Background-Insensitive Scene Text Recognition with Text Semantic Segmentation

Liang Zhao, Zhenyao Wu, Xinyi Wu, Greg Wilsbacher, and Song Wang†

University of South Carolina, Columbia SC 29201, USA
{lz4, zhenyao, xinyiw}@email.sc.edu, gregw@mailbox.sc.edu,
songwang@cec.sc.edu

Abstract. Scene Text Recognition (STR) has many important applications in computer vision. Complex backgrounds continue to be a big challenge for STR because they interfere with text feature extraction. Many existing methods use attentional regions, bounding boxes or polygons to reduce such interference. However, the text regions located by these methods still contain much undesirable background interference. In this paper, we propose a Background-Insensitive approach BINet by explicitly leveraging the text Semantic Segmentation (SSN) to extract texts more accurately. SSN is trained on a set of existing segmentation data, whose volume is only 0.03% of STR training data. This prevents the large-scale pixel-level annotations of the STR training data. To effectively utilize the segmentation cues, we design new segmentation refinement and embedding blocks for refining text-masks and reinforcing visual features. Additionally, we propose an efficient pipeline that utilizes Synthetic Initialization (SI) for STR models trained only on real data (1.7% of STR training data), instead of on both synthetic and real data from scratch. Experiments show that the proposed method can recognize text from complex backgrounds more effectively, achieving state-of-the-art performance on several public datasets.

Keywords: Scene text recognition; Semantic segmentation

1 Introduction

Scene Text Recognition (STR) aims at accurately recognizing irregular and incidental texts in complicated scenes and it has wide applications in video information retrieval [63], criminal investigation [1], robotic intelligence [45], and autonomous driving [30]. STR is still a very challenging task in computer vision, due to large variation of the text color, font and size, as well as the possible complex background where the text is located. In real world, the complexity of scene-text background comes from many factors, such as camera motion, scene change, low lighting, etc. In extracting text features for STR, any interference of background information may negatively affect the recognition performance. In this paper, we study BINet that is less sensitive to the complex background.

† Corresponding author.

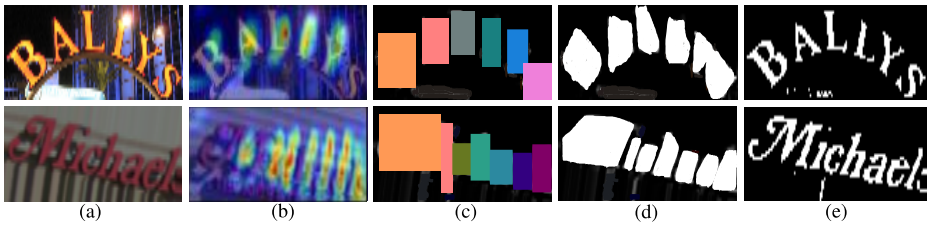

Fig. 1. An illustration of strategies focusing on text features for Scene Text Recognition (STR): (a) whole input images, (b) attention map [19,84], (c) bounding box [48], (d) conventional text segmentation [40], and (e) text semantic segmentation (ours).

To mitigate the background interference for STR, different strategies have been explored to derive only-text-related feature representation. These mainly can be categorized into attention-based [85,49,84,19], bounding-box-based [48], and conventional-segmentation-based [43,40,39] by focusing on different areas around the text, as illustrated in items (b) through (d) of Fig.1. However, none of these strategies can optimally exclude the background interference. For example, the attention-based strategy could wrongly entangle neighboring characters or miss some distinctive characters, as shown in Fig.1(b). The bounding box-based and conventional-segmentation-based strategies may still include unrelated background regions as part of the text, as shown in Fig.1(c-d). Clearly, a strategy that can more accurately separate out text from background may further facilitate accurate text feature extraction and STR.

To achieve this goal, in this paper we propose to conduct *text semantic segmentation* (SSN) for enhancing STR. Following general-purpose image semantic segmentation [52,57], the text semantic segmentation here aims to classify each pixel of text image as either text or background and therefore, partitions the image into two semantic segments, as shown in Fig.1(e). Note that, text semantic segmentation is different from conventional text segmentation. The latter segments out a compact text region which may still contain part of the background, e.g., the hollow regions in character 'B' in Fig.1(d). On the contrary, text semantic segmentation fully separates the text and background at pixel level, and they can provide more accurate text features for recognition.

However, the SSN requires accurate pixel-level annotations, which are laborious to obtain. Embedding a pretrained network instead of integrated training for assisting specific tasks has been widely used in many general image segmentation works [6,66,34,18]. However, few studies [39,17] design the text segmentation (usually by generating pseudo mask annotations), let alone any independent text segmentation, and the previous recognition improvements are limited [17]. In this work, we leverage the explicit text semantic segmentation with limited knowledge to our advantage in STR without labeling mask annotations for huge STR training data. Here, the SSN is trained on existing real data, whose volume is only 0.03% compared to the STR training data.

Furthermore, to address the problems of data deficiencies and domain differences between SSN and STR, and expand SSN's generalization capability in STR task, we further design two blocks of segmentation refinement and embedding that steer the knowledge from SSN to STR. Recent state-of-the-art STR models typically design the pixel-wise fusion [79] or transformer [84] in utilizing the sequential, attentional or positional information. The guiding methods are not sufficiently studied and the transformer is extremely expensive in computing resources. In natural language processing (NLP) community, the large-scale pretrained model [56] is modified for downstream tasks [12] by CNNs or distribution losses [67]. Following this insight, we develop efficient networks to utilize the limited prior knowledge from SSN and facilitate the visual cues in STR.

Typically, the state-of-the-art STR models are required to be end-to-end trained on the synthetic dataset, and further training on real data can boost the performance. However, the whole training process is cumbersome and takes up to 672 GPU hours [19]. Inspired by the wide practice that ResNet based models are initialized by ImageNet weights, we propose to use the backbone weights pretrained on synthetic data for general-purpose STR model initialization denoted as Synthetic Initialization (SI). It can be adopted by any STR models without training on synthetic data. Experiments show that our method, trained after 5 hours with 1 GPU card, can achieve the state-of-the-art STR performance on several public evaluation benchmarks.

In summary, our contribution and achievement are described below:

1. We propose a novel and efficient BINet that leverage text semantic segmentation (SSN) for enhancing scene text recognition without large-scale pixel-level annotations. It also gets rid of generating pseudo segmentation labels for self-supervised, semi-supervised, or weakly-supervised training.
2. The text segmentation refinement and embedding modules are specially designed for conditioning background-insensitive features, which efficiently steer the limited knowledge from SSN to STR.
3. We design a new pipeline with Synthetic Initialization (SI) for STR, replacing conventional expensive end-to-end training on synthetic data. The model overcomes great performance degradation when trained only on real data.
4. The proposed method achieves new state-of-the-art performances on various widely-used datasets.

2 Related Work

2.1 Scene Text Recognition

Attention-based Text Recognition. Initially enlightened by contextual inference in natural language processing (NLP) community, the attention-based STR models are equipped with the RNN [36,81], canonical attention BLSTM [60,13,42,86], bidirectional attention BLSTM [61,11], two-dimension attention BLSTM [38,80,75,4] and Transformer decoders [93,7,55,19,3], instead of Convolutional Neural Networks (CNN) based classifier [20], in an effort to boost the

language expression in STR. The language models utilize one or two dimensional visual features to consider the character relationships from unidirectional or bidirectional way with attention. Later, the attention is expanded to encoders to improve feature representations [76,41,87,25,55,19]. These methods sufficiently explore the linguistic information and feature representations in lack of character details, but the attention maps might easily miss some small characters with arbitrary locations, or entangle with neighbors to generate wrong predictions.

Box-based Text Recognition. Early studies with limited bounding boxes or polygon ground truths mainly adopted segmentation methods for the localization and detection of characters or words. Neumann *et al.* [51] proposed the connected components [24] to binarize the image as coarse segmentation for text recognition. Then a discriminating clustering algorithm [82] and a hybrid HMM Maxout [2] technique were used to capture character substructures from regions. But they fail to separate contiguous characters and integrate broken strokes. The CNN-based supervised methods [29,74,48] usually compute a text saliency map by using the character classifier and then generate character or word bounding boxes. They are usually boosted by strengthened recognizer [47,10,44], or rectification modules [42,40,90,82]. Without well-designed module integration and sufficient pixel-level annotations, the segmentation module is barely learned and evaluated, mainly for text detection, instead of directly for text recognition.

Conventional Segmentation-based Text Recognition. To handle irregular scene text recognition, Jaderberg *et al.* [28] proposed to recognize the characters or words within the detected regions by using at least 9 million images and a 90k word dictionary. The problem is formulated indirectly by embedding text strings into subspace vectors, which could calculate the nearest neighbor prediction results. Following this work, many segmentation-region-based methods [64,26,60,36,43] were proposed for STR. Recently, Liu *et al.* [42] employed a character encoder to rectify the accurate local character region, but it might contain neighboring characters. Liao *et al.* [40] utilized a Fully Convolutional Network (FCN) to estimate character polygons under the supervision of the processed bounding boxes.But such polygons and ground truths still contain background parts. Zhang *et al.* [90] utilized attention to adaptively increase the tightness of the local character regions, and Wan *et al.* [70] further rectified spurious attention by complementing the character regions with position and order attention. These methods applied multiple modules to refine segmentation regions for predicting more accurate recognition probabilities of each class. As mentioned earlier, these conventional text segmentation methods may still mix part of background into text segments. In this paper, we propose to leverage text semantic segmentation into STR by more accurately separating the text and background in the image.

2.2 Text Semantic Segmentation

Aiming at predicting all the text pixels, the text semantic segmentation has been studied in several text-related tasks, e.g., text style transfer [33] and scene text

removal [22]. These methods directly implement the supervised semantic segmentation and transfer text styles in small datasets of the tasks, in incorporating with generative adversarial networks (GANs). For the Scene Text Recognition (STR), the closest work is from Luo *et al.* [46] that uses the GANs to separate text content from the backgrounds. However, it fails to generate characters on complex backgrounds due to the mode-dropping phenomenon [8] and the lingering gap [92] of GANs trained on the supervised synthesized samples. The power of generators highly depends on the synthesized character style samples and then the capacity of synthesis engines. Several studies [83,40] purposely generate polygons as pseudo ground truths on self-supervised or weakly-supervised training. To overcome the limitations of synthesized training samples and pseudo ground truths, we independently pretrain text semantic segmentation network (SSN) only on real data, which is only 0.03% of STR training data. The model does not need to generate pseudo ground truths or synthesize reference samples. We also well design text segmentation refinement and embedding modules for steering knowledge from SSN to STR.

2.3 Training Strategy

Most state-of-the-art STR models are trained end-to-end from scratch on synthetic data [27,23]. Some works [39,38,41] continually train extra real data to reduce domain drifts. Recently, Beak *et al.* [5] demonstrate that training only on real data (1.7% of synthetic data) causes great performance degradation, while training on synthetic and real data together can definitely boost the recognition. Inspired by NLP works [16,35] that utilize general pretrained models for diverse downstream tasks, we further propose the Synthetic Initialization (SI) that is a backbone weights solely pretrained on synthetic data. Note that the backbone refers to ResNet (and an encode unit) that is generally used as feature extractor in STR models [55,19,84]. Thus, the pretrained backbone weights can be widely used to initialize feature extractor in any STR models. Only the specific design of each method needs to be trained, instead of the whole model trained on synthetic data from scratch. We explore whether it can overcome the performance degradation caused by real data deficiency (1.7% of synthetic data).

3 STR with Text Segmentation

3.1 Overview

The proposed BINet is shown in Fig.2. It consists of a text semantic segmentation network (SSN), a feature extractor, a segmentation refinement module, a segmentation embedding module and a transformer-based language decoder.

Following recent works [55,19,84], we use the combination of ResNet $\mathcal{R}$ and Transformer units $\mathcal{T}$ as the feature extractor. Given an image I, the feature map F is generated by $F = \mathcal{T}(\mathcal{R}(I))$ with the shape of $H \times W \times C$, where H and W are the height and width of the features, and C is the number of channels (we

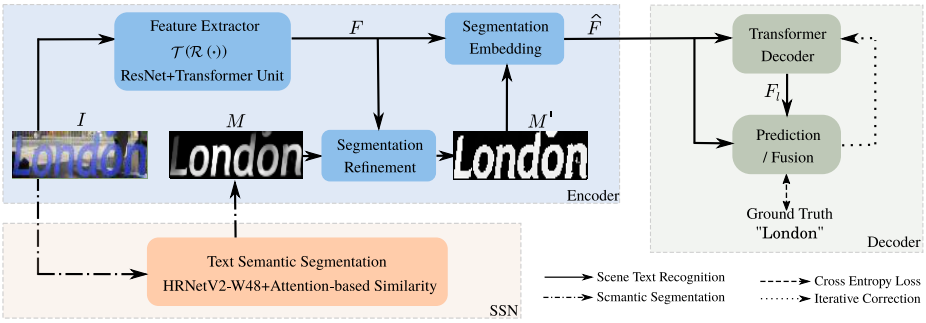

Fig. 2. The framework of the proposed BINet. The input image I and text semantic segmentation M from SSN are fed into the network. The extracted feature F is used to refine the segmentation map M and obtain M'. Then M' is embedded back to F for visual feature $\hat{F}$. The decoder generates language feature F_l with 3 iterations of correction. The text predictions from the combination of features are supervised by the text ground truths via a cross-entropy recognition loss.

set $C = 512$ in this paper). At the same time, the image I is also fed into the text semantic segmentation network which produces a semantic segmentation M. The initial feature F and segmentation map M are fed into the proposed segmentation refinement and the segmentation embedding blocks to obtain the segmentation-embedded feature $\hat{F}$. Later, a language decoder is employed to obtain the final prediction, which is composed of multi-layer transformers [68] with the iterative correction strategy. Specifically, the language decoder is pre-trained on an unlabeled natural language processing dataset following Fang *et al.* [19] to learn the linguistic knowledge F_l and corresponding language vectors V_d^i $(i = 1, 2, 3)$ from $\hat{F}$. Following previous works [84], $\hat{F}$ and F_l are fused recurrently to generate the refined probability distributions as final text prediction.

3.2 Semantic Segmentation Network

Text semantic segmentation can indicate accurate characters from background. To achieve background-insensitive text feature representation, we propose to explicitly model the text semantic segmentation network (SSN) on real data. The SSN is firstly pre-trained on two real-world text datasets [14,78] with pixel-level annotations. The total data here is around 4,000 images, accounting for only 0.03% of STR 16 million training data. The SSN is equipped with HRNetV2-W48 [72] as the backbone, and utilizes an attention module [78] as head to generate the final semantic segmentation results. Specifically, the image I is fed into the backbone to get the feature map x_f and initial semantic segmentation x_{seg}. Then taking x_{seg}, x_f and I as inputs, the attention module generates the segmentation result M, which is a soft map, as shown in Fig.2.

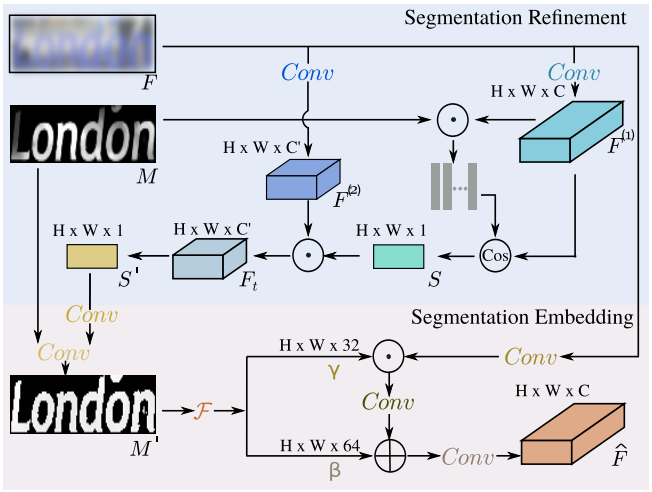

Fig. 3. The network of segmentation refinement and embedding blocks in BINet. Different colors of *Conv* represents different series of convolution layers, the *Cos* indicates cosine operation, and the $\odot$ and $+$ mean pixel-wise multiplication and addition, as described in Sections 3.3 and 3.4.

3.3 Segmentation Refinement

Due to limited number of images with pixel-level annotations used for training the SSN and the problem of domain shifts in STR datasets, the quality of text semantic segmentation results produced by the SSN are possibly unsatisfactory for STR. To address these issues, we propose to refine the segmentation M by utilizing the image features from STR training data, which compensate the details of segmentation maps, as shown in Fig.3. Firstly, the features F is fed into a two-dimensional convolution layer $Conv$ to get global features $F^{(1)}$:

$$F^{(1)} = Conv(F) \in \mathbb{R}^{H \times W \times C}, \tag{1}$$

where the number of channels C is 512. Then we extract the representative vector $V_c \in \mathbb{R}^C$ [89] by pixel-wise multiplying $F^{(1)}$ and the segmentation M, and then averaging over pixels within M,

$$V_c = \frac{\sum_{m=1,n=1}^{W,H} F^{(1)}_{m,n,c} \odot M_{m,n}}{\sum_{m=1,n=1}^{W,H} M_{m,n}}. \tag{2}$$

The vector V_c is related to texts by removing the background interference. To find features similar to the vector V_c and compensate the segmentation map M, we use the cosine distance $CosSim$ to calculate the similarity map $S \in \mathbb{R}^{H \times W \times 1}$ between the representative vector V_c and each pixel of the features $F^{(1)}$:

$$S = CosSim(F^{(1)}, V_c). \tag{3}$$

The map is applied to optimize the semantic feature $F^{(2)} \in \mathbb{R}^{H\times W\times C'}$ which is constructed from the last two layers of ResNet $\mathcal{R}$ in the same way as $F^{(1)}$. The number of channels C' for $F^{(2)}$ is 128. $F^{(2)}$ is used to get the accurate text features F_t, which is then fed into a classification layer cls to obtain the corresponding semantic segmentation $S' \in \mathbb{R}^{H\times W\times 1}$,

$$S' = cls(F_t) = cls(S \odot F^{(2)}). \tag{4}$$

Finally, we propose to reinforce the segmentation M by the fusion with the optimized semantic features which contains features similar to the text and restores certain details of the segmentation map. The refined segmentation $M' \in \mathbb{R}^{H\times W\times 1}$ is obtained by two convolutions $Conv$ with a residual fusion.

$$M' = Conv(M + Conv(S')). \tag{5}$$

To encourage the refinement block to produce segmentation maps with high confidence, we design a segmentation regularization term $\mathcal{L}_m$ as:

$$\mathcal{L}_m = \frac{1}{H\times W}\sum(\sigma - |M' - \sigma|), \tag{6}$$

where $H \times W$ is the number of pixels in the segmentation map M', and σ is a threshold of the convergence.

3.4 Segmentation Embedding

The affine transformation could learn to recover high-quality texture based on semantic segmentation maps [77]. Thus, we develop the segmentation embedding module to embed the segmentation M' into image features F to indicate text details for STR. Specifically, the refined segmentation M' is modeled into two transformation parameters γ and β by a mapping function $\mathcal{F}$ [77]:

$$\gamma, \beta = \mathcal{F}(M'), \tag{7}$$

where $\mathcal{F}$ contains two branches of convolutional layers that are optimized with our BINet. Then, the learned parameters are adopted into the features as:

$$\hat{F} = F \odot \gamma + \beta, \tag{8}$$

where $\odot$ is the element-wise multiplication. With the learned conditions, the feature maps F are guided by the refined text segmentation for text recognition.

3.5 Optimization

For the individual training of text recognition in our proposed BINet, the objectives consist of text recognition losses and the segmentation regularization term. The total loss function is defined as:

$$\mathcal{L} = \lambda_e\mathcal{L}_{CE}(V_e, GT) + \frac{\lambda_d}{N}\sum_{i=1}^{N}\mathcal{L}_{CE}(V_d^i, GT) + \frac{\lambda_f}{N}\sum_{i=1}^{N}\mathcal{L}_{CE}(V_f^i, GT) + \lambda_m\mathcal{L}_m, \tag{9}$$

where $\mathcal{L}_{CE}$ denotes the cross-entropy recognition losses from the predictions of the encoder vectors V_e, decoder vectors V_d, and the fusion vectors V_f following Fang *et al.* [19] with $N = 3$ iterations for the decoder, GT is the ground truth text, and $\mathcal{L}_m$ is our proposed regularization loss. The balanced weights λ_e, λ_d, λ_f and λ_m are set to 1.0, 1.0, 1.0, and 1.0, respectively.

4 Experiments

4.1 Datasets and Implementation Details

Most STR models are trained on the synthetic text datasets referring to the MJSynth (MJ) [27,26] and SynthText (ST) [23], which totally have more than 16 million images.

Unlike previous works trained on synthetic datasets and real data, we just train the STR model on real data [5] that contains 276K images, which is 1.7% of synthetic data. It contains a group of real datasets. Street View Text (SVT) [73] consists of 257 training and 647 testing street scene text images. IIIT5k-Words dataset (IIIT) [47] is collected from Google images which includes 2,000 training and 3,000 testing images. ICDAR2013 (IC13) is built in the ICDAR 2013 Robust Reading Competition [32] with 848 images for training and 1,015 images for testing. ICDAR2015 (IC15) [31], consisting of 4,468 training and 2,077 testing images, has more irregular texts with perspective and blur attributes. COCO-Text (COCO) [69] includes occluded and low-resolution texts of around 39K images. Further, RCTW [62], Uber-Text (Uber) [91], ArT [15], LSVT [65], MLT19 [50], and ReCTS [88] are also included with 8,186, 92K, 29K, 34K, 46K, and 23K images, respectively. The final real data is accounted to 276K images in total. In addition to the above, for evaluation we also use SVT Perspective (SVTP) [53] which consists of 645 street view perspective-text images, and CUTE [58] which is a dataset of 288 curved texts. The evaluation metric is the widely used word-level recognition accuracy on the benchmark datasets.

The experiment settings are described below. The model is trained on batch size of 64 on one 16G NVIDIA v100 graphic card. The initial learning rate is set to $1e^{-4}$ and then decayed to tenth of it for last four epochs in total 10 epochs, with Adam optimizer. The input images are processed with data augmentation including random rotation, affine transformation, perspective distortion, and color editing [19]. All inputs are resized to 32×128 for training and testing.

4.2 Comparing with State-of-the-Art Methods

We compare our BINet with the state-of-the-art methods in Table 1. Note that it is hard to fairly compare with different methods due to various pre-processing, rectification, training data, training strategies, etc. It is also not possible to reproduce most previous works with the same configuration in this paper due to limited available codes. However, training on synthetic data is a great advantage [5] for existing methods, compared with our data deficiency.

Table 1. Accuracy (%) comparison of STR models on different training strategies and six benchmark evaluation datasets. MJ, ST, Real and Real' represent MJSynth, SynthText, and two different unions of real datasets, with 9 million, 7 million, 48K and 276K images, respectively. The "Total" means evaluation on the union of all testing datasets. The top accuracy is in bold for each evaluation dataset on two versions of benchmarks. Most works are evaluated on the original version of IC13 of 1015 images and IC15 of 2,077 images, while recent studies are evaluated on the developed version of IC13 of 857 images and IC15 of 1,811 images, denoted with "*".

Method	Training Datasets	Testing Datasets						
		IIIT	SVT	IC13	IC15	SVTP	CUTE	Total
ESIR[86]	MJ+ST	93.3	90.2	91.3	76.9	79.6	83.3	86.8
DAN[76]	MJ+ST	94.3	89.2	93.9	74.5	80.0	84.4	86.9
ASTER[61]	MJ+ST	93.4	89.5	91.8	76.1	78.5	79.5	86.4
SE-ASTER[55]	MJ+ST	93.8	89.6	92.8	80.0	81.4	83.6	88.2
ScRN[80]	MJ+ST	94.4	88.9	93.9	78.7	80.8	87.5	88.2
PlugNet[49]	MJ+ST	94.4	92.3	95.0	82.2	84.3	85.0	89.8
Bhunia *et al.* [9]	MJ+ST	95.2	92.2	95.5	84.0	85.7	89.7	90.9
Li *et al.* [38]	MJ+ST+Real	91.5	84.5	91.0	69.2	76.4	83.3	83.2
Hu *et al.* [25]	MJ+ST+Real	95.8	92.9	94.4	79.5	85.7	92.2	90.0
TextScanner[70]	MJ+ST+Real	95.7	92.7	94.9	83.5	84.8	91.6	91.2
RobustScanner[85]	MJ+ST+Real	95.4	89.3	94.1	79.2	82.9	92.4	89.2
PIMNet[54]	MJ+ST+Real	96.7	94.7	95.4	**85.9**	88.2	92.7	92.5
CRNN[59]	MJ+ST	84.3	78.9	88.8	61.5	64.8	61.3	75.8
CRNN[5]	MJ+ST+Real'	89.8	84.3	90.9	73.1	74.6	82.3	83.4
TRBA[4]	MJ+ST	92.1	88.9	93.1	74.7	79.5	78.2	85.7
TRBA[5]	MJ+ST+Real'	95.2	92.0	94.7	81.2	84.6	88.7	90.0
Ours	SI+Real'	**97.3**	**96.4**	**96.7**	85.0	**89.9**	**95.8**	**93.1**
SRN[84]*	MJ+ST	94.8	91.5	95.5	82.7	85.1	87.8	90.4
PREN2D[79]*	MJ+ST	95.6	94.0	96.4	83.0	87.6	91.7	91.5
PIMNet[54]	MJ+ST+Real	96.7	94.7	96.6	88.7	88.2	92.7	93.5
ABINet[19]* (original)	SI+MJ+ST	96.2	93.5	**97.4**	86.0	89.3	89.2	92.6
ABINet[19]* (reproduce)	SI+Real'	97.0	94.9	96.1	88.2	88.5	94.4	93.7
Ours*	SI+Real'	**97.3**	**96.4**	96.8	**89.2**	**89.9**	**95.8**	**94.4**

Specifically, there are four training strategies indicated as $MJ + ST + Real$, $MJ + ST + Real'$, $SI + MJ + ST$, and $SI + Real'$, compared with conventional methods on synthetic data $MJ+ST$, as shown in Table 2. The $MJ+ST+Real$ is the STR model trained on two synthetic datasets [27,23] and several real datasets (IIIT5K, SVT, IC03, IC13, IC15, COCO) [38,25,70,85,54]. The $MJ+ST+Real'$ means that the STR model is trained on both synthetic datasets and another version of real data (IIIT5K, SVT, IC13, IC15, COCO, RCTW, Uber-Text, ArT, MLT19, ReCTS) [5]. The SI refers to use synthetic initialization. It is the backbone weights appeared in ABINet [19] and treated as the pre-training of feature extractor on synthetic data. The whole model will be trained on synthetic data, which is denoted as $SI + MJ + ST$. We figure out that the weights from the backbone or feature extractor can be widely used as model initialization for downstream recognition task, instead of training whole STR model on synthetic data from scratch. We remove the positional encoding module designed by the

Table 2. Comparison of different training strategies. For different stages, MJ, ST, Real and Real' represent MJSynth, SynthText, and two different unions of real datasets. The SI denotes separated pretraining on synthetic data. The training time is calculated as the sum of "Train" and "Finetune".

Training Strategy	Number of Images	Pretrain	Train	Finetune	Training Time
MJ+ST	16 Million	-	MJ+ST	-	$>$ 1 week
MJ+ST+Real	16 Million + 48K	-	MJ+ST	Real	$>$ 1 week
MJ+ST+Real'	16 Million +276K	-	MJ+ST	Real'	$>$ 1 week
SI+MJ+ST	16 Million	SI	MJ+ST	-	$>$ 1 week
SI+Real'	276K	SI	Real'	-	$\sim$ 5 hours

original work and only keep the backbone weights for the feature extractor. To verify this idea, we train our method only on real data denoted as $SI + Real'$.

Corresponding to the original version of benchmarks (with IC13 of 1015 images and IC15 of 2077 images), most works are trained on synthetic data and real data *Real*. Baek *et al.* [5] constructed another real data *Real'* and reproduced two typical STR models on the synthetic data and *Real'*, i.e., CRNN and TRBA, with additional ROTNet [21] and unlabeled data [37], as shown after corresponding original results. Our method outperforms most previous models with remarkable margins especially on complex and irregular datasets. Specifically, the performance is improved by 2.2%, 4.8%, 2.1%, 4.7%, 6.3%, 8.0% and 3.4% on IIIT, SVT, IC13, IC15, SVTP, CUTE datasets and the total evaluation, respectively, comparing with the latest TRBA [5] trained on the synthetic and real data as well as extra unlabeled datasets. The results are boosted by 0.6%, 1.8%, 1.4%, -1.0%, 1.9%, 3.3% and 0.6% on above datasets and the total evaluation, comparing with the previous SOTA PIMNet [54]. Note that synthetic data of 16 million is extremely large compared to both kinds of real data, which are accounted for 0.3% and 1.7%, respectively. Baek *et al.* [5] demonstrate that training on synthetic data is an advantage for STR models and can beat any models training only on real data. With the novel design of SI, training *only* on real data now can compete the synthetic training of STR models. Besides, it could finish training in several hours that is less than 0.7% of the time for conventional methods training on synthetic datasets.

Several recent works are evaluated on a developed version of IC13 and IC15, which contains 857 and 1,811 images, respectively. Results are shown in the bottom half of the Table 1. For the regular text datasets IIIT, SVT and IC13, there are the 1.1%, 3.1% and -0.7% improvements, respectively, compared to the original ABINet [19]. For the challenging irregular text datasets IC15, SVTP, and CUTE, our model achieves the best results by increasing of 3.7%, 0.7%, and 7.4% performance compared to the same method. The total evaluation result is boosted by around 2.0%. For fair comparison, we reproduce the results of ABINet [19] trained on real data (while other codes are not fully provided for reproducing the results). The reproduced results trained with the same configuration on real data are improved in total evaluation from 92.6% to 93.7%. Training the

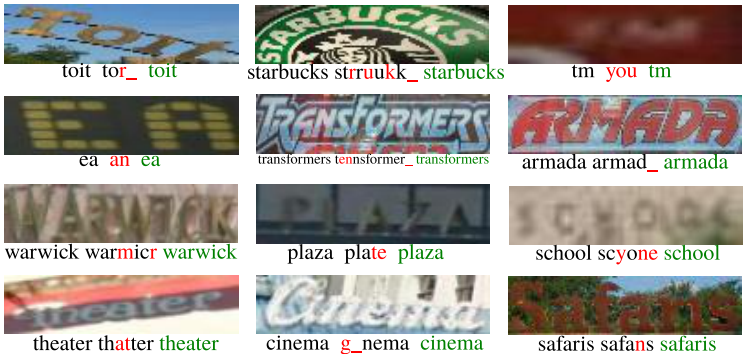

Fig. 4. Qualitative challenging examples. Under each image, the left text is the ground truth; the middle one is the prediction from the SOTA work [19] while the red color indicates the wrongly predicted or missed characters; and the right one in green color is the prediction from our model.

whole model solely on real data previously causes great performance degradation [5] due to its 1.7% volume of synthetic data. Our results indicate that solely training on real data with SI can beat the end-to-end training on synthetic data in total accuracy. Moreover, BINet shows impressive superiority on the challenging datasets (IC15, SVTP, and CUTE) that contains various kinds of background-interference scene text images (as shown in Fig.4), which previous works are generally short at confidently recognizing them.

The Fig.4 shows certain qualitative results. For the perspective, curved and blur texts entangled with background shapes in the first row, the attention-based method is vulnerable in missing or wrongly recognizing characters. For the styled texts in the second row, the background interference with unexpected breaks cause the recognition difficulties. When the background is extremely similar to the texts, the interference is much more irregular in STR as shown in the third row of Fig.4. In the case of complicated background changing in the forth row, the character shapes are much more important in recognizing the characters. Our model handles the challenging cases more robust than the previous method.

4.3 Ablation Study

SSN Strategy and Modules. To verify the effectiveness of main components in our BINet, we design two different training strategies for the text semantic segmentation (SSN), i.e., conventional jointed training versus pretrained SSN, together with or without the segmentation refinement (SR) and segmentation embedding (SE) blocks described in Section 3.3 and 3.4, respectively. Experiments are evaluated on both the original and developed version of benchmarks as described in Section 4.2.

As shown in Table 3, the performance of jointed segmentation is inferior to that of explicitly pretrained strategy. More specifically, by using pretrained strat-

Table 3. Accuracy (%) comparison on different components of BINet. For the SSN, the "J" represents the jointed training while the "P" represents the pretrained strategy. The developed version of benchmarks is denoted with "*".

SSN	SR	SE	IIIT	SVT	SVTP	CUTE	IC13	IC15	IC13*	IC15*	Total	Total*
J	-	-	96.2	95.1	88.5	94.4	95.3	83.9	95.8	87.9	91.9	93.2
J	-	✓	96.2	94.3	87.3	92.4	94.3	82.9	96.4	89.4	91.3	93.4
P	-	-	96.7	95.1	88.4	94.4	95.9	84.3	96.6	88.6	92.3	93.7
P	-	✓	96.9	95.2.	89.0	94.4	96.2	84.6	96.7	88.6	92.6	93.8
P	✓	✓	97.3	96.4	89.9	95.8	96.7	85.0	96.8	89.2	93.1	94.4

egy instead of jointed training, the recognition accuracy increases from 91.9% to 92.3% without SE, and from 91.3% to 92.6% with SE, for the original version of benchmarks. The jointed training might cause insufficient learning of the segmentation and deficient communication with the recognition, but pretrained SSN could boost the learning of text semantic segmentation and reduce the negative effect for recognition. That is, the text semantic segmentation is not severed as both for improving segmentation and recognition performance at the same time, which reduces the unexpected noises and interruptions from models [19,6]. We equip SR and SE blocks based on the pretrained SSN and the performance further increases 0.5%. Compared with SSN and SE with higher improvements of 0.9% and 1.5% on IC13 and SVTP, SR improves more accuracy of 1.2% and 1.4% on SVT and CUTE, respectively, for the original version of benchmarks. The marked improvements on each of the benchmarks demonstrate the effectiveness of each component in our BINet.

Segmentation Embedding. We also try different segmentation embedding strategy to better utilize the semantic segmentation map for STR, including stacking images with masks [71] (Concat), adding feature maps as residual fusion (Add), attention-based multiplying [68] (Multiply), and our proposed segmentation embedding (SE). They are corresponding to each row of Table 4, respectively. The results show that our designed SE block works more effectively than other competitors for STR.

Segmentation Refinement. We further explore the effectiveness of segmentation refinement module. Due to the domain gap between the datasets for semantic segmentation and the datasets for STR, directly applying the pretrained model to generate the segmentation map for STR dataset might get inaccurate results. We can see that some initial text segmentation in the middle column of Fig.5 are not satisfactory. After refinement, the segmentation results are improved as shown in the right column of Fig.5. It is obvious that some missing strokes and components of segmentation are compensated to exhibit more distinctive features, which is especially important in the recognition of similar characters. For example, the initially segmented first and last characters in the

Table 4. Accuracy (%) comparison on different embedding methods. The developed version of benchmarks is denoted with "*".

Level	Strategy	IIIT	SVT	SVTP	CUTE	IC13	IC15	IC13*	IC15*	Total	Total*
Image	Concat	91.2	89.2	80.6	83.3	92.2	74.6	93.1	79.4	85.5	87.0
Feature	Add	96.6	94.7	90.2	94.8	96.0	83.7	96.6	88.0	92.3	93.6
Feature	Multiply	96.7	95.1	88.4	94.4	95.9	84.3	96.6	88.6	92.3	93.7
Feature	Ours	96.9	95.2	89.0	94.4	96.2	84.6	96.7	88.6	92.6	93.8

Fig. 5. Samples of the initial segmentation map (middle column) and the refined segmentation map (right column).

bottom row look like "c" and "i", respectively. After the refinement, it is more easier to identify the characters "F" and "t", respectively.

5 Conclusion

In STR models, the background interference always causes ineffective feature representations for recognition. In this paper, we proposed an effective framework BINet that leverages the text semantic segmentation to STR by novel segmentation refinement and segmentation embedding blocks. We also design an efficient pipeline for training the model only on real data with synthetic initialization. It can be widely used for any STR models with ResNet backbone, instead of training the whole model on synthetic datasets from scratch. Experiments showed the superiority of our BINet on standard benchmarks, especially on challenging and irregular scene text recognition.

Acknowledgment: The work is supported by XSEDE Program of National Science Foundation, and Aspire-II Research Program in University of South Carolina. This work used GPUs provided by the NSF MRI-2018966.

References

1. Al-Zaidy, R., Fung, B.C., Youssef, A.M., Fortin, F.: Mining criminal networks from unstructured text documents. Digital Investigation **8**(3-4), 147–160 (2012) 1
2. Alsharif, O., Pineau, J.: End-to-end text recognition with hybrid hmm maxout models. arXiv preprint arXiv:1310.1811 (2013) 4
3. Atienza, R.: Vision transformer for fast and efficient scene text recognition. arXiv preprint arXiv:2105.08582 (2021) 3
4. Baek, J., Kim, G., Lee, J., Park, S., Han, D., Yun, S., Oh, S.J., Lee, H.: What is wrong with scene text recognition model comparisons? dataset and model analysis. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 4715–4723 (2019) 3, 10
5. Baek, J., Matsui, Y., Aizawa, K.: What if we only use real datasets for scene text recognition? toward scene text recognition with fewer labels. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3113–3122 (2021) 5, 9, 10, 11, 12
6. Bao, W., Lai, W.S., Ma, C., Zhang, X., Gao, Z., Yang, M.H.: Depth-aware video frame interpolation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3703–3712 (2019) 2, 13
7. Bartz, C., Bethge, J., Yang, H., Meinel, C.: Kiss: Keeping it simple for scene text recognition. arXiv preprint arXiv:1911.08400 (2019) 3
8. Bau, D., Zhu, J.Y., Wulff, J., Peebles, W., Strobelt, H., Zhou, B., Torralba, A.: Seeing what a gan cannot generate. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 4502–4511 (2019) 5
9. Bhunia, A.K., Sain, A., Kumar, A., Ghose, S., Chowdhury, P.N., Song, Y.Z.: Joint visual semantic reasoning: Multi-stage decoder for text recognition. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 14940–14949 (2021) 10
10. Bissacco, A., Cummins, M., Netzer, Y., Neven, H.: Photoocr: Reading text in uncontrolled conditions. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 785–792 (2013) 4
11. Chen, X., Wang, T., Zhu, Y., Jin, L., Luo, C.: Adaptive embedding gate for attention-based scene text recognition. Neurocomputing **381**, 261–271 (2020) 3
12. Chen, Y., Li, V.O., Cho, K., Bowman, S.R.: A stable and effective learning strategy for trainable greedy decoding. arXiv preprint arXiv:1804.07915 (2018) 3
13. Cheng, Z., Xu, Y., Bai, F., Niu, Y., Pu, S., Zhou, S.: Aon: Towards arbitrarily-oriented text recognition. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5571–5579 (2018) 3
14. Ch'ng, C.K., Chan, C.S.: Total-text: A comprehensive dataset for scene text detection and recognition. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). vol. 1, pp. 935–942. IEEE (2017) 6
15. Chng, C.K., Liu, Y., Sun, Y., Ng, C.C., Luo, C., Ni, Z., Fang, C., Zhang, S., Han, J., Ding, E., et al.: Icdar2019 robust reading challenge on arbitrary-shaped text-rrc-art. In: 2019 International Conference on Document Analysis and Recognition (ICDAR). pp. 1571–1576. IEEE (2019) 9
16. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018) 5
17. Diaz-Escobar, J., Kober, V.: Natural scene text detection and segmentation using phase-based regions and character retrieval. Mathematical Problems in Engineering **2020** (2020) 2

18. Engelmann, F., Kontogianni, T., Hermans, A., Leibe, B.: Exploring spatial context for 3d semantic segmentation of point clouds. In: IEEE International Conference on Computer Vision workshops. pp. 716–724 (2017) 2
19. Fang, S., Xie, H., Wang, Y., Mao, Z., Zhang, Y.: Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7098–7107 (2021) 2, 3, 4, 5, 6, 9, 10, 11, 12, 13
20. Fang, S., Xie, H., Zha, Z.J., Sun, N., Tan, J., Zhang, Y.: Attention and language ensemble for scene text recognition with convolutional sequence modeling. In: ACM International Conference on Multimedia. pp. 248–256 (2018) 3
21. Gidaris, S., Singh, P., Komodakis, N.: Unsupervised representation learning by predicting image rotations. arXiv preprint arXiv:1803.07728 (2018) 11
22. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. Advances in Neural Information Processing Systems (NeurIPS) **27** (2014) 5
23. Gupta, A., Vedaldi, A., Zisserman, A.: Synthetic data for text localisation in natural images. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2315–2324 (2016) 5, 9, 10
24. Hong, T., Hull, J.J.: Visual inter-word relations and their use in ocr postprocessing. In: Proceedings of 3rd International Conference on Document Analysis and Recognition (ICDAR). vol. 1, pp. 442–445. IEEE (1995) 4
25. Hu, W., Cai, X., Hou, J., Yi, S., Lin, Z.: Gtc: Guided training of ctc towards efficient and accurate scene text recognition. In: Association for the Advancement of Artificial Intelligence (AAAI). vol. 34, pp. 11005–11012 (2020) 4, 10
26. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Deep structured output learning for unconstrained text recognition. arXiv preprint arXiv:1412.5903 (2014) 4, 9
27. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Synthetic data and artificial neural networks for natural scene text recognition. arXiv preprint arXiv:1406.2227 (2014) 5, 9, 10
28. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Reading text in the wild with convolutional neural networks. International Journal of Computer Vision (IJCV) **116**(1), 1–20 (2016) 4
29. Jaderberg, M., Vedaldi, A., Zisserman, A.: Deep features for text spotting. In: European Conference on Computer Vision (ECCV). pp. 512–528. Springer (2014) 4
30. Jung, S., Lee, U., Jung, J., Shim, D.H.: Real-time traffic sign recognition system with deep convolutional neural network. In: International Conference on Ubiquitous Robots and Ambient Intelligence (URAI). pp. 31–34. IEEE (2016) 1
31. Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., Matas, J., Neumann, L., Chandrasekhar, V.R., Lu, S., et al.: Icdar 2015 competition on robust reading. In: 2015 13th International Conference on Document Analysis and Recognition (ICDAR). pp. 1156–1160. IEEE (2015) 9
32. Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., i Bigorda, L.G., Mestre, S.R., Mas, J., Mota, D.F., Almazan, J.A., De Las Heras, L.P.: Icdar 2013 robust reading competition. In: 2013 12th International Conference on Document Analysis and Recognition (ICDAR). pp. 1484–1493. IEEE (2013) 9
33. Krishnan, P., Kovvuri, R., Pang, G., Vassilev, B., Hassner, T.: Textstylebrush: Transfer of text aesthetics from a single example. arXiv preprint arXiv:2106.08385 (2021) 4

34. Kundu, A., Li, Y., Dellaert, F., Li, F., Rehg, J.M.: Joint semantic segmentation and 3d reconstruction from monocular video. In: European Conference on Computer Vision (ECCV). pp. 703–718. Springer (2014) 2
35. Laina, I., Rupprecht, C., Navab, N.: Towards unsupervised image captioning with shared multimodal embeddings. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 7414–7424 (2019) 5
36. Lee, C.Y., Osindero, S.: Recursive recurrent nets with attention modeling for ocr in the wild. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2231–2239 (2016) 3, 4
37. Lee, D.H., et al.: Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In: Workshop on challenges in representation learning, International Conference on Machine Learning (ICML). vol. 3, p. 896 (2013) 11
38. Li, H., Wang, P., Shen, C., Zhang, G.: Show, attend and read: A simple and strong baseline for irregular text recognition. In: Association for the Advancement of Artificial Intelligence (AAAI). vol. 33, pp. 8610–8617 (2019) 3, 5, 10
39. Liao, M., Pang, G., Huang, J., Hassner, T., Bai, X.: Mask textspotter v3: Segmentation proposal network for robust scene text spotting. In: European Conference on Computer Vision (ECCV). pp. 706–722. Springer (2020) 2, 5
40. Liao, M., Zhang, J., Wan, Z., Xie, F., Liang, J., Lyu, P., Yao, C., Bai, X.: Scene text recognition from two-dimensional perspective. In: Association for the Advancement of Artificial Intelligence (AAAI). vol. 33, pp. 8714–8721 (2019) 2, 4, 5
41. Litman, R., Anschel, O., Tsiper, S., Litman, R., Mazor, S., Manmatha, R.: Scatter: selective context attentional scene text recognizer. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11962–11972 (2020) 4, 5
42. Liu, W., Chen, C., Wong, K.Y.K.: Char-net: A character-aware neural network for distorted scene text recognition. In: Association for the Advancement of Artificial Intelligence (AAAI) (2018) 3, 4
43. Liu, W., Chen, C., Wong, K.Y.K., Su, Z., Han, J.: Star-net: a spatial attention residue network for scene text recognition. In: British Machine Vision Conference (BMVC). vol. 2, p. 7 (2016) 2, 4
44. Liu, X., Kawanishi, T., Wu, X., Kashino, K.: Scene text recognition with cnn classifier and wfst-based word labeling. In: 2016 23rd International Conference on Pattern Recognition (ICPR). pp. 3999–4004. IEEE (2016) 4
45. Looije, R., Neerincx, M.A., Cnossen, F.: Persuasive robotic assistant for health self-management of older adults: Design and evaluation of social behaviors. International Journal of Human-Computer Studies (IJHCS) **68**(6), 386–397 (2010) 1
46. Luo, C., Lin, Q., Liu, Y., Jin, L., Shen, C.: Separating content from style using adversarial learning for recognizing text in the wild. International Journal of Computer Vision (IJCV) **129**(4), 960–976 (2021) 5
47. Mishra, A., Alahari, K., Jawahar, C.: Scene text recognition using higher order language priors. In: British Machine Vision Conference (BMVC). BMVA (2012) 4, 9
48. Mishra, A., Alahari, K., Jawahar, C.: Enhancing energy minimization framework for scene text recognition with top-down cues. Computer Vision and Image Understanding (CVIU) **145**, 30–42 (2016) 2, 4
49. Mou, Y., Tan, L., Yang, H., Chen, J., Liu, L., Yan, R., Huang, Y.: Plugnet: Degradation aware scene text recognition supervised by a pluggable super-resolution unit. In: European Conference on Computer Vision (ECCV). pp. 158–174. Springer (2020) 2, 10

50. Nayef, N., Patel, Y., Busta, M., Chowdhury, P.N., Karatzas, D., Khlif, W., Matas, J., Pal, U., Burie, J.C., Liu, C.l., et al.: Icdar2019 robust reading challenge on multi-lingual scene text detection and recognition—rrc-mlt-2019. In: 2019 International Conference on Document Analysis and Recognition (ICDAR). pp. 1582–1587. IEEE (2019) 9
51. Neumann, L., Matas, J.: A method for text localization and recognition in real-world images. In: Asian Conference on Computer Vision (ACCV). pp. 770–783. Springer (2010) 4
52. Park, T., Liu, M.Y., Wang, T.C., Zhu, J.Y.: Semantic image synthesis with spatially-adaptive normalization. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019) 2
53. Phan, T.Q., Shivakumara, P., Tian, S., Tan, C.L.: Recognizing text with perspective distortion in natural scenes. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 569–576 (2013) 9
54. Qiao, Z., Zhou, Y., Wei, J., Wang, W., Zhang, Y., Jiang, N., Wang, H., Wang, W.: Pimnet: a parallel, iterative and mimicking network for scene text recognition. In: ACM International Conference on Multimedia. pp. 2046–2055 (2021) 10, 11
55. Qiao, Z., Zhou, Y., Yang, D., Zhou, Y., Wang, W.: Seed: Semantics enhanced encoder-decoder framework for scene text recognition. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 13528–13537 (2020) 3, 4, 5, 10
56. Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., Sutskever, I.: Zero-shot text-to-image generation. In: International Conference on Machine Learning (ICML). pp. 8821–8831. PMLR (2021) 3
57. Ren, W., Zhang, J., Xu, X., Ma, L., Cao, X., Meng, G., Liu, W.: Deep video dehazing with semantic segmentation. IEEE Transactions on Image Processing (TIP) **28**(4), 1895–1908 (2018) 2
58. Risnumawan, A., Shivakumara, P., Chan, C.S., Tan, C.L.: A robust arbitrary text detection system for natural scene images. Expert Systems with Applications **41**(18), 8027–8048 (2014) 9
59. Shi, B., Bai, X., Yao, C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **39**(11), 2298–2304 (2016) 10
60. Shi, B., Wang, X., Lyu, P., Yao, C., Bai, X.: Robust scene text recognition with automatic rectification. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4168–4176 (2016) 3, 4
61. Shi, B., Yang, M., Wang, X., Lyu, P., Yao, C., Bai, X.: Aster: An attentional scene text recognizer with flexible rectification. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **41**(9), 2035–2048 (2018) 3, 10
62. Shi, B., Yao, C., Liao, M., Yang, M., Xu, P., Cui, L., Belongie, S., Lu, S., Bai, X.: Icdar2017 competition on reading chinese text in the wild (rctw-17). In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). vol. 1, pp. 1429–1434. IEEE (2017) 9
63. Sivic, J., Zisserman, A.: Video google: A text retrieval approach to object matching in videos. In: IEEE/CVF International Conference on Computer Vision (ICCV). vol. 3, pp. 1470–1470. IEEE Computer Society (2003) 1
64. Su, B., Lu, S.: Accurate scene text recognition based on recurrent neural network. In: Asian Conference on Computer Vision (ACCV). pp. 35–48. Springer (2014) 4

65. Sun, Y., Ni, Z., Chng, C.K., Liu, Y., Luo, C., Ng, C.C., Han, J., Ding, E., Liu, J., Karatzas, D., et al.: Icdar 2019 competition on large-scale street view text with partial labeling-rrc-lsvt. In: 2019 International Conference on Document Analysis and Recognition (ICDAR). pp. 1557–1562. IEEE (2019) 9
66. Tchapmi, L., Choy, C., Armeni, I., Gwak, J., Savarese, S.: Segcloud: Semantic segmentation of 3d point clouds. In: 2017 International Conference on 3D Vision (3DV). pp. 537–547. IEEE (2017) 2
67. Tewel, Y., Shalev, Y., Schwartz, I., Wolf, L.: Zero-shot image-to-text generation for visual-semantic arithmetic. arXiv preprint arXiv:2111.14447 (2021) 3
68. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems (NeurIPS). pp. 5998–6008 (2017) 6, 13
69. Veit, A., Matera, T., Neumann, L., Matas, J., Belongie, S.: Coco-text: Dataset and benchmark for text detection and recognition in natural images. arXiv preprint arXiv:1601.07140 (2016) 9
70. Wan, Z., He, M., Chen, H., Bai, X., Yao, C.: Textscanner: Reading characters in order for robust scene text recognition. In: Association for the Advancement of Artificial Intelligence (AAAI). vol. 34, pp. 12120–12127 (2020) 4, 10
71. Wang, J., Li, X., Yang, J.: Stacked conditional generative adversarial networks for jointly learning shadow detection and shadow removal. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1788–1797 (2018) 13
72. Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., et al.: Deep high-resolution representation learning for visual recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2020) 6
73. Wang, K., Babenko, B., Belongie, S.: End-to-end scene text recognition. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 1457–1464. IEEE (2011) 9
74. Wang, K., Belongie, S.: Word spotting in the wild. In: European Conference on Computer Vision (ECCV). pp. 591–604. Springer (2010) 4
75. Wang, S., Wang, Y., Qin, X., Zhao, Q., Tang, Z.: Scene text recognition via gated cascade attention. In: IEEE International Conference on Multimedia and Expo (ICME). pp. 1018–1023. IEEE (2019) 3
76. Wang, T., Zhu, Y., Jin, L., Luo, C., Chen, X., Wu, Y., Wang, Q., Cai, M.: Decoupled attention network for text recognition. In: Association for the Advancement of Artificial Intelligence (AAAI). vol. 34, pp. 12216–12224 (2020) 4, 10
77. Wang, X., Yu, K., Dong, C., Loy, C.C.: Recovering realistic texture in image super-resolution by deep spatial feature transform. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 606–615 (2018) 8
78. Xu, X., Zhang, Z., Wang, Z., Price, B., Wang, Z., Shi, H.: Rethinking text segmentation: A novel dataset and a text-specific refinement approach. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 12045–12055 (2021) 6
79. Yan, R., Peng, L., Xiao, S., Yao, G.: Primitive representation learning for scene text recognition. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 284–293 (2021) 3, 10
80. Yang, M., Guan, Y., Liao, M., He, X., Bian, K., Bai, S., Yao, C., Bai, X.: Symmetry-constrained rectification network for scene text recognition. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 9147–9156 (2019) 3, 10

81. Yang, X., He, D., Zhou, Z., Kifer, D., Giles, C.L.: Learning to read irregular text with attention mechanisms. In: International Joint Conference on Artificial Intelligence (IJCAI). vol. 1, p. 3 (2017) 3
82. Yao, C., Bai, X., Shi, B., Liu, W.: Strokelets: A learned multi-scale representation for scene text recognition. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4042–4049 (2014) 4
83. Ye, J., Chen, Z., Liu, J., Du, B.: Textfusenet: Scene text detection with richer fused features. In: International Joint Conference on Artificial Intelligence (IJCAI). pp. 516–522 (2020) 5
84. Yu, D., Li, X., Zhang, C., Liu, T., Han, J., Liu, J., Ding, E.: Towards accurate scene text recognition with semantic reasoning networks. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 12113–12122 (2020) 2, 3, 5, 6, 10
85. Yue, X., Kuang, Z., Lin, C., Sun, H., Zhang, W.: Robustscanner: Dynamically enhancing positional clues for robust text recognition. In: European Conference on Computer Vision (ECCV). pp. 135–151. Springer (2020) 2, 10
86. Zhan, F., Lu, S.: Esir: End-to-end scene text recognition via iterative image rectification. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2059–2068 (2019) 3, 10
87. Zhang, H., Yao, Q., Yang, M., Xu, Y., Bai, X.: Autostr: Efficient backbone search for scene text recognition. In: European Conference on Computer Vision (ECCV). pp. 751–767. Springer (2020) 4
88. Zhang, R., Zhou, Y., Jiang, Q., Song, Q., Li, N., Zhou, K., Wang, L., Wang, D., Liao, M., Yang, M., et al.: Icdar 2019 robust reading challenge on reading chinese text on signboard. In: 2019 international conference on document analysis and recognition (ICDAR). pp. 1577–1581. IEEE (2019) 9
89. Zhang, X., Wei, Y., Yang, Y., Huang, T.S.: Sg-one: Similarity guidance network for one-shot semantic segmentation. IEEE Transactions on Cybernetics **50**(9), 3855–3865 (2020) 7
90. Zhang, Y., Nie, S., Liu, W., Xu, X., Zhang, D., Shen, H.T.: Sequence-to-sequence domain adaptation network for robust text image recognition. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2740–2749 (2019) 4
91. Zhang, Y., Gueguen, L., Zharkov, I., Zhang, P., Seifert, K., Kadlec, B.: Uber-text: A large-scale dataset for optical character recognition from street-level imagery. In: IEEE International Conference on Computer Vision workshops. vol. 2017, p. 5 (2017) 9
92. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 2223–2232 (2017) 5
93. Zhu, Y., Wang, S., Huang, Z., Chen, K.: Text recognition in images based on transformer with hierarchical attention. In: IEEE International Conference on Image Processing (ICIP). pp. 1945–1949. IEEE (2019) 3