International Journal of Computer Vision (2021) 129:960–976
https://doi.org/10.1007/s11263-020-01411-1

Separating Content from Style Using Adversarial Learning for Recognizing Text in the Wild

Canjie Luo[1] · **Qingxiang Lin**[1] · **Yuliang Liu**[1,2] · **Lianwen Jin**[1,3] · **Chunhua Shen**[2,4]

Received: 19 December 2019 / Accepted: 28 November 2020 / Published online: 5 January 2021

Abstract

Scene text recognition is an important task in computer vision. Despite tremendous progress achieved in the past few years, issues such as varying font styles, arbitrary shapes and complex backgrounds etc. have made the problem very challenging. In this work, we propose to improve text recognition from a new perspective by separating the text content from complex backgrounds, thus making the recognition considerably easier and significantly improving recognition accuracy. To this end, we exploit the generative adversarial networks (GANs) for removing backgrounds while retaining the text content . As vanilla GANs are not sufficiently robust to generate sequence-like characters in natural images, we propose an adversarial learning framework for the generation and recognition of multiple characters in an image. The proposed framework consists of an attention-based recognizer and a generative adversarial architecture. Furthermore, to tackle the issue of lacking paired training samples, we design an interactive joint training scheme, which shares attention masks from the recognizer to the discriminator, and enables the discriminator to extract the features of each character for further adversarial training. Benefiting from the character-level adversarial training, our framework requires only unpaired simple data for style supervision. Each target style sample containing only one randomly chosen character can be simply synthesized online during the training. This is significant as the training does not require costly paired samples or character-level annotations. Thus, only the input images and corresponding text labels are needed. In addition to the style normalization of the backgrounds, we refine character patterns to ease the recognition task. A feedback mechanism is proposed to bridge the gap between the discriminator and the recognizer. Therefore, the discriminator can guide the generator according to the confusion of the recognizer, so that the generated patterns are clearer for recognition. Experiments on various benchmarks, including both regular and irregular text, demonstrate that our method significantly reduces the difficulty of recognition. Our framework can be integrated into recent recognition methods to achieve new state-of-the-art recognition accuracy.

Keywords Text recognition · Attention mechanism · Generative adversarial network · Separation of content and style

1 Introduction

Recognizing text in the wild has attracted great interest in computer vision (Ye and Doermann 2015; Zhu et al. 2016; Yang et al. 2017; Shi et al. 2018; Yang et al. 2019a). Recently, methods based on convolutional neural networks (CNNs) (Wang et al. 2012; Jaderberg et al. 2015, 2016) have significantly improved the accuracy of scene text recognition. Recurrent neural networks (RNNs) (He et al. 2016b; Shi et al. 2016, 2017) and attention mechanism (Lee and Osindero

Communicated by Cha Zhang, Ph.D.

✉ Lianwen Jin
eelwjin@scut.edu.cn

Canjie Luo
canjie.luo@gmail.com

Qingxiang Lin
qingxiang.lin@foxmail.com

Yuliang Liu
liu.yuliang@mail.scut.edu.cn

Chunhua Shen
chunhua.shen@adelaide.edu.au

[1] South China University of Technology, Guangzhou, China

[2] The University of Adelaide, Adelaide, Australia

[3] SCUT-Zhuhai Institute of Modern Industrial Innovation, Guangzhou, China

[4] Monash University, Melbourne, Australia

Fig. 1 Examples of scene text with complex backgrounds, making recognition very challenging

Fig. 2 Text content extraction of (**a**) CycleGAN, (**b**) (Liu et al. 2018b) and (**c**) our method. Our method uses character-level adversarial training and thus better preserves the strokes of every character and removes complex backgrounds

2016; Cheng et al. 2017, 2018; Yang et al. 2017) are also beneficial for recognition.

Nevertheless, recognizing text in natural images is still challenging and largely remains unsolved (Shi et al. 2018). As shown in Fig. 1, text is found in various scenes, exhibiting complex backgrounds. The complex backgrounds cause difficulties for recognition. For instance, the complicated images often lead to attention drift (Cheng et al. 2017) for attention networks. Thus, if the complex background style is normalized to a clean one, the recognition difficulty will significantly decreases.

With the development of GANs (Johnson et al. 2016; Cheng et al. 2019; Jing et al. 2019) in recent years, it is possible to migrate the scene background from a complex style to a clean style in scene text images. However, vanilla GANs are not sufficiently robust to generate sequence-like characters in natural images (Fang et al. 2019). As shown in Fig. 2a, directly applying the off-the-shelf CycleGAN fails to retain some strokes of the characters. In addition, as reported by (Liu et al. 2018b), applying a similar idea of image recovery to normalize the backgrounds for sequence-like objects fails to generate clean images. As illustrated in Fig. 2b, some characters on the generated images are corrupted, which leads to misclassification. One possible reason for this may be the discriminator is designed to focus on non-sequential object with a global coarse supervision (Zhang et al. 2019). Therefore, the generation of sequence-like characters requires more fine-grained supervision.

One potential solution is to employ the pixel-wise supervision (Isola et al. 2017), which requires paired training samples aligning at pixel level. However, it is impossible to collect paired training samples in the wild. Furthermore, annotating scene text images with pixel-wise labels can be intractably expensive. To address the lack of paired data, it is possible to synthesize a large number of paired training samples, because synthetic data is cheaper to obtain. This may be why most state-of-the-art scene text recognition methods (Cheng et al. 2018; Shi et al. 2018; Luo et al. 2019) only use synthetic samples (Jaderberg et al. 2014a; Gupta et al. 2016) for training, as tens of millions of training data are immediately available. However, experiments of (Li et al. 2019) suggest that there exists much room for improvement in synthesis engines. Typically a recognizer trained using real data significantly outperforms the ones trained using synthetic data due to the domain gap between artificial and real data. Thus, to enable broader application, our goal here is to improve GANs to meet the requirement of text image generation and address the unpaired data issue.

We propose an adversarial learning framework with an interactive joint training scheme, which achieves success in separating text content from background noises by using only source training images and the corresponding text labels. The framework consists of an attention-based recognizer and a generative adversarial architecture. We take advantage of the attention mechanism in the attention-based recognizer to extract the features of each character for further adversarial training. In contrast to global coarse supervisions, character-level adversarial training provides guidance for the generator in a fine-grained manner, which is critical to the success of our approach.

Our proposed framework is a meta framework. Thus, recent mainstream recognizers (Cheng et al. 2018; Shi et al. 2018; Luo et al. 2019; Li et al. 2019) equipped with attention-based decoders (Bahdanau et al. 2015) can be integrated into our framework. As illustrated in Fig. 3, the attention-based recognizer predicts a mask for each character, which is shared with the discriminator. Thus, the discriminator is able to focus on every character and guide the generator to filter out various background styles while retaining the character content. Benefiting from the advantage of the attention mechanism, the interactive joint training scheme requires only the images and corresponding text labels, without requirement of character bounding box annotation. Simultaneously, the target style training samples can be simply synthesized online during the training. As shown in Fig. 4, for each target style sample, we randomly choose one character and simply render the character onto a clean background. Each sample contains a black character on a white background or a white character on a black background. The target style samples are character-level, whereas the input style samples are word-level. The unpaired training samples suggest our training process is flexible.

Moreover, we take a further step of the interactive joint training scheme. In addition to the sharing of attention masks, we proposed a feedback mechanism, which bridge the gap

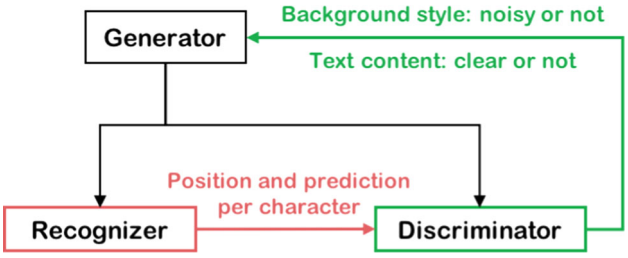

Fig. 3 Interactive joint training of our framework. The attention-based recognizer shares the position and prediction of every character with the discriminator, whereas the discriminator learns from the confusion of the recognizer, and guides the generator so that it can generate clear text content and clean background style to ease reading

between the recognizer and the discriminator. The discriminator guides the generator according to the confusion of the recognizer. Thus, the erroneous character patterns on the generated images are corrected. For instance, the patterns of the characters "C" and "G" are similar, which can easily cause failed prediction of the recognizer. After the training using our feedback mechanism, the generated patterns are more discriminative, and incorrect predictions on ambiguous characters can be largely avoided.

To summarize, our main contributions are as follows.

1) We propose a framework that separates text content from complex background styles to reduce recognition difficulty. The framework consists of an attention-based recognizer and a generative adversarial architecture. We devise an interactive joint training of them, which is critical to the success of our approach.
2) The shared attention mask enables character-level adversarial training. Thus, the unpaired target style samples can be simply synthesized online. The training of our framework requires only the images and corresponding text labels. Additional annotations such as bounding boxes or pixel-wise labels are unnecessary.
3) We further propose a feedback mechanism to improve the robustness of the generator. The discriminator learns from the confusion of the recognizer and guides the generator so that it can generate clear character patterns that facilitate reading.
4) Our experiments demonstrate that mainstream recognizers can benefit from our method and achieve new state-of-the-art performance by extracting text content from complex background styles. This suggests that our framework is a meta-framework, which is flexible for integration with recognizers.

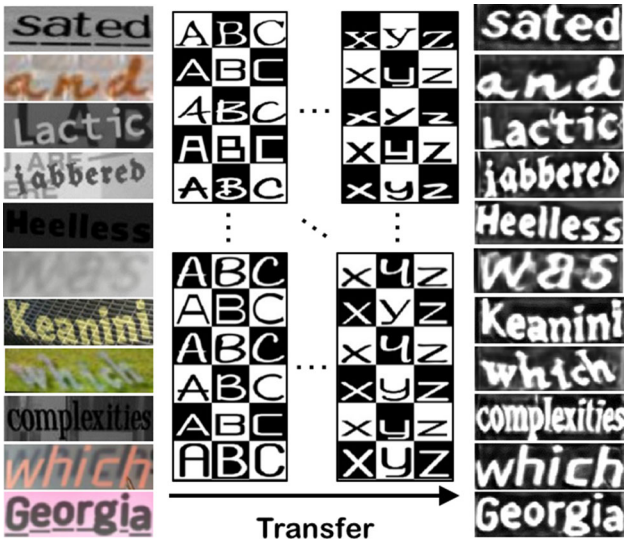

Fig. 4 Training samples and generations. Left: Widely used training datasets released by (Jaderberg et al. 2014a) and (Gupta et al. 2016). Middle: Unpaired target style samples, which are character-level and synthesized online. Right: Output of the generator

2 Related Work

In this section, we review the previous methods that are most relevant to ours with respect to 2 categories: scene text recognition and generative adversarial networks.

2.1 Scene Text Recognition

Overviews of the notable work in the field of scene text detection and recognition have been provided by (Ye and Doermann 2015) and (Zhu et al. 2016). The methods based on neural networks outperform the methods with hand crafted features, such as HOG descriptors (Dalal and Triggs 2005), connected components (Neumann and Matas 2012), strokelet generation (Yao et al. 2014), and label embedding (Rodriguez-Serrano et al. 2015), because the trainable neural network is able to adapt to various scene styles. For instance, (Bissacco et al. 2013) applied a network with five hidden layers for character classification, and (Jaderberg et al. 2015) proposed a CNN for unconstrained recognition. The CNN-based methods significantly improve the performance of recognition.

Moreover, the recognition models yield better robustness when they are integrated with RNNs (He et al. 2016b; Shi et al. 2016, 2017) and attention mechanisms (Lee and Osindero 2016; Cheng et al. 2017, 2018; Yang et al. 2017). For example, (Shi et al. 2017) proposed an end-to-end trainable network using both CNNs and RNNs, namely CRNN. (Lee and Osindero 2016) proposed a recursive recurrent network using attention modeling for scene text recognition. (Cheng et al. 2017) used a focusing attention network to correct atten-

tion alignment shifts caused by the complexity or low-quality of images. These methods have made great progress in regular scene text recognition.

With respect to irregular text, the irregular shapes introduce more background noise into the images, which increases recognition difficulty. To tackle this problem, (Yang et al. 2017) and (Li et al. 2019) used the two-dimensional (2D) attention mechanism for irregular text recognition. (Liao et al. 2019b) recognized irregular scene text from a 2D perspective with a semantic segmentation network. Additionally, (Liu et al. 2016), (Shi et al. 2016, 2018), and (Luo et al. 2019) proposed rectification networks to transform irregular text images into regular ones, which alleviates the interference of the background noise, and the rectified images become readable by a one-dimensional (1D) recognition network. (Yang et al. 2019a) used character-level annotations for supervision for a more accurate description for rectification. Despite the many praiseworthy efforts that have been made, irregular scene text on complex backgrounds is still difficult to recognize in many cases.

2.2 Generative Adversarial Networks

With the widespread application of GANs (Goodfellow et al. 2014; Mao et al. 2017; Odena et al. 2017; Zhu et al. 2017), font generation methods (Azadi et al. 2018; Yang et al. 2019b) using adversarial learning have been successful on document images. These methods focus on the style of a single character and achieve incredible visual effects.

However, our goal is to perform style normalization on noisy background, rather than the font, size or layout. A further challenge is to keep multiple characters for recognition. That means style normalization of the complex backgrounds of scene text images requires accurate separation between the text content and background noise. Traditional binarization/segmentation methods (Casey and Lecolinet 1996) typically work well on document images, but fail to handle the substantial variation in text appearance and the noise in natural images (Shi et al. 2018). Style normalization of background in scene text images remains an open problem.

Recently, several attempts on scene text generation have taken a crucial step forward. (Liu et al. 2018b) guided the feature maps of an original image towards those of a clean image. The feature-level guidance reduces the recognition difficulty, whereas the image-level guidance does not result in a significant improvement in text recognition performance. (Fang et al. 2019) designed a two-stage architecture to generate repeated characters in images. An additional 10k synthetic images boost the performance, but more synthetic images do not improve accuracy linearly. (Wu et al. 2019) edited text in natural images using a set of corresponding synthetic training samples to preserve the style of both background and text. These methods provided sufficient visualized examples. However, the poor recognition performance on complex scene text remains a challenging problem.

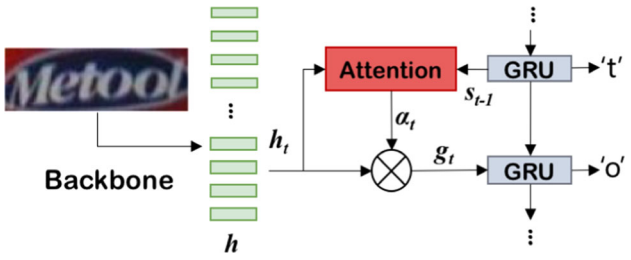

Fig. 5 Attention decoder, which recurrently attends to informative regions and outputs predictions

We are interested in taking a further step to enable recognition performance to benefit from generation. Our method integrates the advantages of the attention mechanism and the GAN, and jointly optimizes them to achieve better performance. The text content is separated from various background styles, which are normalized for easier reading.

3 Methodology

We design a framework to separate text content from noisy background styles, through an interactive joint training of an attention-based recognizer and a generative adversarial architecture. The shared attention masks from the attention-based recognizer enable character-level adversarial training. Then, the discriminator guides the generator to achieve background style normalization. In addition, a feedback mechanism bridges the gap between the discriminator and recognizer. The discriminator guides the generator according to the confusion of the recognizer. Thus, the generator can generate clear character patterns that facilitate reading.

In this section, we first introduce the attention decoder in mainstream recognizers. Then, we present a detailed description of the interactive joint training scheme.

3.1 Attention Decoder

To date the attention decoder (Bahdanau et al. 2015) has become widely used in recent recognizers (Shi et al. 2018; Luo et al. 2019; Li et al. 2019; Yang et al. 2019a). As shown in Fig. 5, the decoder sequentially outputs predictions ($y_1, y_2..., y_N$) and stops processing when it predicts an end-of-sequence token $''EOS''$ (Sutskever et al. 2014). At time step t, output y_t is given by

$$y_t = \text{softmax}(\boldsymbol{W}_{\text{out}}\boldsymbol{s}_t + b_{\text{out}}), \tag{1}$$

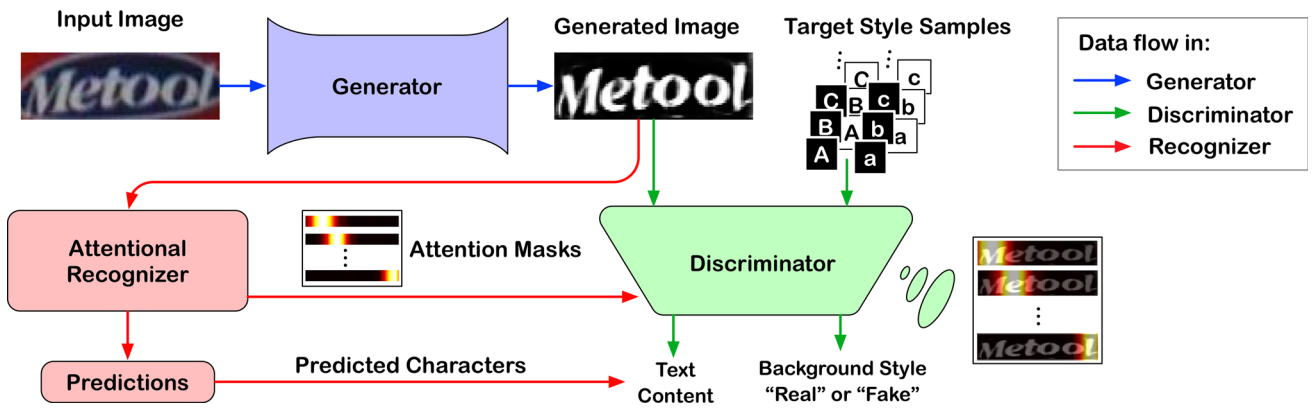

Fig. 6 Interactive joint training. The recognizer shares attention masks with the discriminator, whereas the discriminator learns from the predictions of the recognizer and updates the generator using ground truth. The shared attention masks work on feature maps, which we present in the generated images for better visualization

where $\boldsymbol{s}_t$ is the hidden state at the t-th step. Then, we update $\boldsymbol{s}_t$ by

$$\boldsymbol{s}_t = GRU(\boldsymbol{s}_{t-1}, (y_{t-1}, \boldsymbol{g}_t)), \tag{2}$$

where $\boldsymbol{g}_t$ represents the glimpse vectors

$$\boldsymbol{g}_t = \sum_{i=1}^{n}(\alpha_{t,i}\boldsymbol{h}_i), \boldsymbol{\alpha}_t \in \mathbb{R}^n, \tag{3}$$

where $\boldsymbol{h}_i$ denotes the sequential feature vectors. Vector $\boldsymbol{\alpha}_t$ is the vector of attention mask, expressed as follows:

$$\alpha_{t,i} = \frac{\exp(e_{t,i})}{\sum_{j=1}^{n}(\exp(e_{t,j}))}, \tag{4}$$

$$e_{t,i} = \boldsymbol{w}^{\mathrm{T}}\mathrm{Tan}h(\boldsymbol{W}_s\boldsymbol{s}_{t-1} + \boldsymbol{W}_h\boldsymbol{h}_i + b). \tag{5}$$

Here, $\boldsymbol{W}_{out}$, b_{out}, $\boldsymbol{w}^{\mathrm{T}}$, $\boldsymbol{W}_s$, $\boldsymbol{W}_h$ and b are trainable parameters. Note that y_{t-1} is the $(t-1)$-th character in the ground truth in the training phase, whereas it is the previously predicted output in the testing phase. The training set is denoted as $\boldsymbol{D} = \{I_i, Y_i\}, i = 1...N$. The optimization is to minimize the negative log-likelihood of the conditional probability of $\boldsymbol{D}$ as follows:

$$\mathcal{L}_{\mathrm{reg}} = -\sum_{i=1}^{N}\sum_{t=1}^{|Y_i|}\log p(Y_{i,t} \mid I_i; \theta), \tag{6}$$

where $Y_{i,t}$ is the ground truth of the t-th character in I_i and θ denotes the parameters of the recognizer.

3.2 Interactive Joint Training for Separating Text from Backgrounds

As vanilla discriminator is designed for non-sequential object with a global coarse supervision, directly employing a discriminator fails to provide effective guidance for the generator. In contrast to apply a global discriminator, we supervise the generator in a fine-grained manner, namely, character-level adversarial learning, by taking advantage of the attention mechanism. Training the framework at character level also reduces the complexity of the preparation of target style data. Every target style sample containing one character can be easily synthesized online.

3.2.1 Sharing of Attention Masks

Given an image I as input, the goal of our generator G is to generate a clean image I' without a complex background. The discriminator D encodes the image I' as

$$\boldsymbol{E} = Encode(I'). \tag{7}$$

With similar settings of the backbone in recognizer (e.g., kernel size, stride size and padding size in the convolutional and pooling layers), the encoder in the discriminator is designed to output an embedding vector $\boldsymbol{E}_i$ with the same size as that of $\boldsymbol{h}_i$ in Eq. (3), which enables the recognizer to share attention mask $\boldsymbol{\alpha}_t$ with the discriminator. After that, character-level features of the generation are extracted by

$$\boldsymbol{F}_{gen,t} = \sum_{i=1}^{n}(\alpha_{t,i}\boldsymbol{E}_i), \boldsymbol{\alpha}_t \in \mathbb{R}^n. \tag{8}$$

The extracted character features are used for further adversarial training.

3.2.2 Unpaired Target Style Samples

Benefiting from our character-level adversarial learning, target style samples can be simply synthesized online. As illustrated in Figs. 4 and 6, every target style sample contains only a black character on a white background or a white character on a black background. The characters are randomly chosen. Following the previous methods for data synthesis (Jaderberg et al. 2014a; Gupta et al. 2016), we collect fonts[1] to synthesize the target style samples. The renderer is a simple and publicly available engine[2] that can efficiently synthesize samples online. Owing to the diversity of the fonts, the font sensitivity of the discriminator is thereby decreased, which enables the discriminator to focus on the background styles.

Because there is only one character in a target style image, we apply global average-pooling to the embedding features for every target style sample I_t as follows:

$$\boldsymbol{F}_{tgt} = \text{averagePooling}\big(\text{Encode}(I_t)\big). \tag{9}$$

The features of the t-th character in the generated image $\boldsymbol{F}_{gen,t}$ and the target style sample $\boldsymbol{F}_{tgt}$ are prepared for the following adversarial training.

3.2.3 Adversarial Training on Style

We use a style classifier in the discriminator to classify the style of characters in the generated images as fake and the characters in the target style samples as real. We use the 0−1 binary coding (Mao et al. 2017) for style adversarial training, which is formulated as

$$\begin{aligned} \min_D \mathcal{L}_s &= \mathbb{E}_{I_t}[\big(1 - Style(\boldsymbol{F}_{tgt})\big)^2] + \mathbb{E}_{I',t}\big[Style(\boldsymbol{F}_{gen,t})^2\big], \\ \min_G \mathcal{L}_s &= \mathbb{E}_{I',t}\big[\big(1 - Style(\boldsymbol{F}_{gen,t})\big)^2\big], \end{aligned} \tag{10}$$

where $Style(\cdot)$ denotes the style classifier.

The advantages of character-level adversarial training are threefold: (1) Because the background is complicated in a scene text image, the background noise varies substantially in different character regions. Considering the text string as a whole and supervising the training in a global manner may cause the failure of the generator, as discussed previously in Sect. 1 and Fig. 2. Thus, we encourage the discriminator to inspect the generation in a more fine-grained manner, namely, character-level supervision, which contributes to the effective learning. (2) Training at character level brings a benefit for the preparation of target style data. For the synthesis of a text string, it is necessary to consider the text shape, the space between neighboring characters and the rotation of every character (Jaderberg et al. 2014a; Gupta et al. 2016). In contrast, we can simply synthesize only one character on a clean background for every target style sample. Therefore, our target style samples can be simply synthesized online during the training. (3) The training is free of the need for paired data. Because the attention mechanism decomposes a text string into several characters and benefits the further training, only input scene text images and corresponding text labels are required. Hence, our framework is potentially flexible enough to make full use of available data to gain robustness.

3.2.4 Feedback Mechanism

As our goal is to improve recognition performance, we are not only interested in the styles of the backgrounds, but also the quality of the generated content. Therefore, we use a content classifier in the discriminator to supervise content generation.

In contrast to the previous work auxiliary classifier GAN (Odena et al. 2017), which used ground truth to supervise the content classifier, our content classifier learns from the predictions of the recognizer. This bridges the gap between the recognizer and discriminator. The discriminator thus can guide the generator according to the confusion of the recognizer. After the training with this feedback mechanism, the generated patterns are more discriminative, which facilitates recognition. The details of the feedback mechanism are present as follows.

The generator G and discriminator D are updated by alternately optimizing

$$\begin{aligned} \min_D \mathcal{L}_{c,D} \quad &= \mathbb{E}_{(I,P),(I_t,GT)}[-\log \text{Content}(GT|\boldsymbol{F}_{tgt}) \\ &\quad -\frac{1}{|P|}\sum_{t=1}^{|P|}\log \text{Content}(P_t|\boldsymbol{F}_{gen,t})], \end{aligned} \tag{11}$$

$$\min_G \mathcal{L}_{c,G} = \mathbb{E}_{I,GT}[-\frac{1}{|GT|}\sum_{t=1}^{|GT|}\log Content(GT_t|\boldsymbol{F}_{gen,t})], \tag{12}$$

where GT denotes the ground truth of the input image I and target style sample I_t. In addition, $Content(\cdot)$ is the content classifier. Note that the discriminator learns from the predictions P on I of the recognizer, whereas it uses GT of I to update the generator. This is an adversarial process that is similar to that of GAN training (Goodfellow et al. 2014; Mao et al. 2017; Odena et al. 2017; Zhu et al. 2017). They use different labels for the discriminator and generator, but backpropagate the gradient using the same parameters as those of the discriminator. Alternately optimizing the discriminator and generator achieves adversarial learning.

[1] https://fonts.google.com

[2] https://pillow.readthedocs.io/en/stable/reference/ImageDraw.html

There are some substitution errors in the predictions P that are different from the GT. Therefore, the second term of the right side in Eq. (11) can be formulated as content adversarial training as

$$
\begin{aligned}
&-\frac{1}{|P|}\sum_{t=1}^{|P|}\log Content(P_t|\boldsymbol{F}_{gen,t}) = \\
&-\frac{1}{|P|}[\sum_{i=1}^{|P_{real}|}\log Content(P_{real,i}|\boldsymbol{F}_{gen,i}) \\
&\quad+\sum_{j=1}^{|P_{fake}|}\log Content(P_{fake,j}|\boldsymbol{F}_{gen,j})],
\end{aligned} \tag{13}
$$

where P_{real} and P_{fake} present the correct and incorrect predictions of the recognizer, respectively. Note that $P_{real} \cup P_{fake} = P$.

Since the discriminator with the content classifier learns from the predictions of the recognizer, it guides the generator to correct erroneous character patterns in the generated images. For instance, similar patterns such as "C" and "G", or "O" and "Q", may cause failed prediction of the recognizer. If a "G" is transformed to look more like a "C" and the recognizer predicts it to be a "C", the discriminator will learn that the pattern is a "C" and guide the generator to generate a clearer "G". We show more examples and further discuss this issue in Sect. 4.

3.2.5 Interactive Joint Training

The pseudocode of the interactive joint training scheme is presented in Algorithm 1. During the training of our framework, we found that the discriminator often learns faster than the generator. A similar problem has also been reported by others (Berthelot et al. 2017; Heusel et al. 2017). The Wasserstein GAN (Arjovsky et al. 2017) uses more update steps for the generator than the discriminator. We simply adjust the number of steps according to a balance factor $\beta \in (0, 1)$. If the discriminator learns faster than the generator, then the value of β decreases, potentially resulting in a pause during the update steps for the discriminator. In practice, this trick contributes to the training stability of the generator.

We first sample a set of input samples, and randomly synthesize unpaired samples of target style. Then, the recognizer makes predictions on the generated images and shares its attention masks with the discriminator. To avoid the effects of incorrect alignment between character features and labels (Bai et al. 2018), we filter out some predictions using the metrics of edit distance and string length. The corresponding images are also filtered out. Only substitution errors exist in the remaining predictions. Finally, the discriminator and generator are alternately optimized to achieve adversarial learning.

After the adversarial training, the generator can separate text content from complex background styles. The generated patterns are clearer and easier to read. As illustrated in Fig. 7, the generator works well on both regular text and slanted/curved text. Because the irregular shapes of the text introduce more surrounding background noise, the recognition difficulty can be significantly reduced by using our method.

4 Experiments

In this section, we provide the training details and report the results of extensive experiments on various benchmarks, including both regular and irregular text datasets, demonstrating the effectiveness and generality of our method.

As paired text images in the wild are not available and there exists great diversity in the number of characters and image structure between the input images and our target style images, popular GAN metrics such as the inception

Algorithm 1: Interactive joint training

Discriminator: D; Generator: G;
Batch size: B;
Balance factor: β (initialized as 1.0);
while *not at the end of training* **do**
  Sample B training images as I, $I' = G(I)$;
  Randomly synthesize B target style samples as I_t;
  Obtain the predictions P on I';
  Use I' and GT to update the recognizer: min $\mathcal{L}_{reg}$, and obtain attention masks for the D;
  $I_{chosen} = \varnothing$;
  $P_{chosen} = \varnothing$;
  $GT_{chosen} = \varnothing$;
  for i *in* $1 : B$ **do**
    if $length(P_i) = length(GT_i)$ **then**
      if *edit distance of* $(P_i, GT_i) \leq 1$ **then**
        $I_{chosen} \leftarrow I_{chosen} \cup \{I'_i\}$;
        $P_{chosen} \leftarrow P_{chosen} \cup \{P_i\}$;
        $GT_{chosen} \leftarrow GT_{chosen} \cup \{GT_i\}$;
      end
    end
  end
  if $I_{chosen} \neq \varnothing$ **then**
    Generate a random number $k \in [0, 1)$;
    if $k \leq \beta$ **then**
      Use I_{chosen}, P_{chosen}
      to update the D: $\max_D \mathcal{L}_s$, $\min_D \mathcal{L}_{c,D}$;
    end
    Use I_{chosen}, GT_{chosen}
    to update the G: $\min_G \mathcal{L}_s$, $\min_G \mathcal{L}_{c,G}$;
    $\beta \leftarrow \frac{\mathcal{L}_s + \mathcal{L}_{c,D}}{\mathcal{L}_s + \mathcal{L}_{c,G}}$;
  end
end

Fig. 7 Generated images for (**a**) regular and (**b**) irregular text. Input images are on the left and the corresponding generated images are on the right. The text content is separated by the generator from the noisy background styles. In the generated images, the font style tends to be an average style

score (Salimans et al. 2016) and Fréchet inception distance (Heusel et al. 2017) cannot be directly applied in our evaluation. Instead, we use the word accuracy of recognition, which is a more straightforward metric, and is of interest for our target task, to measure the performance of all the methods. Recall that our goal here is to improve recognition accuracy.

4.1 Datasets

4.1.1 SynthData

Which contains 6-million data released by (Jaderberg et al. 2014a) and 6-million data released by (Gupta et al. 2016), is a widely used training dataset. Following the most recent work for fair comparison, we select it as the training dataset. Only word-level labels are used, but other extra annotation is unnecessary in our framework. The model is trained using only synthetic text images, without any fine-tuning for each specific dataset.

IIIT5K-Words (Mishra et al. 2012) (IIIT5K) contains 3,000 cropped word images for testing. Every image has a 50-word lexicon and a 1,000-word lexicon. The lexicon consists of the ground truth and some randomly picked words.

Street View Text (Wang et al. 2011) (SVT) was collected from the Google Street View, and consists of 647 word images. Each image is associated with a 50-word lexicon. Many images are severely corrupted by noise and blur or have very low resolutions.

ICDAR 2003 (Lucas et al. 2003) (IC03) contains 251 scene images that are labeled with text bounding boxes. For fair comparison, we discarded images that contain non-alphanumeric characters or those have fewer than three characters, following (Wang et al. 2011). The filtered dataset contains 867 cropped images. Lexicons comprise of a 50-word lexicon defined by (Wang et al. 2011) and a "full lexicon". The latter lexicon combines all lexicon words.

ICDAR 2013 (Karatzas et al. 2013) (IC13) inherits most of its samples from IC03. It contains 1,015 cropped text images. No lexicon is associated with this dataset.

SVT-Perspective (Quy Phan et al. 2013) (SVT-P) contains 645 cropped images for testing. Images were selected from side-view angle snapshots in Google Street View. Therefore, most images are perspective distorted. Each image is associated with a 50-word lexicon and a full lexicon.

CUTE80 (Risnumawan et al. 2014) (CUTE) contains 80 high-resolution images taken of natural scenes. It was specifically collected for evaluating the performance of curved text recognition. It contains 288 cropped natural images for testing. No lexicon is associated with this dataset.

ICDAR 2015 (Karatzas et al. 2015) (IC15) contains 2077 images by cropping the words using the ground truth word bounding boxes. (Cheng et al. 2017) filtered out some extremely distorted images and used a small evaluation set (referred as IC15-S) containing only 1811 test images.

4.2 Implementation Details

As our proposed method is a meta-framework for recent attention-based recognition methods (Shi et al. 2018; Luo et al. 2019; Li et al. 2019; Yang et al. 2019a), recent recognizers can be readily integrated with our framework. Thus the recognizer implementation follows their specific design. Here we present details of the discriminator, generator, and training.

4.2.1 Generator

The generator is a feature pyramid network (FPN)-like (Lin et al. 2017) architecture that consists of eight residual units. Each residual unit comprises a 1×1 convolution followed by two 3×3 convolutions. Feature maps are downsampled by 2×2 stride convolutions in the first three residual units. The numbers of output channels of the first four residual units are 64, 128, 256, and 256, respectively. The last four units are symmetrical with the first four, but we upsample the feature map by simple resizing. We apply element-wise addition to the output of the third and fifth units. At the top of the generator, there are two convolution layers that have 16 filters and one filter, respectively.

4.2.2 Discriminator

The encoder in the discriminator consist of 7 convolutional layers that have 16, 64, 128, 128, 192 and 256 filters. Their kernel sizes are all 3×3, except for the size of the last one, which is 2×2. The first, second, fourth and sixth convolutional layers are each followed by an average-pooling layer. Using settings similar to those of the backbone in the recognizer (e.g., kernel size, stride size and padding size in the convolutional and pooling layers), the output size of the encoder can be controlled to meet the requirements of the attention mask sharing of the recognizer. Both the style and content classifiers in the discriminator are one-layer fully connected networks.

4.2.3 Training

We use Adam (Kingma et al. 2015) to optimize the GAN. The learning rate is set to 0.002. It is decreased by a factor of 0.1 at epochs 2 and 4. In the interactive joint training, we utilize the attention mechanism in the recognizer. Therefore, an optimized attention decoder is necessary to enable the interaction. To accelerate the training process, we pre-trained the recognizer for three epochs.

4.2.4 Implementation

We implement our method using the PyTorch framework (Paszke et al. 2017). The target style samples are resized to 32×32. Input images are resized to 64×256 for the generator and 32×100 for the recognizer. The outputs of the generator are also resized to 32×100. When the batch size is set to 64, the training speed is approximately 1.7 iterations/sec. Our method takes an average of 1.1 ms to generate an image using an NVIDIA GTX-1080Ti GPU.

4.3 Ablation Study

4.3.1 Experiment Setup

To investigate the effectiveness of separating text content from noisy background styles, we conduct an ablation analysis by using a simple recognizer. The backbone of the recognizer is a 45-layer residual network (He et al. 2016a), which is a popular architecture (Shi et al. 2018). On the top of the backbone, there is an attention-based decoder. In the decoder, the number of GRU hidden units is 256. The decoder outputs 37 classes, including 26 letters, 10 digits, and a symbol that represented "EoS″. The training data is SynthData. We evaluate the recognizer on seven benchmarks, including regular and irregular text.

4.3.2 Input of the Recognizer

We study the contribution of our method by replacing the generated image with the corresponding input image. The results are listed in Table 1. The recognizer trained using SynthData serves as a baseline. Compared to the baseline, the clean images generated by our method boost recognition performance. We observe that the improvement is more substantial on irregular text. One notable improvement is an accuracy increase of 6.6% on CUTE. One possible reason for this is that the irregular text shapes introduce more background noise than the regular ones. Because our method removes the surrounding noise and extracts the text content for recognition, the recognizer can thus focus on characters and avoid noisy interference.

With respect to regular text, the baseline is much higher and there is less room for improvement, but our method also shows advantages in recognition performance. The gain of performance on several kinds of scene text, including low quality images in SVT and real scene images in IC03/IC13, suggests the generality our method. To summarize, the generated clean images of our proposed method greatly decrease recognition difficulty.

4.3.3 Style Supervision

We study the necessity of style supervision by disabling the style classifier in the discriminator. Without style adversarial training, the background style normalization is only weakly supervised by the content label. As shown in the Fig. 8, the generated images suffer from severe image degradation, which leads to poor robustness of the recognizer. The quantitative recognition results of not using/using the style supervision are presented in the second and third row of Table 2. The significant gaps indicate that without the style supervision, the quality of the generated images is insufficient for recognition training. Thus, the style adversarial training is necessary and is used in the basic design of our method.

Table 1 Word accuracy on the testing datasets using different inputs The recognizer is trained on the source and generated images, respectively

Input image	Regular text				Irregular text		
	IIIT5K	SVT	IC03	IC13	SVT-P	CUTE	IC15
Source I	92.2	85.9	94.0	90.7	75.7	74.3	72.0
Generation I'	**92.5**	**86.6**	**95.0**	**91.4**	**79.2**	**80.9**	**73.0**

The best performing results are shown in bold font

Fig. 8 Visualization of background normalization weakly supervised by content label

4.3.4 Feedback Mechanism

We also study the effectiveness of the content classifier in the discriminator and the proposed feedback mechanism. In this experiment, we first disable the content classifier. Therefore, there is no content supervision. Only a style adversarial loss supervises the generator. The result is shown in the first row in Table 2. The accuracy on the generated images decreases to nearly zero. We observe that the generator fails to retain the character patterns for recognition. As the content classifier is designed for assessing the discriminability and diversity of samples (Odena et al. 2017), it is important to guide the generator so that it can determine informative character patterns and retain them for recognition. When the content supervision is not available, the generator is easily trapped into failure modes, namely mode collapse (Salimans et al. 2016). Therefore, the content supervision in the discriminator is necessary.

Then we enable the content classifier and replace the supervision in $\mathcal{L}_c$ with the ground truth. This setting is similar to that of the auxiliary classifier GANs (Odena et al. 2017), which use content supervision for discriminability and diversity in the style adversarial training. After this process, the generated text images contain text content for recognition.

Finally, we replace the content supervision with the predictions of the recognizer. The discriminator thus learns from the confusion of the recognizer, and guides the generator so that it can refine the character patterns to be easier to read. Therefore, the adversarial training is more relevant to the recognition performance. As shown in Table 2, the feedback mechanism further improves the robustness of the generator and benefits the recognition performance.

One interesting observation is that on the SVT-P testing set, the accuracy on the source image (75.7% in Table 1) is higher than that on the generated image with content supervision of the ground truth (75.0% in Table 2). We observe the source samples and find that most images are severely corrupted by noise and blur. Some of them have low resolutions. The characters in the generated samples are also difficult to distinguish. After training with the feedback mechanism, the generator is able to generate clear patterns that facilitate reading, which boosts the recognition accuracy from 75.0 to 79.2%. As illustrated in Fig. 9, the predictions of "C" and "N" are corrected to "G" and "M", respectively. The clear characters in the generated images are easier to read.

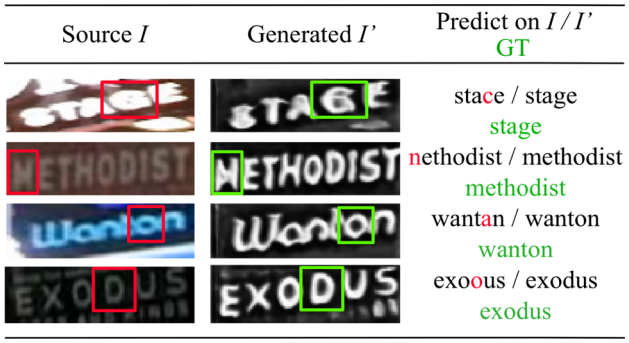

Fig. 9 Predictions of challenging samples in the SVT-P testing set. Recognition errors are marked as red characters. Confusing and distinct patterns are marked by red and green bounding boxes, respectively

Fig. 10 Comparison between the OTSU (Otsu 1979) and our method

Table 2 Word accuracy on generated images using variants of content supervision for the discriminator. Losses $\mathcal{L}_s$ and $\mathcal{L}_c$ denote style loss and content loss, respectively

Content supervision	Adversarial loss		Feedback mechanism	Testing set		
	$\mathcal{L}_s$	$\mathcal{L}_c$		SVT-P	CUTE	IC15
None	✓	×	×	Failed	Failed	Failed
Ground truth	×	✓	×	71.9	74.7	65.5
Ground truth	✓	✓	×	75.0	78.4	72.5
Prediction of recognizer	✓	✓	✓	**79.2**	**80.9**	**73.0**

The best performing results are shown in bold font

Table 3 Word accuracy on testing datasets using different transformation methods

Transformation	Method	Regular text				Irregular text		
		IIIT5K	SVT	IC03	IC13	SVT-P	CUTE	IC15
Style normalization	OTSU (Otsu 1979)	70.3	65.4	76.0	76.0	46.5	50.3	49.3
	CycleGAN (Zhu et al. 2017)	43.6	21.3	37.0	35.9	14.6	18.8	18.0
	Ours	92.5	86.6	95.0	91.4	79.2	80.9	73.0
+ Rectification	Ours + ASTER (Shi et al. 2018)	94.0	90.0	95.6	93.3	81.6	85.1	78.1
	Ours + ESIR (Zhan and Lu 2019)	**94.1**	**90.6**	**96.0**	**94.2**	**82.2**	**87.8**	**78.5**

The best performing results are shown in bold font

Table 4 Word accuracy on regular benchmarks

Method	Add.	IIIT5K			SVT		IC03			IC13
		50	1k	0	50	0	50	Full	0	0
Yao et al. (2014)		80.2	69.3	–	75.9	–	88.5	80.3	–	–
Jaderberg et al. (2014b)		–	–	–	86.1	–	96.2	91.5	–	–
Su and Lu (2014)		–	–	–	83.0	–	92.0	82.0	–	–
Rodriguez-Serrano et al. (2015)		76.1	57.4	–	70.0	–	–	–	–	–
Gordo (2015)		93.3	86.6	–	91.8	–	–	–	–	–
Jaderberg et al. (2015)		95.5	89.6	–	93.2	71.7	97.8	97.0	89.6	81.8
Jaderberg et al. (2016)		97.1	92.7	–	95.4	80.7	98.7	98.6	93.1	90.8
Shi et al. (2016)		96.2	93.8	81.9	95.5	81.9	98.3	96.2	90.1	88.6
Lee and Osindero (2016)		96.8	94.4	78.4	96.3	80.7	97.9	97.0	88.7	90.0
Liu et al. (2016)		97.7	94.5	83.3	95.5	83.6	96.9	95.3	89.9	89.1
Shi et al. (2017)		97.8	95.0	81.2	97.5	82.7	98.7	98.0	91.9	89.6
Yang et al. (2017)	✓	97.8	96.1	–	95.2	–	97.7	–	–	–
Cheng et al. (2017)		98.9	96.8	83.7	95.7	82.2	98.5	96.7	91.5	89.4
Liu et al. (2018a)	✓	–	–	83.6	–	84.4	–	–	91.5	90.8
Liu et al. (2018b)		97.3	96.1	89.4	96.8	87.1	98.1	97.5	94.7	94.0
Liu et al. (2018c)	✓	97.0	94.1	87.0	95.2	–	98.8	97.9	93.1	92.9
Cheng et al. (2018)		99.6	98.1	87.0	96.0	82.8	98.5	97.1	91.5	–
Bai et al. (2018)		99.5	97.9	88.3	96.6	87.5	98.7	97.9	94.6	94.4
Shi et al. (2018)		99.6	**98.8**	93.4	97.4[1]	89.5[1]	98.8	98.0	94.5	91.8
Luo et al. (2019)		97.9	96.2	91.2	96.6	88.3	98.7	97.8	95.0	92.4
Liao et al. (2019b)	✓	**99.8**	**98.8**	91.9	98.8	86.4	–	–	–	91.5
Li et al. (2019)		–	–	91.5	–	84.5	–	–	–	91.0
Zhan and Lu (2019)		99.6	98.8	93.3	97.4	90.2	–	–	–	91.3
Yang et al. (2019a)	✓	99.5	98.8	94.4	97.2	88.9	99.0	98.3	95.0	93.9
ASTER		99.1	97.9	93.5	98.0	88.6	98.8	98.0	94.7	92.0
+ Ours		99.1	98.0	94.0	98.3	90.0	98.8	98.1	95.6	93.3
+ Com.		99.6	98.7	95.4	98.9	92.7	**99.1**	**98.8**	**96.3**	94.8
ESIR		99.2	98.0	93.8	98.0	88.7	98.8	98.2	95.0	93.5
+ Ours		99.5	98.6	94.1	98.0	90.6	98.8	98.5	96.0	94.2
+ Com.		99.6	**98.8**	**95.6**	**99.4**	**92.9**	**99.1**	**98.8**	96.2	**96.0**

The best performing results are shown in bold font

[1]The result was corrected by the authors on https://github.com/bgshih/aster.

"50", "1k" and "0" are lexicon sizes. "Full" indicates the combined lexicon of all images in the benchmarks. "Add." means the method uses extra annotations, such as character-level bounding boxes and pixel-level annotations. "Com." is the proposed ensemble method that outputs prediction of either source or generated image with higher confidence score

4.4 Comparisons with Generation Methods

Recently, a large body of literature (Shi et al. 2018; Luo et al. 2019; Li et al. 2019; Yang et al. 2019a) has explored the use of stronger recognizers to tackle the complications in scene text recognition. However, there is little consideration of the quality of the source images. The background noise in the source image has not been addressed intensively before. To the best of our knowledge, our method may be the first image generation network that removes background noise and retains text content to benefit recognition performance. Although few literature proposed to address this issue stated above, we find several popular generation methods and perform comparisons under fair experimental conditions. A pre-trained recognizer used in the ablation study is adopted in the comparisons. The recognizer is then be fine-tuned on different kinds of generations.

First we use a popular binarization method, namely OTSU method (Otsu 1979), to separate the text content from the background noise by binarizing the source images. As shown in Fig. 10, we visualize the binarized images and find that single threshold value is not sufficiently robust to separate the foreground and background in scene text images, because the background noise usually follows multimodal distribution. Therefore, the recognition accuracy on the generation of OTSU method falls behind ours in Table 3.

Then, we compare our method with generation methods. Considering the high demand for data (pixel-level paired samples) of pixel-to-pixel GANs (Isola et al. 2017), we treat this kind of method as a potential solution when there is no restriction of data. Here, we study the CycleGAN[3] (Zhu et al. 2017). Before the training, we synthesize word-level clean images as target style samples. The results shown in Table 3 and Fig. 11 suggest that modeling a text string with multiple characters as a whole leads to poor retention of character details. The last two rows in Fig. 11 are failed generations, which indicate that the generator fails to model the relationships of the characters. In Table 3, the recognition accuracy on this kind of generation drops substantially.

Compared with previous methods, our method not only normalizes noisy backgrounds to a clean style, but also generates clear character patterns that tend to be an average style. The end-to-end training with the feedback mechanism benefits the recognition performance. We also show the effectiveness of the image rectification by integrating our method with advanced rectification modules (Shi et al. 2018; Zhan and Lu 2019). It can be seen that image rectifiers are still significant for improving recognition performance. Thus, different from irregular text shape, noisy background style is another challenge.

[3] The official implementation is available on https://github.com/junyanz/pytorch-CycleGAN-and-pix2pix

Fig. 11 Comparison between CycleGAN (Zhu et al. 2017) and our method

4.5 Integration with State-of-the-Art Recognizers

As our method is a meta-framework, it can be integrated with recent recognizers (Cheng et al. 2018; Shi et al. 2018; Luo et al. 2019; Li et al. 2019; Yang et al. 2019a) equipped with attention-based decoders (Bahdanau et al. 2015). We conduct experiments using representative methods, namely ASTER (Shi et al. 2018) and ESIR (Zhan and Lu 2019), to investigate the effectiveness of our framework. The reimplementation results are comparable with those in the paper. With respect to the dataset providing a lexicon, we choose the lexicon word under the metric of edit distance. The results of comparison with previous methods are shown in Tables 4 and 5. All the results of the previous methods are collected from their original papers. If a method uses extra annotations, such as character-level bounding boxes and pixel-level annotations, we indicate this with "Add.". For fair comparison, we perform a comparison with the method of (Li et al. 2019) by including the results of their model trained using synthetic data.

Using the strong baseline of ASTER, we first evaluate the contribution of our method on regular text as shown in Table 4. Although the baseline accuracy on these benchmarks is high, thus no much room for improvement, our method still achieves a notable improvement in lexicon-free prediction. For instance, it leads to accuracy increases of 1.4% on SVT and 1.3% on IC13. Some predictions corrected using our generations are shown in Fig. 12. Then, we reveal the superiority of our method by applying it to irregular text recognition. As shown in Table 5, our method significantly boosts the performance of ASTER by generating clean images. The

Table 5 Word accuracy on irregular benchmarks

Method	Add.	SVT-P			CUTE	IC15-S	IC15
		50	Full	0	0	0	0
Shi et al. (2016)		91.2	77.4	71.8	59.2	–	–
Liu et al. (2016)		94.3	83.6	73.5	–	–	–
Shi et al. (2017)		92.6	72.6	66.8	54.9	–	–
Yang et al. (2017)	✓	93.0	80.2	75.8	69.3	–	–
Cheng et al. (2017)		92.6	81.6	71.5	63.9	70.6	-
Liu et al. (2018a)	✓	–	–	–	–	–	60.0
Liu et al. (2018b)		–	–	73.9	62.5	–	–
Cheng et al. (2018)		94.0	83.7	73.0	76.8	–	68.2
Bai et al. (2018)		–	–	–	–	73.9	–
Shi et al. (2018)		–	–	78.5	79.5	76.1	–
Luo et al. (2019)		94.3	86.7	76.1	77.4	–	68.8
Liao et al. (2019b)	✓	–	–	–	79.9	–	–
Li et al. (2019)		–	–	76.4	83.3	–	69.2
Zhan and Lu (2019)		–	–	79.6	83.3	–	76.9
Yang et al. (2019a)	✓	–	–	80.8	87.5	–	78.7
ASTER		94.3	87.3	77.7	79.9	75.8	74.0
+ Ours		$95.0^{\uparrow 0.7}$	$90.1^{\uparrow 2.8}$	$81.6^{\uparrow 3.9}$	$85.1^{\uparrow 5.2}$	$80.1^{\uparrow 4.3}$	$78.1^{\uparrow 4.1}$
+ Com.		95.5	**92.2**	**85.4**	89.6	83.7	81.1
ESIR		94.3	87.3	79.8	83.7	79.3	77.1
+ Ours		$95.0^{\uparrow 0.7}$	$89.3^{\uparrow 2.0}$	$82.2^{\uparrow 2.4}$	$87.8^{\uparrow 4.1}$	$81.1^{\uparrow 1.8}$	$78.5^{\uparrow 1.4}$
+ Com.		**95.8**	91.5	85.1	**91.3**	**83.9**	**81.4**

The best performing results are shown in bold font
"50" and "0" are lexicon sizes. "Full" indicates the combined lexicon of all images in the benchmarks. "Add." means the method uses extra annotations, such as character-level bounding boxes and pixel-level annotations. "Com." is the proposed ensemble method that outputs prediction of either source or generated image with higher confidence score

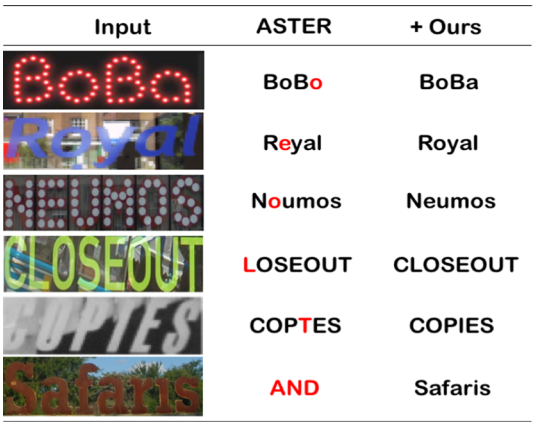

Fig. 12 Predictions corrected by our method

Fig. 13 Failure cases. Top: source images. Bottom: generated images

ASTER integrated with our approach outperforms the baseline by a wide margin on SVT (3.9%), CUTE (5.2%) and IC15 (4.3%). This suggests that our generator removes the background noise introduced by irregular shapes and further reduces difficulty of rectification and recognition. It is noteworthy that the ASTER with our method outperforms ESIR (Zhan and Lu 2019) that uses more rectification iterations (ASTER only rectifies the image once), which demonstrates the significant contribution of our method. The performance is even comparable with the state-of-the-art method (Yang et al. 2019a), which uses character-level geometric descriptors for supervision. Our method achieves a better trade-off between recognition performance and data requirement.

After that, our method is integrated with a different method ESIR to show the generalization. Based on the more advanced recognizer, our method can achieve further gains. For instance, the improvement is still notable on CUTE (4.1%). As a result, the performance of the ESIR is also significantly boost by our method.

Table 6 Word accuracy on testing datasets when we use a little more real training data

Method	Training data	Regular text				Irregular text			
		IIIT5K	SVT	IC03	IC13	SVT-P	CUTE	IC15-S	IC15
ASTER	Millions of synthetic data	93.5	88.6	94.7	92.0	77.7	79.9	75.8	74.0
	+ 50k Real data	94.5	90.4	95.0	92.9	79.4	88.9	84.1	81.3
+ Ours	Millions of synthetic data	95.4	92.7	**96.3**	94.8	85.4	89.6	83.7	81.1
	+ 50k Real data	**96.5**	**94.4**	**96.3**	**95.6**	**86.2**	**92.4**	**87.2**	**84.7**

The best performing results are shown in bold font

4.5.1 Upper Bound of GAN

We are further interested in the upper bound of our method. As our method is designed based on adversarial training, the limitations of the GAN cause some failure cases. As illustrated in Figure 13, the well-trained generator fails to generate character patterns on difficult samples, particularly when the source image is of low quality and the curvature of the text shape is too high. One possible reason is the mode-dropping phenomenon studied by (Bau et al. 2019). Another reason is the lingering gap observed by (Zhu et al. 2017) between the training supervision of paired and unpaired samples. To break this ceiling, one possible solution is to improve the synthesis engine and integrate various paired lifelike samples for training. This may lead to substantially more powerful generators, but heavily dependent on the development of synthesis engines.

Inspired by recent work (Shi et al. 2018; Liao et al. 2019a), it is possible to integrate several outputs of the system and choose the most possible one to achieve performance gain. Therefore, we proposed a simple yet effective method to address the issue stated above. The source image and the corresponding generated image are concatenated as a batch for network inference. Then, we choose the prediction with the higher confidence. As shown in the last row in Tables 4 and 5 (noted as "+Com."), this ensemble mechanism greatly boosts the system performance, which indicates that the source and generated images are complementary to each other.

4.6 More Accessible Data

In the experiments of comparing the proposed method with previous recognition methods, we have used only synthetic data for fair comparison. Here, we use the ASTER (Shi et al. 2018) to explore whether there is room for improvement in synthesis engines.

Following (Li et al. 2019), we collect publicly available real data for training. In contrast to synthetic data, real data is more costly to collect and annotate. Thus, there are only approximately 50k public real samples for training, whereas there are millions of synthetic data. As shown in Table 6, after we add the small real training set to the large synthetic one, the generality of both the baseline ASTER and our method is further boosted. This suggests that synthetic data is not sufficiently real and the model is still data-hungry.

In summary, our approach is able to make full use of real samples in the wild to further gain robustness, because of the training of our method requires only input images and the corresponding text labels. Note that our method trained using only synthetic data even outperforms the baseline trained using real data on most benchmarks, particularly on SVT-P (↑6.0%). Therefore, noisy background style normalization is a promising way to improve recognition performance.

4.7 Discussion

4.7.1 Generation in RGB Space or in Gray

The background noise and text content may be relative easier to be separated in RGB colorful images. To this end, we conduct an experiment to evaluate the influence of RGB color space. The target style samples are synthesized in random color to guide the generation in RGB space. As shown in Table 7, we find that the generation in RGB space cannot outperform the generation in gray. Therefore, the key issue of background normalization is not the color space, but the lack of pixel-level supervision. Without fine-grained guidance at pixel level, the generation is only guided by the attention mechanism of the recognizer to focus informative regions. Other noisy regions on the generated image are unreasonably neglected.

4.7.2 Alignment Issue on Long Text

To tackle the lack of paired training samples, we exploit the attention mechanism to extract every character for adversarial training. However, there exists misalignment problems of the attention mechanism (Cheng et al. 2017; Bai et al. 2018), especially on long text. (Cong et al. 2019) conducted a comprehensive study on the attention mechanism and found that the attention-based recognizers have poor performance on text sentence recognition. Thus, our method still have scope for performance gains on text sentence recognition. This is a

Table 7 Comparisons of generation in RGB space and in gray

Space	Regular text				Irregular text			
	IIIT5K	SVT	IC03	IC13	SVT-P	CUTE	IC15-S	IC15
Gray	**95.4**	**92.7**	**96.3**	94.8	**85.4**	**89.6**	**83.7**	**81.1**
RGB	95.2	92.3	**96.3**	**95.6**	85.3	88.9	**83.7**	**81.1**

The best performing results are shown in bold font

common issue of most attention mechanisms, which merits further study.

5 Conclusion

We have presented a novel framework for scene text recognition from a brand new perspective of separating text content from noisy background styles. The proposed method can greatly reduce recognition difficulty and thus boost the performance dramatically. Benefiting from the interactive joint training of an attention-based recognizer and a generative adversarial architecture, we extract character-level features for further adversarial training. Thus the discriminator focuses on informative regions and provides effective guidance for the generator. Moreover, the discriminator learns from the confusion of the recognizer and further effectively guides the generator. Thus, the generated patterns are clearer and easier to read. This feedback mechanism contributes to the generality of the generator. Our framework is end-to-end trainable, requiring only the text images and corresponding labels. Because of the elegant design, our method can be flexibly integrated with recent mainstream recognizers to achieve new state-of-the-art performance.

The proposed method is a successful attempt to solve the scene text recognition problem from the brand new perspective of image generation and style normalization, which has not been addressed intensively before. In the future, we plan to extend the proposed method to deal with end-to-end scene text recognition. How to extend our method to multiple general object recognition is also a topic of interest.

Acknowledgements This research was in part supported in part by NSFC (Grant No. 61936003), GD-NSF (No. 2017A030312006), the National Key Research and Development Program of China (No. 2016YFB1001405), and Fundamental Research Funds for Central Universities (x2dxD2190570).

References

Arjovsky, M., Chintala, S., Bottou, L. (2017) Wasserstein generative adversarial networks. In: *International Conference on Machine Learning (ICML)*, pp 214–223.

Azadi, S., Fisher, M., Kim, VG., Wang, Z., Shechtman, E., Darrell, T. (2018) Multi-content gan for few-shot font style transfer. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 7564–7573.

Bahdanau, D., Cho, K., Bengio, Y. (2015) Neural machine translation by jointly learning to align and translate. In: *International Conference on Learning Representations (ICLR)*.

Bai, F., Cheng, Z., Niu, Y., Pu, S., Zhou, S. (2018) Edit probability for scene text recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 1508–1516.

Bau, D., Zhu, JY., Wulff, J., Peebles, W., Strobelt, H., Zhou, B., Torralba, A. (2019) Seeing what a gan cannot generate. In: *IEEE International Conference on Computer Vision (ICCV)*, pp 4502–4511.

Berthelot, D., Schumm, T., Metz, L. (2017) BEGAN: boundary equilibrium generative adversarial networks. CoRR abs/1703.10717.

Bissacco, A., Cummins, M., Netzer, Y., Neven, H. (2013) PhotoOCR: Reading text in uncontrolled conditions. In: *IEEE International Conference on Computer Vision (ICCV)*, pp 785–792.

Casey, R. G., & Lecolinet, E. (1996). A survey of methods and strategies in character segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, *18*(7), 690–706.

Cheng, M. M., Liu, X. C., Wang, J., Lu, S. P., Lai, Y. K., & Rosin, P. L. (2019). Structure-Preserving Neural Style Transfer. *IEEE Transactions on Image Processing (TIP)*, *29*, 909–920.

Cheng, Z., Bai, F., Xu, Y., Zheng, G., Pu, S., Zhou, S. (2017) Focusing attention: Towards accurate text recognition in natural images. In: *IEEE International Conference on Computer Vision (ICCV)*, pp 5086–5094.

Cheng, Z., Xu, Y., Bai, F., Niu, Y., Pu, S., Zhou, S. (2018) AON: Towards arbitrarily-oriented text recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 5571–5579.

Cong, F., Hu, W., Huo, Q., Guo, L. (2019) A comparative study of attention-based encoder-decoder approaches to natural scene text recognition. In: *International Conference on Document Analysis and Recognition (ICDAR)*, pp 916–921.

Dalal, N., Triggs, B. (2005) Histograms of oriented gradients for human detection. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 886–893.

Fang, S., Xie, H., Chen, J., Tan, J., Zhang, Y. (2019) Learning to draw text in natural images with conditional adversarial networks. In: *International Joint Conferences on Artificial Intelligence (IJCAI)*.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y. (2014) Generative adversarial nets. *Neural Information Processing Systems (NeurIPS)*, pp 2672–2680.

Gordo, A. (2015) Supervised mid-level features for word image representation. In*: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 2956–2964.

Gupta, A., Vedaldi, A., Zisserman, A. (2016) Synthetic data for text localisation in natural images. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 2315–2324.

He, K., Zhang, X., Ren, S., Sun, J. (2016a) Deep residual learning for image recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 770–778.

He, P., Huang, W., Qiao, Y., Loy, CC., Tang, X. (2016b) Reading scene text in deep convolutional sequences. In: *AAAI Conference on Artificial Intelligence (AAAI)*, pp 3501–3508.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., & Hochreiter, S. (2017). Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Neural Information Processing Systems (NeurIPS)*, *30*, 6626–6637.

Isola, P., Zhu, JY., Zhou, T., Efros, AA. (2017) Image-to-image translation with conditional adversarial networks. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 5967–5976.

Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A. (2014a) Synthetic data and artificial neural networks for natural scene text recognition. Neural Information Processing Systems (NeurIPS) Deep Learning Workshop.

Jaderberg, M., Vedaldi, A., Zisserman, A. (2014b) Deep features for text spotting. In: *European Conference on Computer Vision (ECCV)*, pp 512–528.

Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A. (2015) Deep structured output learning for unconstrained text recognition. In: *International Conference on Learning Representations (ICLR)*.

Jaderberg, M., Simonyan, K., Vedaldi, A., & Zisserman, A. (2016). Reading text in the wild with convolutional neural networks. *International Journal of Computer Vision (IJCV)*, *116*(1), 1–20.

Jing, Y., Yang, Y., Feng, Z., Ye, J., Yu, Y., & Song, M. (2019). Neural style transfer: A review. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, *26*(11), 3365–3385.

Johnson, J., Alahi, A., Fei-Fei, L. (2016) Perceptual losses for real-time style transfer and super-resolution. In: *European Conference on Computer Vision (ECCV)*, pp 694–711.

Karatzas, D., Shafait, F., Uchida, S., Iwamura, M, i Bigorda, LG., Mestre, SR., Mas, J., Mota, DF., Almazan, JA., De Las Heras, LP. (2013) ICDAR 2013 robust reading competition. In*: International Conference on Document Analysis and Recognition (ICDAR)*, pp 1484–1493.

Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., Matas, J., Neumann, L., Chandrasekhar, VR., Lu, S., et al. (2015) ICDAR 2015 competition on robust reading. In: *International Conference on Document Analysis and Recognition (ICDAR)*, pp 1156–1160.

Kingma, D., Ba, L., et al. (2015) Adam: A method for stochastic optimization. In: *International Conference on Learning Representations (ICLR)*.

Lee, CY., Osindero, S. (2016) Recursive recurrent nets with attention modeling for OCR in the wild. In*: IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 2231–2239.

Li,H., Wang, P., Shen, C., Zhang, G. (2019) Show, attend and read: A simple and strong baseline for irregular text recognition. In: *AAAI Conference on Artificial Intelligence (AAAI)*.

Liao, M., Lyu, P., He, M., Yao, C., Wu, W., Bai, X. (2019a) Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.

Liao, M., Zhang, J., Wan, Z., Xie, F., Liang, J., Lyu, P., Yao, C., Bai, X. (2019b) Scene text recognition from two-dimensional perspective. In: *AAAI Conference on Artificial Intelligence (AAAI)*.

Lin, TY., Dollar, P., Girshick, R., He, K., Hariharan, B., Belongie, S. (2017) Feature pyramid networks for object detection. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 2117–2125.

Liu, W., Chen, C., Wong, KYK., Su, Z., Han, J. (2016) STAR-Net: A spatial attention residue network for scene text recognition. In: *British Machine Vision Conference (BMVC)*, pp 7–7.

Liu, W., Chen, C., Wong, KYK. (2018a) Char-net: A character-aware neural network for distorted scene text recognition. In: *AAAI Conference on Artificial Intelligence (AAAI)*.

Liu, Y., Wang, Z., Jin, H., Wassell, I. (2018b) Synthetically supervised feature learning for scene text recognition. In: *European Conference on Computer Vision (ECCV)*, pp 435–451.

Liu, Z., Li, Y., Ren ,F., Goh, WL., Yu, H. (2018c) SqueezedText: A real-time scene text recognition by binary convolutional encoder-decoder network. In: *AAAI Conference on Artificial Intelligence (AAAI)*.

Lucas, SM., Panaretos, A., Sosa, L., Tang, A., Wong, S., Young, R. (2003) ICDAR 2003 robust reading competitions. In*: International Conference on Document Analysis and Recognition (ICDAR)*, pp 682–687.

Luo, C., Jin, L., & Sun, Z. (2019). MORAN: A multi-object rectified attention network for scene text recognition. *Pattern Recognition*, *90*, 109–118.

Mao, X., Li, Q., Xie, H., Lau, RY., Wang, Z., Smolley, SP. (2017). Least squares generative adversarial networks. In: *IEEE International Conference on Computer Vision (ICCV)*, pp 2813–2821.

Mishra, A., Alahari, K., Jawahar, C. (2012). Scene text recognition using higher order language priors. In: *British Machine Vision Conference (BMVC)*, pp 1–11.

Neumann, L., Matas, J. (2012). Real-time scene text localization and recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 3538–3545.

Odena, A., Olah, C., Shlens, J. (2017). Conditional image synthesis with auxiliary classifier GANs. In: *International Conference on Machine Learning (ICML)*, pp 2642–2651.

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics (TSMC)*, *9*(1), 62–66.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A. (2017). Automatic differentiation in PyTorch. Neural Information Processing Systems (NeurIPS) Autodiff Workshop.

Quy Phan, T., Shivakumara, P., Tian, S., Lim Tan, C. (2013) Recognizing text with perspective distortion in natural scenes. In: *IEEE International Conference on Computer Vision (ICCV)*, pp 569–576.

Risnumawan, A., Shivakumara, P., Chan, C. S., & Tan, C. L. (2014). A robust arbitrary text detection system for natural scene images. *Expert Systems with Applications*, *41*(18), 8027–8048.

Rodriguez-Serrano, J. A., Gordo, A., & Perronnin, F. (2015). Label embedding: A frugal baseline for text recognition. *International Journal of Computer Vision (IJCV)*, *113*(3), 193–207.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X. (2016). Improved techniques for training GANs. *Neural Information Processing Systems (NeurIPS)*, pp 2234–2242.

Shi, B., Wang, X., Lyu, P., Yao, C., Bai, X. (2016). Robust scene text recognition with automatic rectification. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 4168–4176.

Shi, B., Bai, X., & Yao, C. (2017). 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.

Shi, B., Yang, M., Wang, X., Lyu, P., Yao, C., & Bai, X. (2018). ASTER: An attentional scene text recognizer with flexible rectification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *41*, 2035.

Su, B., Lu, S. (2014). Accurate scene text recognition based on recurrent neural network. In: *Asian Conference on Computer Vision (ACCV)*, pp 35–48.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. *Neural Information Processing Systems (NeurIPS)*, *2*, 3104–3112.

Wang, K., Babenko, B., Belongie, S. (2011). End-to-end scene text recognition. In: *IEEE International Conference on Computer Vision (ICCV)*, pp 1457–1464.

Wang, T., Wu, DJ., Coates, A., Ng, AY. (2012). End-to-end text recognition with convolutional neural networks. In: *IEEE International Conference on Pattern Recognition (ICPR)*, pp 3304–3308.

Wu, L., Zhang, C., Liu, J., Han, J., Liu, J., Ding, E., Bai, X. (2019). Editing text in the wild. In: *ACM International Conference on Multimedia (ACM MM)*, pp 1500–1508.

Yang, M., Guan, Y., Liao, M., He, X., Bian, K., Bai, S., Yao, C., Bai, X. (2019a). Symmetry-constrained rectification network for scene text recognition. In: *IEEE International Conference on Computer Vision (ICCV)*, pp 9147–9156.

Yang, S., Wang, Z., Wang, Z., Xu, N., Liu, J., Guo, Z. (2019b). Controllable artistic text style transfer via shape-matching GAN. In: *IEEE International Conference on Computer Vision (ICCV)*.

Yang, X., He, D., Zhou, Z., Kifer, D., Giles, CL. (2017). Learning to read irregular text with attention mechanisms. In: *International Joint Conferences on Artificial Intelligence (IJCAI)*, pp 3280–3286.

Yao, C., Bai, X., Shi, B., Liu, W. (2014). Strokelets: A learned multi-scale representation for scene text recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp 4042–4049.

Ye, Q., & Doermann, D. (2015). Text detection and recognition in imagery: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, *37*(7), 1480–1500

Zhan, F., Lu, S. (2019). ESIR: End-to-end scene text recognition via iterative image rectification. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 2059–2068. .

Zhang, Y., Nie, S., Liu, W., Xu, X., Zhang, D., Shen, HT. (2019). Sequence-to-sequence domain adaptation network for robust text image recognition. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 2740–2749.

Zhu, JY., Park, T., Isola, P., Efros, AA. (2017). Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 2242–2251.

Zhu, Y., Yao, C., & Bai, X. (2016). Scene text detection and recognition: Recent advances and future trends. *Frontiers of Computer Science*, *10*(1), 19–36.