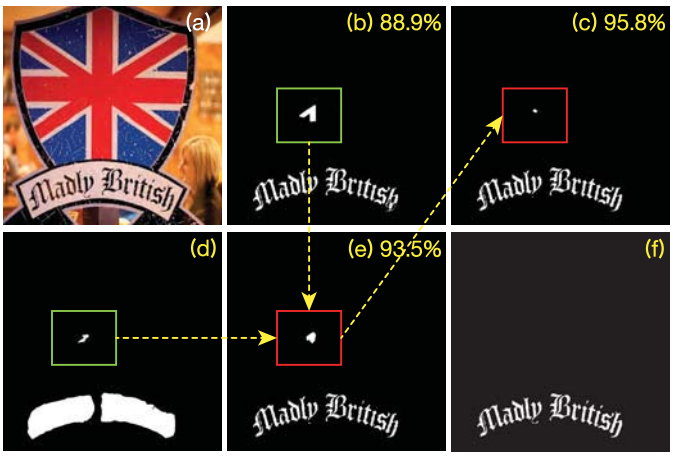


Fig. 3. Effectiveness of losses $\mathcal{L}_{SB}$ and $\mathcal{L}_{CRF}$. (a) The input RGB image. (b) without $\mathcal{L}_{SB}$ enabled, a large false-positive region (green box) is predicted. With $\mathcal{L}_{SB}$ enabled, the polygon-level mask output (d) corrects parts of the false-positive region, making the region smaller (e). With $\mathcal{L}_{CRF}$ involved, the false-positive region is further shrunk (c). (f) The ground truth of the pixel-level mask.

where i represents a pixel location and $l_\kappa \in \{0, 1\}$ is an indicator function to indicate the type of the input data pair, defined as

$$l_\kappa = \mathbf{1}_{\langle I, M_\kappa \rangle}\big(\langle I, M_{\kappa'} \rangle\big) = \begin{cases} 1, & \text{if } \langle I, M_{\kappa'} \rangle = \langle I, M_\kappa \rangle \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where $\kappa' \in \{T, P\}$. l_κ works like a switch to enable or disable the contribution of a loss. Same notation applied for the rest of this paper.

b) Subset loss: It is a prior knowledge that the pixel-level text mask should be a subset of the polygon-level mask for the input image, i.e. O_T should be in O_P. To model such subset relationship, we introduce a so-called *subset loss* defined by a weighted cross-entropy loss:

$$\mathcal{L}_{SB} = \sum_{\substack{Z \in \{O_T, O'_T\}, \\ Y \in \{O_P, O'_P\}}} \sum_i Z_i \cdot L_{CE}(Z_i, Y_i) \tag{3}$$

where $L_{CE}(Z_i, Y_i) = Y_i \log(Z_i) + (1 - Y_i) \log(1 - Z_i)$. This loss penalizes the case if a pixel is assigned by a high probability of pixel-level text mask but a low probability of polygon-level text mask, but not vice versa. In other words, it could potentially reduce the false positive rate if the polygon-level mask is well predicted.

c) CRF loss: We also involve a CRF loss to refine the mask output for the branch with no ground truth mask being provided, written as

$$\mathcal{L}_{CRF} = \sum_{\substack{\kappa \in \{T,P\}, \\ X \in \{O_\kappa, O'_\kappa\}}} (1 - l_\kappa) \cdot L_{CRF}(X, I) \tag{4}$$

Here $L_{CRF}(X, I) = x^\top A x$, where x is a column vector by flattening X, and A is an affinity matrix computed from I's pixel color. Please refer to [27] for more detailed explanations. Generally, it constrains the neighboring pixels in I with similar color to hold a consistent label in O_κ, so as to potentially increase the segmentation accuracy.

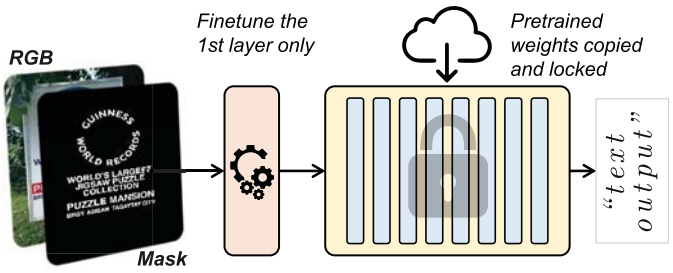


Fig. 4. Illustration of our text recognition network and training method.

The subset and CRF losses are two weakly-supervised losses. Together with SoftIoU loss as strong supervision, they weakly take effect to improve the performance. To sum up, our semi-supervised loss is

$$\mathcal{L} = \mathcal{L}_{IoU} + \lambda_1 \cdot \mathcal{L}_{SB} + \lambda_2 \cdot \mathcal{L}_{CRF} \tag{5}$$

where λ_1 and λ_2 are the balancing hyper-parameters, being set to 10 and 0.1 in our experiments. Table IV reveals the effectiveness of the two weakly-supervised losses, by comparing the performances resulted from all losses (Ours) or part losses (Ours-β, Ours-γ) involved.

2) Training Strategy: We train the proposed mutually guided dual-task network with an empirical strategy. Specifically, we divide the entire training into two separate phases. In Phase 1, we disconnect the loop between the two branches and train the classical dual-task network directly. As mentioned above, we feed the network with batches composed of data pairs $\langle I, M_\kappa \rangle, \kappa \in \{T, P\}$ and the guidance inputs G_T, G_P are fed with 0 as there are no such information. For making the network easier to converge, we only use SoftIoU loss in this stage. In Phase 2, we connect the loop between the two branches and add both subset loss and CRF loss into the total loss. And, we set $G_T = O_P, G_P = O_T$ for the outputs O_T, O_P are gradually informative. With the mutual guidance involved, we found the performance further increased. We demonstrate the increase in an ablation study in Section IV-C.1.

C. Improving Text Recognition

With the pixel-level mask being predicted, its potential efficacy to text recognition is further investigated. Specifically, we modify the input layer of an existing text recognition network and feed it with 4-channel images (RGB + mask) instead of the original 3-channel ones. Then we copy the weights from a pre-trained model of the original version of the network to this modified network, except for the newly replaced input layer, which is for further fine-tuning until convergence (Fig. 4). We found that with this additional mask channel involved, the recognition accuracy is generally improved, as demonstrated in an ablation study detailed in Section IV-C.3, comparing our result with the one by the original text recognition network (3-channel image as input) and the one by setting the 4-th channel to a purely random mask. The random mask is generated by randomly setting each pixel on an all-zero image to 1 with 50% probability, and this random mask serves as a dummy channel for fair comparison with the case of setting the 4-th channel to our produced mask. We believe this performance gain reveals that the mask potentially works as an attention map to guide where to focus on for the recognition network. We demonstrate this phenomenon in two state-of-the-art networks, CRNN [1] and ASTER [28]. Detailed performance values and illustrations are shown in Table IV(b) and Fig. 9.

IV. Experimental Results

A. Datasets and Implementation Details

1) Datasets: The training and validation of our network involve four datasets of real texts. They are

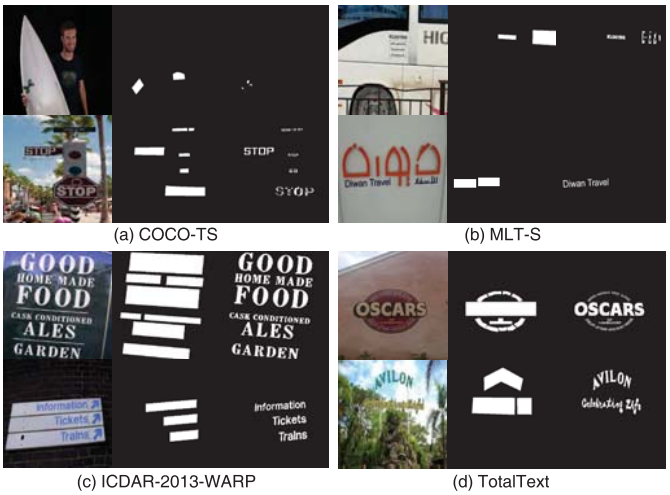
1) COCO-TS [9]: a subset of COCO-Text [18] with 14,690 images. Each one has at least one polygon bounding box.
2) MLT-S [9]: a subset of MLT dataset having 6,896 images.
3) ICDAR-2013-WARP [8]: A warped version of ICDAR-2013 dataset, which contains 229 training and 233 validation images with pixel-level mask ground truth. The box for each text in the original ICDAR-2013 dataset is always rectangle, which is too trivial to serve as our polygon-level mask. Therefore, we randomly warp each image to generate ICDAR-2013-WARP for our experiments.
4) TotalText [7]: it contains 1,255 training images and 300 test images. Unlike ICDAR-2013-WARP, texts here have arbitrary shapes so that the polygon-level masks are complicated enough. Pixel-level mask ground truth exists.

Note that for TotalText and ICDAR-2013-WARP, as the dataset scale is small ($\approx 1,500$), both pixel-level and polygon-level masks are used in our experiments. However, for COCO-TS and MLT-S, although the original datasets provide pixel-level masks for the images, as they are machine-generated and of low-quality, we ignore them and use the polygon-mask only, making our experiments run in a highly weakly-supervised manner for a large scale of images. For ease of understanding, please see Fig. 5 for a glance of these data samples.

2) Implementation Details: Our code is implemented in PyTorch, and the whole training usually costs 48 hours by one NVIDIA GeForce GTX 1080Ti GPU. During training we feed the network with images of size 512×512, and data augmentation including random rotation, saturation, hue, noise, brightness and contrast is applied for the training images. As for the 2-phase training strategy, in practice, in Phase 1 we apply Adam optimizer with a learning rate of 1.0×10^{-4} to the network with the entire COCO-TS and MLT-S datasets for the first 24 hours, and then with the training sets of TotalText and ICDAR-2013-WARP. Then in Phase 2, the learning rate is reduced to 1.0×10^{-5} and only the training sets of TotalText and ICDAR-2013-WARP are left for finetuning. For our competing algorithms, the dataset setting is the same as that in our Phase 1.

B. Comparison With Existing Methods

We compared our method with some state-of-the-art ones, including one traditional image processing algorithm, Maximally Stable Extremal Regions (MSER) [5] + Stroke Width



(a) COCO-TS (b) MLT-S

(c) ICDAR-2013-WARP (d) TotalText

Fig. 5. A glance of the training data samples from COCO-TS, MLT-S, ICDAR-2013-WARP and TotalText (a) - (d) for our experiments. In each subfigure, columns from left to right are input images, polygon-level masks and pixel-level masks. As seen, in COCO-TS and MLT-S, the pixel-level masks are of low quality so that we do not use them in our experiments.

TABLE II

STANDARD COMPARISON. F1, F1 TOP-1 RATE, PRECISION AND RECALL FOR STATE-OF-THE-ART METHODS AND OURS, IN DATASETS TOTALTEXT (BLACK) AND ICDAR-2013-WARP (BLUE)

	F1 (%)	F1 Top-1 (%)	Precision (%)	Recall (%)
MSER [5]+SWT [6]	33.3 / 43.8	0.67 / 3.00	24.9 / 39.4	72.3 / 69.6
CENet [17]	60.5 / 59.9	0.67 / 3.00	63.1 / 62.7	63.5 / 63.7
SegNet [29]	72.3 / 68.1	4.67 / 3.43	78.7 / 75.4	72.3 / 68.8
UNet [30]	75.3 / 70.8	14.0 / 18.5	81.2 / 76.8	75.9 / 73.0
DeepLab-v3 [31]	54.9 / 56.7	1.33 / 3.86	48.1 / 52.1	70.0 / 71.2
SMANet [9]	77.5 / 71.3	28.7 / 14.2	**86.6** / 74.4	73.9 / 73.8
Ours (Synth+D.T.)	50.0 / 59.1	2.33 / 15.5	60.2 / 73.9	52.0 / 56.2
Ours	**80.5** / **74.5**	**47.7** / **38.6**	83.3 / **79.0**	**81.6** / **77.0**

Transform (SWT) [6], five deep learning based algorithms [9], [17], [29]–[31] and our own network trained using synthetic dataset. The method of MSER+SWT detects the connected regions in an image and analyzes each region by SWT, to finally determine whether the region is a text or not. As this method has no learning mechanism, its performance is far from robustness for images with complex background in the wild. The five deep learning algorithms are state-of-the-art image segmentation networks [29], [30], which mainly differ in the structures. As these networks are not originally designed to support polygon-level mask data, we trained them using the same real pixel-level mask datasets without the polygon-level ones. We also trained a model with our network using synthesized data and then domain-transferred it by real data, i.e. the method Ours (Synth+D.T.) in Table II. The reason to add this comparison is that for text segmentation, synthesized training data is relatively affordable than that for general object segmentation, while the domain gap between the synthesized and real data always exists for the final deployment.

1) Quantitative Evaluation: We trained the six networks to convergence with multiple trials and selected their best results. To quantitatively compare our method with them, we evaluated precision ($\mathcal{P}$), recall ($\mathcal{R}$), and F1 score ($F1 = \frac{2\cdot\mathcal{P}\cdot\mathcal{R}}{\mathcal{P}+\mathcal{R}}$) of all the results, and list them at Table II. The data shows that in terms of the 3 metrics, our approach outperforms the state-of-the-art for pixel-level text segmentation, for both datasets TotalText (in black) and ICDAR (in blue), except on the TotalText dataset, our precision is the 2nd best with a 3% lower value. For the mean F1 score of all the validation data samples, our method is about 3% higher than the 2nd best in average. We also compute an F1 Top-1 rate ($\mathcal{T}$), i.e. the percentage of data each method can beat the others in terms of F1 score. For the j-th method, its F1 Top-1 rate $\mathcal{T}_j$ is calculated as

$$\mathcal{T}_j = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}_{\arg\max_k \{F1_i^k\}}(j), \quad \begin{cases} j = 1, 2, \ldots, M \\ k = 1, 2, \ldots, M \end{cases} \tag{6}$$

where N, M are the number of samples and methods separately. We found our method won for nearly 40% validation data and none of the others has a comparable performance. We illustrate the curves of F1, precision, and recall at Fig. 7 for the TotalText dataset.

2) Qualitative Evaluation: We list several typical segmented pixel-level text masks in Fig. 6 and Fig. 1 for qualitative comparison, where the examples cover various text colors / fonts / alignments, complex illuminance, areas in the images etc. Fig. 6(a) shows a flag of Chelsea and the background contains similar colors and patterns, (b)(c)(e) shows texts in various colors and aligned horizontally and circularly, in high or low contrast conditions. (d) contains texts in very artistic fonts. (f)(i) show examples with texts captured in complex illuminance including flares and reflection, making the text very hard to distinguish and segment. (g) is an example of text located in a very small area in the image. These challenging conditions make other methods fail or perform poorly, for example traditional MSER+SWT method totally fail in almost all images and the end-to-end CNN methods work unstable in (f)(i). It is worth noting that, for the method of Ours (Synth+D.T.), it produces good segmentation result in (c)(h), as the 2 examples contain texts in regular fonts, which is easy to synthesize for training. For the rest examples, as the texts are far from regular, its performance is naturally degraded due to the domain gap even after a domain transfer being applied. In contrast, our approach overcame the difficulties as stated above and produced relatively accurate and robust results. Please check the values below each image for detailed illustration in Fig. 6.

3) Fairness: As mentioned above, the 5 competitive methods are not originally designed to support polygon-level masks so that we did not feed polygon-level mask data during training in standard comparison. Considering this may result in potential unfairness, we conducted additional two experiments on the 5 competitive methods by feeding extra polygon-level masks just like in our method. To enable the polygon-level mask feeding, the two experiments were designed as follows:

a) Exp-1: We pre-trained the networks using polygon-level masks and fine-tuned them with pixel-level masks.
b) Exp-2: We added one branch at the middle layer on each network and fed these branches with polygon-level masks for dual-task training.

We also list all the evaluation data in Table III. Please note that the unmodified methods MSER [5]+SWT [6] and Ours

Fig. 6. Text segmentation results on TotalText (a ∼ e) and ICDAR-2013-WARP (f ∼ i) validation datasets, by various methods: MSER+SWT, CENet, SegNet, UNet, DeepLab-v3, Synth+D.T., SMANet and ours (Column 2 to 9). Synth+D.T. refers to training with the synthetic data and then applying domain transfer. Input and ground truth are shown in column 1 and 10. We list the precision, recall and F1 score under each resultant image, in black, blue and red respectively.

(Synth+D.T.) are also shown for unified comparison. As seen, the two kinds of adaption did not bring us better performance on the 5 competitive methods, as polygon-level masks may interfere or mislead the training. It results in a relative gain in F1 Top-1 value for our method, i.e. raising for nearly 10%.

C. Ablation Studies

1) Mutually Guided Network: We verified the effectiveness of the mutually guided network by an ablation study of involving Phase 2 training or not. The evaluations are compared in the row of Ours-α (Phase 2 training disabled) and Ours in Table IV(a). As seen, with Phase 2 training, the F1 score further rises by 1.3% and 0.4% for TotalText and ICDAR datasets respectively. This demonstrate the guidance takes effects for both branches. Note that, even for the case without Phase 2 training involved, i.e. Ours-α, our network achieves higher F1 scores than the 2nd and 3rd best state-of-the-art single-task CNN solutions (UNet, SMANet). It reveals that with more polygon-level mask involved in training, the shared encoder E could be finetuned as well. We also show 4 examples in Fig. 8

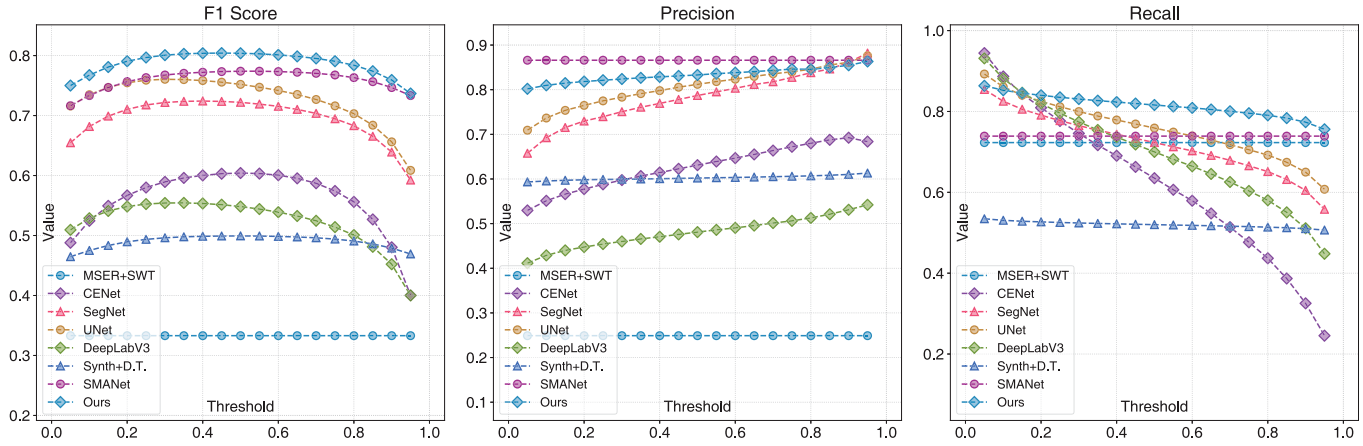


Fig. 7. Precision, Recall and F1 Score of our method in comparison with the state-of-the-art, on the TotalText dataset.

TABLE III

COMPARISON WITH POLYGON-LEVEL MASK FEEDING ENABLED IN TWO EXPERIMENTS (A) AND (B) FOR 5 COMPETITIVE METHODS CENET [17], SEGNET [29], UNET [30], DEEPLAB-V3 [31] AND SMANET [9]. THE SAME EVALUATION AND DATASETS ARE APPLIED AS IN TABLE II

	F1 (%)	F1 Top-1 (%)	Precision (%)	Recall (%)
MSER [5]+SWT [6]	33.3 / 43.8	1.67 / 3.00	24.9 / 39.4	72.3 / 69.6
CENet [17]	69.5 / 65.8	4.67 / 4.72	72.2 / 66.0	71.4 / 72.0
SegNet [29]	74.7 / 68.9	10.0 / 8.15	76.4 / 74.0	78.4 / 70.9
UNet [30]	77.1 / 70.6	16.7 / 13.3	80.2 / 73.1	78.8 / 75.4
DeepLab-v3 [31]	55.1 / 56.8	0.67 / 1.72	49.1 / 53.1	69.0 / 69.5
SMANet [9]	73.6 / 67.8	10.0 / 6.87	75.2 / 71.0	76.2 / 71.7
Ours (Synth+D.T.)	50.0 / 59.1	2.00 / 16.3	60.2 / 73.9	52.0 / 56.2
Ours	**80.5** / **74.5**	**54.3** / **45.9**	**83.3** / **79.0**	**81.6** / **77.0**

(a) Exp-1: Pre-trained using polygon-level mask and fine-tuned using pixel-level mask.

	F1 (%)	F1 Top-1 (%)	Precision (%)	Recall (%)
MSER [5]+SWT [6]	33.3 / 43.8	1.33 / 3.43	24.9 / 39.4	72.3 / 69.6
CENet [17]	63.8 / 65.9	2.00 / 1.72	63.9 / 67.4	68.9 / 71.0
SegNet [29]	72.2 / 69.2	9.67 / 3.86	75.1 / 73.4	74.1 / 72.2
UNet [30]	73.1 / 71.6	11.7 / 17.2	78.1 / 77.7	73.6 / 72.4
DeepLab-v3 [31]	53.9 / 58.2	1.67 / 8.58	46.5 / 52.6	70.4 / 72.4
SMANet [9]	72.0 / 67.0	4.67 / 3.86	72.6 / 71.8	75.9 / 69.1
Ours (Synth+D.T.)	50.0 / 59.1	2.00 / 13.7	60.2 / 73.9	52.0 / 56.2
Ours	**80.5** / **74.5**	**67.0** / **47.6**	**83.3** / **79.0**	**81.6** / **77.0**

(b) Exp-2: Adding one branch on the state-of-the-art methods to enable training using both polygon-level and pixel-level masks.

to illustrate the performance gain, where with Phase 2 training involved, the polygon-level masks provided positive guidance for the prediction of the pixel-level masks, removing some false-positive regions. This phenomenon is demonstrated for both datasets, TotalText and ICDAR.

2) Weakly-Supervised Losses: We also demonstrate the effectiveness of the introduced weakly supervised losses by gradually removing CRF and Subset losses. The evaluations are shown in the rows of Ours-γ ($\mathcal{L}_{CRF}$ removed) and Ours-β (both $\mathcal{L}_{CRF}$, $\mathcal{L}_{SB}$ removed) in Table IV(a). As seen, with these weakly supervised losses removed, the F1 score decreases by 0.2% to 0.6%. We also show an example in Fig. 3 to visualize the effects of $\mathcal{L}_{CRF}$, $\mathcal{L}_{SB}$. With only SoftIoU loss involved, there is an obvious false-positive region segmented in (b), and the region is restrained to a smaller size (e) by involving $\mathcal{L}_{SB}$ because the polygon-level segmented mask has only a tiny corresponding area (d). Finally, with $\mathcal{L}_{CRF}$ taking the affinity of neighboring pixels into account, the region further shrank so that the F1 score finally goes from 93.5% to 95.8% (c).

Fig. 8. Illustration of ablation study on the effectiveness of mutually guided network, for dataset ICDAR-2013-WARP (a)(b) and TotalText (c)(d).

TABLE IV

ABLATION STUDIES ON MUTUALLY-GUIDED NETWORK, WEAKLY SUPERVISED LOSSES (A) AND EFFECTIVENESS CAUSED BY THE PREDICTED MASK (B)

	M.G.	$\mathcal{L}_{SB}$	$\mathcal{L}_{CRF}$	F1 (%)	Precision (%)	Recall (%)
UNet [30]	✗	✗	✗	75.3 / 70.8	81.2 / 76.8	75.9 / 73.0
SMANet [9]	✗	✗	✗	77.5 / 71.3	86.6 / 74.4	73.9 / 73.8
Ours-α	✗	✓	✓	79.2 / 74.1	83.6 / 79.7	79.0 / 74.6
Ours-β	✓	✗	✗	79.9 / 74.1	84.7 / 79.7	79.7 / 74.9
Ours-γ	✓	✓	✗	80.3 / 74.4	82.3 / 77.6	82.1 / 77.7
Ours	✓	✓	✓	80.5 / 74.5	83.3 / 79.0	81.6 / 77.0

(a) Ablation studies on mutually-guided network (M.G.), subset loss and CRF loss

	ICDAR-03 (%)	ICDAR-13 (%)	SVT (%)
CRNN	89.40	86.70	80.80
CRNN-RM (Random Mask)	89.85	86.80	80.99
CRNN-PM (Predicted Mask)	91.35 (↑ 1.95)	87.68 (↑ 0.98)	82.84 (↑ 2.04)

	CUTE80 (%)	SVT-Perspective (%)
ASTER	79.50	78.50
ASTER-RM (Random Mask)	79.90	79.20
ASTER-PM (Predicted Mask)	83.70 (↑ 4.20)	83.90 (↑ 5.40)

(b) Improvement by the predicted mask for text recognition.

Even though in this example, the false-positive region is not thoughtfully removed due to the false prediction in the polygon-level mask, the performance gain is well illustrated.

Fig. 9. Working as an attention map to improve the text recognition accuracy, taking CRNN as an example.

3) Attention for Text Recognition: The predicted pixel-level mask enables lots of image editing operations such as changing the color of the text, inpainting or copying-and-pasting the text elsewhere etc. More importantly, we observe that it could serve as an attention map to facilitate the text recognition task. As introduced in Section III-C, we selected CRNN [1] as our network backbone, adding one additional pixel-level mask channel to the input layer. Following CRNN [1], we train the model on the synthetic text dataset Synth90k [32] and validate it on SVT [33], ICDAR-2003 [34] and ICDAR-2013 [8] respectively. To be specific, we initialize CRNN with an officially pre-trained model except the newly replaced input layer, which is further finetuned only instead. We conduct two variant experiments, i.e. feeding the mask channel with random mask (CRNN-RM) or our predicted mask (CRNN-PM). In both experiments, we finetune the CRNN variants to convergence. As a result shown in Table. IV(b), the performance of CRNN-RM is similar to that of CRNN, while CRNN-PM is about 1.95%, 0.98% and 2.04% accuracy gain over CRNN-RM, in ICDAR-03, ICDAR-13 and SVT datasets respectively. We also show 3 typical examples in Fig. 9, where with our predicted masks involved, the mis-recognized characters are corrected. For example, for the image of `"Marriott"`, the original CRNN network wrongly predicts the 1st character to `'n'`, and CRNN-RM mis-recognizes the 5th character as `'l'` due to the blurry pixels. However, with our predicted mask as assistant, the entire word is correctly recognized. The same observations could be also found for the other 2 examples. We also verify this phenomenon in another network ASTER [28]. Unlike CRNN, ASTER is trained by two synthetic text datasets Synth90k [32] and SynthText [35], and is validated by CUTE80 [36] and SVT-Perspective [37]. Detailed in Table IV(b), we demonstrate the performance gain achieves 4.20% and 5.40%, significantly outperforms the original networks.

V. Conclusion

We have presented a new data-driven method for pixel-level text segmentation from a single image. To achieve this goal, we designed a mutually guided dual-task neural network for joint segmentation of text in pixel level and polygon level. Our dual-task network contains a shared encoder but two decoders, for the two tasks separately. The two decoders work in a mutually guided manner, i.e. the output of either the pixel-level or polygon-level decoder also serves as a guidance to boost the segmentation performance for its counterpart. Furthermore, our network can be trained in a semi-supervised manner, i.e. we do not require both types of ground truth exist in one training data sample. It is achieved by a newly designed semi-supervised loss as proposed. We conducted extensive experiments to demonstrate the effectiveness of the mutually guided network structure and semi-supervised losses. Results show that our method outperforms the state-of-the-art in pixel-level text segmentation. We also demonstrate that with the predicted mask as an attention map, the text recognition accuracy could be further improved.

References

[1] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 11, pp. 2298–2304, Nov. 2017.

[2] M. Liao, B. Shi, X. Bai, X. Wang, and W. Liu, "Textboxes: A fast text detector with a single deep neural network," in *Proc. AAAI*, 2017, pp. 2–8.

[3] J. Ma *et al.*, "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Trans. Multimedia*, vol. 20, no. 11, pp. 3111–3122, Nov. 2018.

[4] X. Zhou *et al.*, "EAST: An efficient and accurate scene text detector," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5551–5560.

[5] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image Vis. Comput.*, vol. 22, no. 10, pp. 761–767, 2004.

[6] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2963–2970.

[7] C. K. Ch'ng and C. S. Chan, "Total-text: A comprehensive dataset for scene text detection and recognition," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Nov. 2017, pp. 935–942.

[8] D. Karatzas *et al.*, "ICDAR 2013 robust reading competition," in *Proc. 12th Int. Conf. Document Anal. Recognit.*, Aug. 2013, pp. 1484–1493.

[9] S. Bonechi, M. Bianchini, F. Scarselli, and P. Andreini, "Weak supervision for generating pixel–level annotations in scene text segmentation," *Pattern Recognit. Lett.*, vol. 138, pp. 1–7, Oct. 2020.

[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. NeurIPS*, 2015, pp. 91–99.

[11] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[12] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. ECCV*. Springer, 2016, pp. 21–37.

[13] C. Yao, X. Bai, N. Sang, X. Zhou, S. Zhou, and Z. Cao, "Scene text detection via holistic, multi-channel prediction," 2016, *arXiv:1606.09002*. [Online]. Available: http://arxiv.org/abs/1606.09002

[14] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.

[15] Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai, "Multi-oriented text detection with fully convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4159–4167.

[16] D. Deng, H. Liu, X. Li, and D. Cai, "PixelLink: Detecting scene text via instance segmentation," in *Proc. AAAI*, 2018, pp. 6773–6780.

[17] Z. Gu *et al.*, "Ce-Net: Context encoder network for 2D medical image segmentation," *IEEE Trans. Med. Imag.*, vol. 38, no. 10, pp. 2281–2292, Oct. 2019.

[18] A. Veit, T. Matera, L. Neumann, J. Matas, and S. Belongie, "COCO-text: Dataset and benchmark for text detection and recognition in natural images," 2016, *arXiv:1601.07140*. [Online]. Available: http://arxiv.org/abs/1601.07140

[19] N. Nayef *et al.*, "ICDAR2017 robust reading challenge on multi-lingual scene text detection and script identification–RRC-MLT," in *Proc. 14th IAPR Int. Conf. Document Anal. Recognit. (ICDAR)*, Nov. 2017, pp. 1454–1459.

[20] R. Cipolla, Y. Gal, and A. Kendall, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7482–7491.

[21] C. Wang, H. Huang, X. Han, and J. Wang, "Video inpainting by jointly learning temporal structure and spatial details," in *Proc. AAAI*, 2018, pp. 5232–5239.
[22] S. Cai *et al.*, "Disentangled image matting," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8819–8828.
[23] S. Caelles, K.-K. Maninis, J. Pont-Tuset, L. Leal-Taixe, D. Cremers, and L. Van Gool, "One-shot video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 221–230.
[24] P. Voigtlaender, Y. Chai, F. Schroff, H. Adam, B. Leibe, and L.-C. Chen, "FEELVOS: Fast end-to-end embedding learning for video object segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9481–9490.
[25] Z. Yang, Y. Wei, and Y. Yang, "Collaborative video object segmentation by foreground-background integration," in *Proc. ECCV*, Aug. 2020, pp. 332–348.
[26] J. Miao, Y. Wei, and Y. Yang, "Memory aggregation networks for efficient interactive video object segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10366–10375.
[27] M. Tang, F. Perazzi, A. Djelouah, I. Ben Ayed, C. Schroers, and Y. Boykov, "On regularized losses for weakly-supervised CNN segmentation," in *Proc. ECCV*, Sep. 2018, pp. 507–522.
[28] B. Shi, M. Yang, X. Wang, P. Lyu, C. Yao, and X. Bai, "ASTER: An attentional scene text recognizer with flexible rectification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 9, pp. 2035–2048, Sep. 2019.
[29] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.
[30] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. MICCAI*, 2015, pp. 234–241.
[31] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*. [Online]. Available: http://arxiv.org/abs/1706.05587
[32] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic data and artificial neural networks for natural scene text recognition," 2014, *arXiv:1406.2227*. [Online]. Available: http://arxiv.org/abs/1406.2227
[33] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 1457–1464.
[34] S. M. Lucas *et al.*, "ICDAR 2003 robust reading competitions: Entries, results, and future directions," *Int. J. Document Anal. Recognit.*, vol. 7, nos. 2–3, pp. 105–122, 2005.
[35] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2315–2324.
[36] A. Risnumawan, P. Shivakumara, C. S. Chan, and C. Tan, "A robust arbitrary text detection system for natural scene images," *Expert Syst. Appl.*, vol. 41, no. 18, pp. 8027–8048, 2014.
[37] T. Q. Phan, P. Shivakumara, S. Tian, and C. L. Tan, "Recognizing text with perspective distortion in natural scenes," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 569–576.

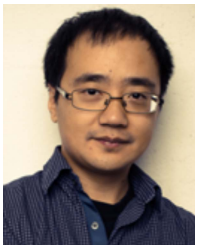
Chuan Wang received the B.Eng. degree from the University of Science and Technology of China in 2010 and the Ph.D. degree from The University of Hong Kong in 2015. He was a Computer Vision Staff Researcher at Lenovo Group Ltd., Hong Kong. He worked as a Visiting Scholar with the State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China, in 2010. He started his training program at Megvii Technology in 2018. His research interests include video analysis and computer vision.

Shan Zhao received the B.E. degree from Northwestern Polytechnical University, Xi'an, China, in 2016, and the M.S. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2019. She is currently an Assistant Researcher at Megvii Technology, Chengdu. Her research interests include computer vision and deep learning.

Li Zhu received the bachelor's degree from Sichuan Normal University in 2017 and the master's degree in electronic and communication engineering from the Wireless Technology Transmission Laboratory, Chongqing University of Posts and Telecommunications. She worked as an Intern at Megvii Technology, Chengdu, from March 2019 to March 2020. Her research interests include computer vision and image processing.

Kunming Luo received the B.Eng. and M.S. degrees from the University of Electronic Science and Technology of China, Chengdu, China, in 2016 and 2019, respectively. He is currently an Assistant Researcher at Megvii Technology, Chengdu. His research interests include computer vision and deep learning.

Yanwen Guo (Senior Member, IEEE) received the Ph.D. degree in applied mathematics from the State Key Lab of CAD & CG, Zhejiang University, China, in 2006. He was a Visiting Scholar with the Department of Electrical and Computer Engineering, University of Illinois at Urbana–Champaign, in 2013. He is currently an Associate Professor with the National Key Laboratory for Novel Software Technology, Department of Computer Science and Technology, Nanjing University, Jiangsu, China. His research interests include image and video processing, vision, and computer graphics.



Jue Wang (Senior Member, IEEE) received the B.E. and M.Sc. degrees from the Department of Automation, Tsinghua University, Beijing, China, in 2000 and 2003, respectively, and the Ph.D. degree in electrical engineering from the University of Washington, Seattle, WA, USA, in 2007. He is currently the Senior Director of Megvii Technology. Before that, he has been a Principle Research Scientist at Adobe Research for nine years. His research interests include image and video processing and computational photography. He is a Senior Member of the ACM. He received the Microsoft Research Fellowship and Yang Research Award from the University of Washington in 2006.

Shuaicheng Liu (Member, IEEE) received the B.E. degree from Sichuan University, Chengdu, China, in 2008, and the M.Sc. and Ph.D. degrees from the National University of Singapore, Singapore, in 2010 and 2014, respectively. In 2014, he joined the University of Electronic Science and Technology of China. He is currently an Associate Professor with the School of Information and Communication Engineering, Institute of Image Processing, Chengdu. His research interests include computer vision and computer graphics.