SGBANet: Semantic GAN and Balanced Attention Network for Arbitrarily Oriented Scene Text Recognition

Dajian Zhong[1], Shujing Lyu[1]*, Palaiahnakote Shivakumara[2], Bing Yin[3], Jiajia Wu[3], Umapada Pal[4], and Yue Lu[1]

[1] Shanghai Key Laboratory of Multidimensional Information Processing, East China Normal University, Shanghai, China
djzhong@stu.ecnu.edu.cn,{sjlv, ylu}@cs.ecnu.edu.cn
[2] Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur, Malaysia
shiva@um.edu.my
[3] iFLYTEK Research, iFLYTEK, Hefei, China
{bingyin, jjwu}@iflytek.com
[4] CVPR Unit, Indian Statistical Institute, Kolkata, India
umapada@isical.ac.in

Abstract. Scene text recognition is a challenging task due to the complex backgrounds and diverse variations of text instances. In this paper, we propose a novel Semantic GAN and Balanced Attention Network (SGBANet) to recognize the texts in scene images. The proposed method first generates the simple semantic feature using Semantic GAN and then recognizes the scene text with the Balanced Attention Module. The Semantic GAN aims to align the semantic feature distribution between the support domain and target domain. Different from the conventional image-to-image translation methods that perform at the image level, the Semantic GAN performs the generation and discrimination on the semantic level with the Semantic Generator Module (SGM) and Semantic Discriminator Module (SDM). For target images (scene text images), the Semantic Generator Module generates simple semantic features that share the same feature distribution with support images (clear text images). The Semantic Discriminator Module is used to distinguish the semantic features between the support domain and target domain. In addition, a Balanced Attention Module is designed to alleviate the problem of attention drift. The Balanced Attention Module first learns a balancing parameter based on the visual glimpse vector and semantic glimpse vector, and then performs the balancing operation for obtaining a balanced glimpse vector. Experiments on six benchmarks, including regular datasets, i.e., IIIT5K, SVT, ICDAR2013, and irregular datasets, i.e., ICDAR2015, SVTP, CUTE80, validate the effectiveness of our proposed method.

Keywords: Semantic GAN · Semantic Generator · Semantic Discriminator · Balanced Attention · Scene Text Recognition

* Corresponding author

1 Introduction

Scene text recognition has attracted growing research attention as a great need in real-life applications such as visual question answering [6], license plate recognition [54], driver-less vehicles [24]. Although previous works [1,3,21,26,56,5] have achieved great success, scene text recognition is still a challenging task because scene text images can be affected by multiple factors, such as complex background, arbitrarily shaped text, non-uniform spacing, etc.

With the development of Convolutional Neural Network (CNN) [18,20] and attention-based recognizer [9,25], text instances with simple background can be well recognized. Although, several models are proposed for addressing challenges of scene text recognition especially for regular text images, achieving good performance for arbitrarily oriented and irregularly shaped scene images is still considered as an open challenge. The key reason is that every character in the above two cases has a different orientation and shape in contrast to regular text. It can be observed from Fig. 1 that the existing methods [4,11] do not recognize the characters correctly for arbitrarily shaped text and text with complex backgrounds. Therefore, designing a robust method for recognizing arbitrarily shaped text is still a challenging task that remains to be solved.

Since it is easy to recognize clear images while those complex images are hard to recognize, there comes a question is it possible to transform complex images into simple ones? We can get inspiration from the existing works. Wang et al. [46] proposed a scene text dataset that contains paired real low-resolution and high-resolution images in the wild. Then the TSRNet is developed to reconstruct the high-resolution images for scene text images and recognize them. In this situation, the complex images (low-resolution images) can be transformed into clear images (high-resolution images). However, paired scene text images are needed in this method and it is hard to obtain these images and annotations. Besides, since the method can only deal with low-resolution images, scene text images with complex backgrounds are hard to recognize. Luo et al. [31] introduced the generative adversarial network [13,53] to remove the backgrounds while retaining the text content and then recognize the new reconstructed images. It satisfies the need for more scenarios, but it combines two networks into one. One network is for generating new images and another one is to recognize them. The method is time-consuming. The above mentioned methods used GAN for scene text recognition, but none of them used GAN as semantic generator and discriminator for scene text recognition. It motivated us to propose a novel model that explores GAN for extracting semantic features.

In this work, we introduce the Semantic GAN which consists of a Semantic Generator Module and a Semantic Discriminator Module. The Semantic Generator Module directly generates the semantic features upon the convolutional features for complex scene text images. Then the Semantic Discriminator Module distinguishes the semantic features between clear images and complex scene text images. In this way, the semantic feature distribution can be aligned and the generated features share more characteristics with those clear images and

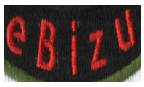
Input				
GT	START	EBIZU	FRIKKIE	SAFARIS
Baek et al. [4]	STARI	BBIZU	FAR?KING	SABAttS
Fang et al. [11]	START	BBZZU	FRIKKLE	SAFA?NS

Fig. 1. Challenges of scene text recognition. GT represents the GroundTruth. Misrecognized characters are marked in red and '?' represents the missed character.

thus can be easier to recognize. In this framework, paired images and an extra image-to-image translation network are not needed.

Besides, the attention drift problem is first raised in [8]. They proposed the FANet method to draw back the drifted attention with a focusing attention mechanism. It first computes the attention center of each predicted label and then generates the probability distributions on the attention regions. Though the FANet method works well, the complex operations cost huge computations. In our work, we propose a Balanced Attention Module, which directly utilizes the convolutional features to correct the attention weights on the semantic features. The Balanced Attention Module significantly alleviates the problem of attention drift.

The Semantic GAN is capable of aligning the distribution between the support domain and target domain and the Balanced Attention is designed to address the problem of attention drift, thus alleviating the challenges. Our main contributions can be summarized as follows:

1. We propose a novel text recognition network, called SGBANet, which consists of Semantic GAN and Balanced Attention Module. The proposed method integrates GAN into the recognition network without introducing an extra image-to-image translation network, thus reducing the time complexity.
2. We introduce the Semantic GAN, which consists of a Semantic Generator Module and a Semantic Discriminator Module. To the best of our knowledge, it is the first time to use the Semantic Generator and Discriminator to generate semantic features for overcoming the challenge of scene text recognition.
3. We design a Balanced Attention Module, which automatically learns a balancing parameter based on the convolutional and semantic features. It corrects the attention weights on the semantic features and draws back the drifted attention to some extent.

2 Related Work

2.1 Scene Text Recognition

Scene text recognition aims to decode a character sequence from the scene images. The methods can be categorized into language-free methods and language-

based methods. The language-free methods usually utilize convolutional features without consideration of the character dependency, such as segmentation-based methods and CTC-based methods. Segmentation-based methods [33,50,42] first segment the character regions and then recognize each character region to form the final character sequence. CTC-based methods [12,17,16] first extract visual features through CNN and then train with RNN and CTC loss to find the most possible combination. However, these methods lack linguistic information and thus easily miss one or two characters.

The language-based methods mainly employ the attention mechanism [38]. The encoder-decoder architecture makes use of linguistic information and character dependency. To boost the performance, some methods focus on learning a new feature representation. For example, [1] proposed a contrastive learning algorithm that first divides each feature map into a sequence of individual elements and performs the contrastive loss [15]. Then the learned representation features are fed to the recognizer. Yan et al. [51] proposed a primitive representation learning method that aims to exploit intrinsic representations of scene text images. Others may focus on integrating the rectification module [30,41,55,52] to reconstruct normal images for those irregular images. Then the reconstructed images are fed to the encoder-decoder module for further recognition. However, scene text usually has a variety of shapes and sizes. It is difficult for the rectification module to transform all the irregular text instances into regular ones. Since current attention-based methods suffer from the attention drift problem [8,27], directly decoding upon the convolutional features or linguistic features will degrade the recognition performance. Inspired by [44] which generates the character center masks to help focus attention on the right position. In this work, we introduce the Balanced Attention Module, which learns a balancing parameter based on the convolutional and semantic features and draws back the drifted attention. Through the method, the attention drift problem can be alleviated to some extent.

2.2 Generative Adversarial Network

With the development of GANs [34,59,36,58], image-to-image translation has achieved great success. Several methods integrate GANs to generate clear text images for scene text images [10]. This inspires the researchers to combine the GAN and recognition network. Thus, recent recognition methods focus on integrating the adversarial learning concept into the recognition network. For example, [57] introduced a gated attention similarity (GAS) unit to adaptively focus on aligning the distribution of the source and target sequence data. Luo et al. [31] introduced the generative adversarial network to generate a simple image without the complex background for each scene text image. However, it costs much computation if we first generate the required text image and then feed them to a recognition network. Since the two networks both contain huge convolution operations that can result in huge computation, our goal is to design a novel Semantic GAN to align the semantic feature distribution where the Semantic Generator Module directly generates the high-level semantic features and

the Semantic Discriminator Module distinguishes between the support domain and target domain.

3 Methodology

In this section, details of SGBANet are presented. We first describe the overall architecture of the proposed method. Then we dissect the Generator module and the Discriminator module. Finally, the Balanced Attention module is introduced.

3.1 Overall Architecture

As can be seen in Fig. 2, the overall architecture of the proposed method consists of the CNN Encoder, the Semantic GAN and the Balanced Attention Module. The CNN Encoder is used to extract the basic convolutional features. The Semantic GAN consists of the Semantic Generator Module(SGM) and the Semantic Discriminator Module (SDM). The Semantic Generator Module is applied to directly generate the high-level semantic features that can be easier to recognize and the Semantic Discriminator Module aims to distinguish between the support semantic features and target semantic features. After the adversarial learning, the semantic feature distribution between the source and target images can be aligned. The Balanced Attention Module is designed to draw back the drifted attention by learning a balancing parameter based on the convolutional and semantic features. Then the balancing operation can be performed to get the balanced glimpse vector.

There are two inputs for the whole architecture. One is the support image , which represents the simple image containing pure text instance. Another is the target image , which contains text instances with complex backgrounds. Firstly, the two images are fed into the CNN Encoder and visual feature map $_v$ and $_v$ are obtained. Secondly, they are fed to a Bi-LSTM layer and the Semantic Generator Module, respectively, and $_s$ and $_s$ are obtained. Then they are fed to the Semantic Discriminator Module. The Semantic Generator Module and the Semantic Discriminator Module contribute to generating simple semantic features. For the Semantic Generator Module, it generates the semantic feature that the discriminator can't distinguish from the support domain and target domain. At the same time, the Semantic Discriminator Module aims to distinguish them correctly. After the training of Semantic GAN, the Semantic Generator Module successfully generates the simple semantic feature for scene text images, which shares the same feature distribution with those of clear images. And $_v$ and $_s$ together with $_v$ and $_s$ are fed to the Balanced Attention Module for the final recognition.

3.2 CNN Encoder

Similar to [7], we take ResNet-based structure as the backbone network to extract basic visual features. We remove the CBAM module [47] and assemble an extra

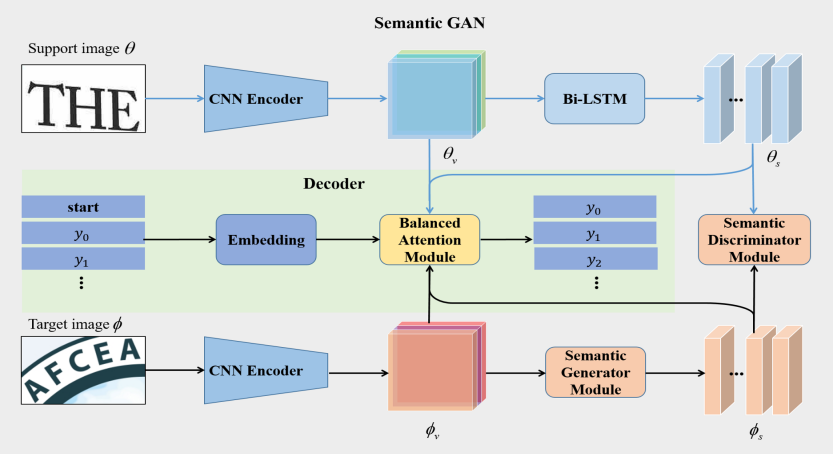


Fig. 2. Architecture of the proposed SGBANet. The support image θ and target image ϕ are fed to the CNN Encoders, which share the same parameters. The visual convolutional features θ_v and ϕ_v are fed to a Bi-LSTM and the Generator Module, respectively. Moreover, the semantic features θ_s and ϕ_s are fed to the Semantic Discriminator and Balanced Attention for discrimination and recognition. The Embedding module is to make a one-hot embedding on the input labels.

down-sampling layer. As a result, the size of $_v$ and $_v$ is restored to $\frac{H}{8}$ $\frac{W}{8}$ C, where H, W and C represent the height, width and channel of the input images, respectively. Since the size of text instances varies and there is no extra image-to-image network, FPN [28] is assembled to make the CNN Encoder capable of extracting different levels of features. The input support image and target image can be fed to a shared CNN Encoder for saving memory and computation cost.

3.3 Semantic GAN

The Semantic GAN consists of a Semantic Generator Module and a Semantic Discriminator Module. The Semantic Generator Module directly generates the semantic features for complex scene text images. A Bi-LSTM module is used to extract the semantic features of clear images. Then the Semantic Discriminator Module distinguishes the semantic features between support and target domain. In this way, the semantic feature distribution between the source and target images can be aligned and thus can be easier to recognize.

Semantic Generator Module For the support image , the visual convolutional feature $_v$ is directly fed to a Bi-LSTM layer and the semantic feature $_s$ is obtained. As for the target image , the visual convolutional feature $_v$ is fed to the Semantic Generator Module for generating the semantic feature $_s$ so that the new generated feature shares the same feature distribution with those of support images. Different from the structure of the conventional generator that

stacks the convolutional layers, the main components of the Semantic Generator Module are Bi-LSTM layers and Fully connected (FC) layers. The generator consists of two basic units and each unit comprises a Bi-LSTM layer followed by two FC layers. Given the input visual convolutional feature $_v$, the Semantic Generator Module generates a new semantic feature $_s$ without encoding the background information. The size of $_s$ is the same as $_s$, which is $\frac{W}{8}$ 256.

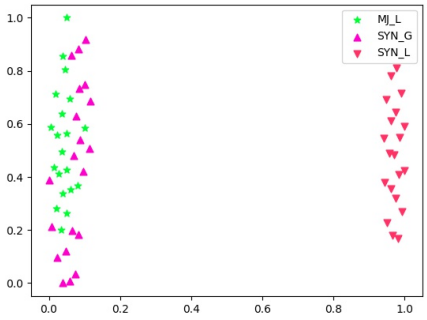


Fig. 3. T-SNE plot of learned feature representations. 'MJ_L' denotes learned features from Bi-LSTM with sampled MJSynth images. 'SYN_G' and 'SYN_L' denote generated semantic features and learned features from the Semantic Generator Module and Bi-LSTM using sampled SynthText images, respectively.

Semantic Discriminator Module The Semantic Discriminator Module is used to discriminate the semantic features $_s$ and $_s$ and the architecture contains no convolutional layers. The Discriminator consists of two basic units and an FC layer. Each unit comprises a Bi-LSTM layer and an FC layer. The first two units are used to reduce the size of input features and the last FC layer is to do the final discrimination. Given the semantic features $_s$ and $_s$, the discriminator distinguishes them between the support image and target image. The output size of the discriminator is 1. To illustrate the effectiveness of the proposed Semantic GAN, we randomly sample 20 MJSynth images and 20 SynthText images, which represent the clear text images and complex scene text images respectively, and visualize the learned features. As can be seen from Fig. 3, the Semantic Generator Module successfully generates the semantic features for the SynthText images that share the same domain with those of MJSynth images.

3.4 Balanced Attention Module

Since convolutional features contain positions of characters, we can utilize the convolutional features to correct the attention weights on the semantic features.

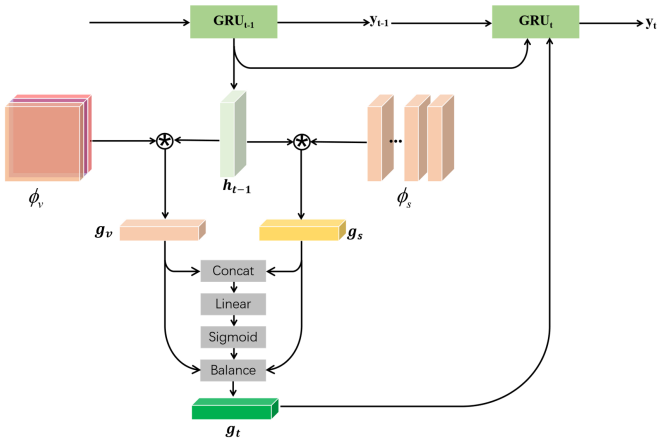


Fig. 4. Architecture of Balanced Attention Module. The 'Balance' operation means making a balance between the two glimpse vectors and it is defined by Equation. (3) and Equation. (4).

The Balanced Attention Module is designed to draw back the drifted attention and recognize the character sequence. It works iteratively for T steps, producing a target sequence of length T, denoted by $(y_1, y_2, ..., y_T)$. As can be seen in Fig. 4, at time step t, the output y_t for target images is defined by:

$$y_t = Softmax(W_{out}h_t + b_{out}) \tag{1}$$

where W_{out} and b_{out} are trainable parameters and h_t is the hidden state at the time step t. h_t is updated as follows:

$$h_t = GRU(y_{t\ \ 1}; h_{t\ \ 1}; g_t) \tag{2}$$

where $y_{t\ \ 1}$ is the output at time step $t\ \ \ 1$ and g_t represents the glimpse vector defined by:

$$g_t = \ \ g_v + (1\ \ \ \ )g_s\ ; \quad 2\ R^n \tag{3}$$

where is learnable parameters. g_v and g_s are internal glimpse vectors. They are computed as follows:

$$= Sigmoid(W_\lambda cat(g_v; g_s) + b_\lambda) \tag{4}$$

$$g_v = \sum_{i=1} \ \ \ \ _{t,i}^{v} \ \ \ \ v,i \tag{5}$$

$$g_s = \sum_{i=1} \ \ \ \ _{t,i}^{s} \ \ \ \ s,i \tag{6}$$

where α_t^v and α_t^s are attention weights, which are defined by:

$$\alpha_{t,i}^v = \frac{\exp(e_{t,i}^v)}{\sum_{j=1}^{P} \exp(e_{t,j}^v)} \tag{7}$$

$$\alpha_{t,i}^s = \frac{\exp(e_{t,i}^s)}{\sum_{j=1}^{P} \exp(e_{t,j}^s)} \tag{8}$$

$$e_{t,i}^v = W_c \quad \tanh(W_h h_{t\ 1} + W_v \quad AP(\ _v) + b_v) \tag{9}$$

$$e_{t,i}^s = W_c \quad \tanh(W_h h_{t\ 1} + W_s \quad _s + b_s) \tag{10}$$

where W_c, W_h, W_v, W_s, b_v and b_s are trainable parameters and AP represents average pooling on the height of $_v$. Thus the size of $AP(\ _v)$ is restored into $\frac{W}{8}$ C (C is set to 256), which is the same as that of $_s$. For the recognition of support images, the $_v$ and $_s$ can be replaced by $_v$ and $_s$, and the above operations can be performed in the same way. As can be seen in Fig. 5, the balanced glimpse vector learns an accurate attention weight and guide the recognition of characters.

Fig. 5. An example of the balanced glimpse vector. The first image is the input and the others are attention maps under different time steps.

In the inference stage, since the input image can be a clear image or a complex scene text image, we feed the convolutional feature $_v$ to the Bi-LSTM and Semantic Generator Module both. Then the two learned features are fed to the Balanced Attention Module for recognition and two character sequences are produced. Finally, we choose the character sequence with a higher confidence score as the final result and the other one will be discarded.

3.5 Training

The objective function L of our proposed SGBANet consists of two parts: the recognition loss L_R and the GAN loss L_G, as defined by:

$$L = L_R + \quad L_G \tag{11}$$

where is the coefficient used to balance the importance of the recognition network and GAN network. We set to 0 during the pre-training stage and 1 during the fine-tuning stage. The GAN loss is defined by:

$$L_G = L_g + \quad L_d \tag{12}$$

$$L_d = E[\max(0; 1 \quad D(\ _s))] + E[\max(0; 1 + D(G(\ _v))] \tag{13}$$

$$L_g = \quad E[D(G(\ _v)] \tag{14}$$

where $G(\ )$ and $D(\ )$ denote the Semantic Generator Module and the Semantic Discriminator Module. $_s$ and $_v$ are semantic feature and visual convolutional feature of support image and target image, respectively.

As it is hard to train the recognition network combined with GAN, joint training of the recognition network and GAN will easily cause the instability of recognition loss. To make the whole network stable and robust enough, we have tried several loss functions for the GAN network, such as the original GAN loss [13], Wasserstein GAN Loss [2], and Hinge loss [48]. Additionally, we have tried several architectures of the Semantic Generator Module and the Semantic Discriminator Module, such as integrating convolution layers, LSTM layers and FC layers. As a result, we find that the Hinge loss, together with the architecture reported in Section 3.3 and Section 3.3, gets a robust training.

4 Experiments

4.1 Datasets

We train the proposed method SGBANet with the public available synthetic datasets, *i:e:* SynthText [14] and MJSynth [19] without fine-tuning on individual scene text datasets and lexicons. We evaluate the performance on the six widely used benchmarks, including three regular text datasets (IIIT5K, ICDAR2013, SVT), and three irregular text datasets (ICDAR2015, SVTP, CUTE80). Details of the datasets are as follows.

IIIT5K [35] contains 3000 test images cropped from natural scene images. Most of the text instances are regular with the horizontal layout.

ICDAR2013 (IC13) [23] has 1015 test images. The dataset only contains horizontal text instances.

Street View Text (SVT) [43] consists of 647 word patches cropped from Google Street View for testing. This dataset contains blur, noise and low-resolution text images.

ICDAR2015 (IC15) [22] contains 2077 test images collected by Google Glasses. Most of the text instances are irregular (oriented, perspective or curved). We discard the vertical text images, which results in 2002 test images.

Street View Text Perspective (SVTP) [37] contains 645 cropped images from side view angle snapshots in Google Street View. Most of the images are perspective distorted.

CUTE80 [39] contains 288 images cropped from high-resolution scene text images. Most of the images contain curved text.

4.2 Implementation Details

The proposed method is implemented by using PyTorch. All the experiments are conducted on an NVIDIA Tesla V100 GPU with 32 GB memory. In our experiments, all the input images are rescaled to the size of 64 256 with the aspect ratio preserved. The character set contains 64 classes, including 10 digits, 52 case-sensitive letters, the *SOS* token and the *EOS* token. The maximum sequence length is set to 32. AdaDelta is chosen as the optimizer and the batch size is set to 185.

The training process of the SGBANet is divided into two stages: the pre-training stage and the fine-tuning stage. In the pre-training stage, as we described in Equation. (11), we set the to 0. Thus, we train the network on the SynthText and MJSynth from scratch for 2 epochs. Since images in MJSynth contain only pure text instances and images in SynthText contain complex backgrounds, images in the MJSynth and SynthText can be considered as support images and target images, respectively. In the fine-tuning stage, is set to 1, and we jointly train the whole network for another 5 epochs.

4.3 Comparison with State-of-the-art Approaches

We evaluate our method on the aforementioned six benchmark datasets and compare it with those state-of-the-art methods. For a fair comparison, all the methods are trained on the SynthText and MJSynth without fine-tuning on individual scene text datasets and no lexicons are used during the inference stage. Table 1 presents the details of the comparison results. It has been shown that our proposed SGBANet achieves the best on four datasets including IIIT5K, ICDAR2013, ICDAR2015 and SVTP, and achieves competitive performance on two datasets including SVT and CUTE80. Note that, for training the Semantic GAN, we have to divide the MJSynth and SynthText into support images and target images, respectively. Only the SynthText dataset is trained on the Semantic Generator Module. As a result, the proposed method doesn't achieve the best on the SVT and CUTE80. If we compare our method with SSDAN[57], which exploits the domain adaptation network, we can find that there is a large performance gap between the two methods on the regular text images. Qualitative results for text recognition of the existing methods and our proposed method are presented in Fig. 6. The existing methods [3,4] do not recognize the characters correctly, while the proposed method reports correct recognition results. In addition, We have calculated the average FPS of the proposed and existing methods [3,4] for all the 8 datasets and the results are 6.76, 6.68 and 7.58 for the methods [3], [4] and SGBANet, respectively. This shows that our method is faster than the existing methods. The key reason is that the existing methods perform rectification before prediction while the proposed method does not.

Table 1. Comparison with state-of-the-art methods.

Methods	Regular Text			Irregular Text		
	IIIT5K	SVT	IC13	IC15	SVTP	CUTE80
Liao et al. [27]	92.0	82.1	91.4	-	-	-
Baek et al. [3]	87.9	87.5	92.3	71.8	79.2	74.0
Luo et al. [30]	91.2	88.3	92.4	68.8	76.1	77.4
Li et al. [25]	91.5	84.5	91.0	69.2	76.4	83.3
Shi et al. [40]	81.2	82.7	89.6	-	-	-
Zhang et al. [57]	83.8	84.5	91.8	-	-	-
Xie et al. [49]	-	-	-	68.9	70.1	82.6
Yue et al. [54]	95.3	88.1	94.8	77.1	79.5	**90.3**
Wang et al. [44]	90.5	82.2	-	-	-	83.3
Long et al. [29]	93.7	88.9	92.4	76.6	78.8	86.8
Wang et al. [45]	94.3	**89.2**	93.9	74.5	80.0	84.4
Luo et al. [32]	-	-	-	76.1	79.2	84.4
Aberdam et al. [1]	82.9	87.9	-	-	-	-
Baek et al. [4]	92.1	88.9	93.1	74.7	79.5	78.2
SGBANet	**95.4**	89.1	**95.1**	**78.4**	**83.1**	88.2

Input				
GT	BALLACK	LEST	CHELSEA	COMPANY
Baek et al. [3]	BALLACK	LEST	CHELSE?	COMPERS
Baek et al. [4]	BALLACH	? EST	CHELSEA	COMPANY
SGBANet	BALLACK	LEST	CHELSEA	COMPANY

Fig. 6. Qualitative results for text recognition of the existing methods and our proposed method.

4.4 Ablation Study

The proposed method mainly consists of three modules: the Semantic Generator Module, the Semantic Discriminator Module and the Balanced Attention Module. To demonstrate the effectiveness of the proposed method, we will first evaluate the performance of individual components and then make a further discussion on the Balance Attention Module.

The effectiveness of the individual components. Since the Semantic GAN and Balanced Attention Module are two core components of the proposed method, we combine the baseline method with the individual components and conduct experiments on the six benchmarks. The proposed method without considering Semantic GAN and Balanced Attention Module is considered as the baseline method. The baseline method combined with Semantic GAN is considered as 'Baseline+SGAN'. In the same way, the baseline method combined with the two components is considered as 'Baseline+SGAN+BA'. It is observed from Table 2, 'Baseline+SGAN+BA' outperforms the 'Baseline' method on all the six benchmarks. It indicates that the Semantic GAN successfully generates simple semantic features that are easier to recognize. In the same way, 'Baseline+SGAN+BA' outperforms 'Baseline+SGAN', which attests to the effectiveness of the proposed Balance Attention Module.

Table 2. Effectiveness of the key components of the proposed method. 'SGAN' and 'BA' represent the Semantic GAN and Balance Attention.

Datasets	IIIT5K	SVT	IC13	IC15	SVTP	CUTE80
Baseline	90.1	84.5	91.8	70.5	76.8	80.0
Baseline+SGAN	92.1	86.6	91.1	72.5	77.1	80.5
Baseline+SGAN+BA	**95.4**	**89.1**	**95.1**	**78.4**	**83.1**	**88.2**

Discussions about the Balance Attention Module. The Balance Attention is the core component of our proposed method. The key step of the Balance Attention is the balancing operation, which learns a dynamic balancing parameter and makes a balance between the two glimpse vectors. The details of our balancing operation are defined in Equation. (3) and Equation. (4). We discuss the effectiveness of the proposed balancing operation and other balancing operations. In the experiment, we evaluate four balancing operations, including 'Single', 'Add', 'CL', and our 'Balance' operation. The 'Single' operation only uses the glimpse vector g_v. The 'Add' operation directly makes an add operation on g_v and g_s. The 'CL' operation generates a new glimpse vector by first concatenating the two glimpse vectors and then feeding it to an FC layer. 'Balance'

is our proposed balancing operation. As can be seen in Table 3, the 'Single' operation gets the worst performance on the six benchmarks. Text instances with complex backgrounds and arbitrary shapes are challenging for the 'Single' operation. 'Add' and 'CL' operations improve the performance and achieve the best on SVT and CUTE80, respectively. Our 'Balance' operation further improves the performance and achieves the best performance on four datasets including IIIT5k, IC13, IC15, and SVTP, and there is only a gap of 0.3 on the other two datasets. Thus, our balancing operation gets the overall best performance.

Table 3. Evaluation of the balancing operations.

Datasets	IIIT5K	SVT	IC13	IC15	SVTP	CUTE80
Single	92.1	86.6	91.9	72.5	77.1	80.5
Add	94.2	**89.4**	94.0	77.3	81.7	87.1
CL	95.1	88.3	94.1	77.2	83.0	**88.5**
Balance	**95.4**	89.1	**95.1**	**78.4**	**83.1**	88.2

5 Conclusions

In this paper, we propose a novel Semantic GAN and Balanced Attention Network (SGBANet) for arbitrarily oriented scene Text recognition. The Semantic GAN is designed to align the semantic feature distribution between the support and target domain. The Semantic Generator Module focuses on generating simple semantic features for the scene text images. The Semantic Discriminator Module aims to distinguish the semantic features between the support domain and target domain. Experiments show that the Semantic GAN successfully generates simple semantic features for complex scene text images. The generated simple semantic features share the same feature distribution with those of clear images. Furthermore, to alleviate the problem of attention drift, the Balanced Attention Module is designed. It utilizes the convolutional features to correct the attention weights on the semantic features. A new balancing operation is performed on the two glimpse vectors and a balanced glimpse vector is learned. Ablation study on the baseline method combined with the proposed modules demonstrates the advantages of the designed modules. Extensive experiments on six benchmarks demonstrate the effectiveness of our proposed method.

Acknowledgements

This work is supported by the National Key Research and Development Program of China under Grant No. 2020AAA0107903, the National Natural Science Foundation of China under Grant No. 62176091, and the Shanghai Natural Science Foundation of China under Grant No. 19ZR1415900.

References

1. Aberdam, A., Litman, R., Tsiper, S., Anschel, O., Slossberg, R., Mazor, S., Manmatha, R., Perona, P.: Sequence-to-sequence contrastive learning for text recognition. In: Proc. CVPR. pp. 15302–15312 (2021)
2. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein generative adversarial networks. In: Proc. ICML. pp. 214–223 (2017)
3. 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: Proc. ICCV. pp. 4715–4723 (2019)
4. 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: Proc. CVPR. pp. 3113–3122 (2021)
5. Bhunia, A.K., Ghose, S., Kumar, A., Chowdhury, P.N., Sain, A., Song, Y.Z.: Metahtr: Towards writer-adaptive handwritten text recognition. In: Proc. CVPR. pp. 15830–15839 (2021)
6. Biten, A.F., Tito, R., Mafla, A., Gomez, L., Rusinol, M., Valveny, E., Jawahar, C., Karatzas, D.: Scene text visual question answering. In: Proc. ICCV. pp. 4291–4301 (2019)
7. Cai, H., Sun, J., Xiong, Y.: Cstr: A classification perspective on scene text recognition. arXiv e-prints pp. arXiv–2102 (2021)
8. Cheng, Z., Bai, F., Xu, Y., Zheng, G., Pu, S., Zhou, S.: Focusing attention: Towards accurate text recognition in natural images. In: Proc. ICCV. pp. 5076–5084 (2017)
9. Cheng, Z., Xu, Y., Bai, F., Niu, Y., Pu, S., Zhou, S.: Aon: Towards arbitrarily-oriented text recognition. In: Proc. CVPR. pp. 5571–5579 (2018)
10. Fang, S., Xie, H., Chen, J., Tan, J., Zhang, Y.: Learning to draw text in natural images with conditional adversarial networks. In: Proc. IJCAI. pp. 715–722 (2019)
11. Fang, S., Xie, H., Wang, Y., Mao, Z., Zhang, Y.: Read like humans: autonomous, bidirectional and iterative language modeling for scene text recognition. In: Proc. CVPR. pp. 7098–7107 (2021)
12. Gao, Y., Chen, Y., Wang, J., Tang, M., Lu, H.: Reading scene text with fully convolutional sequence modeling. Neurocomputing **339**, 161–170 (2019)
13. 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 **27** (2014)
14. Gupta, A., Vedaldi, A., Zisserman, A.: Synthetic data for text localisation in natural images. In: Proc. CVPR. pp. 2315–2324 (2016)
15. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proc. CVPR. pp. 9729–9738 (2020)
16. He, P., Huang, W., Qiao, Y., Loy, C.C., Tang, X.: Reading scene text in deep convolutional sequences. In: Proc. AAAI. pp. 3501–3508 (2016)
17. Hu, W., Cai, X., Hou, J., Yi, S., Lin, Z.: Gtc: Guided training of ctc towards efficient and accurate scene text recognition. In: Proc. AAAI. pp. 11005–11012 (2020)
18. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Deep structured output learning for unconstrained text recognition. arXiv preprint arXiv:1412.5903 (2014)
19. 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)

20. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Reading text in the wild with convolutional neural networks. International journal of computer vision **116**(1), 1–20 (2016)
21. Kang, L., Rusinol, M., Fornés, A., Riba, P., Villegas, M.: Unsupervised writer adaptation for synthetic-to-real handwritten word recognition. In: Proc. WACV. pp. 3502–3511 (2020)
22. 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: Proc. ICDAR. pp. 1156–1160 (2015)
23. 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: Proc. ICDAR. pp. 1484–1493 (2013)
24. Le, Q.N.N., Bhattacharyya, A., Chembakasseril, M.T., Hartanto, R.: Real-time sign detection and recognition for self-driving mini rovers based on template matching and hierarchical decision structure. In: Proc. ICAART. pp. 208–215 (2020)
25. Li, H., Wang, P., Shen, C., Zhang, G.: Show, attend and read: A simple and strong baseline for irregular text recognition. In: Proc. AAAI. pp. 8610–8617 (2019)
26. Liao, M., Pang, G., Huang, J., Hassner, T., Bai, X.: Mask textspotter v3: Segmentation proposal network for robust scene text spotting. In: Proc. ECCV. pp. 706–722 (2020)
27. Liao, M., Zhang, J., Wan, Z., Xie, F., Liang, J., Lyu, P., Yao, C., Bai, X.: Scene text recognition from two-dimensional perspective. In: Proc. AAAI (2019)
28. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proc. CVPR. pp. 2117–2125 (2017)
29. Long, S., Guan, Y., Bian, K., Yao, C.: A new perspective for flexible feature gathering in scene text recognition via character anchor pooling. In: Proc. ICASSP. pp. 2458–2462 (2020)
30. Luo, C., Jin, L., Sun, Z.: MORAN: A multi-object rectified attention network for scene text recognition. Pattern Recognition. **90**, 109–118 (2019)
31. 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 **129**(4), 960–976 (2021)
32. Luo, C., Zhu, Y., Jin, L., Wang, Y.: Learn to augment: Joint data augmentation and network optimization for text recognition. In: Proc. CVPR. pp. 13746–13755 (2020)
33. Lyu, P., Liao, M., Yao, C., Wu, W., Bai, X.: Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. In: Proc. ECCV. pp. 67–83 (2018)
34. Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z., Paul Smolley, S.: Least squares generative adversarial networks. In: Proc. ICCV. pp. 2794–2802 (2017)
35. Mishra, A., Alahari, K., Jawahar, C.: Top-down and bottom-up cues for scene text recognition. In: Proc. CVPR. pp. 2687–2694 (2012)
36. Odena, A., Olah, C., Shlens, J.: Conditional image synthesis with auxiliary classifier gans. In: Proc. ICML. pp. 2642–2651. PMLR (2017)
37. Phan, T.Q., Shivakumara, P., Tian, S., Tan, C.L.: Recognizing text with perspective distortion in natural scenes. In: Proc. ICCV. pp. 569–576 (2013)
38. Qiao, Z., Zhou, Y., Yang, D., Zhou, Y., Wang, W.: Seed: Semantics enhanced encoder-decoder framework for scene text recognition. In: Proc. CVPR. pp. 13528–13537 (2020)

39. 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)
40. 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 **39**, 2298–2304 (2016)
41. 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 **41**(9), 2035–2048 (2018)
42. Wan, Z., He, M., Chen, H., Bai, X., Yao, C.: Textscanner: Reading characters in order for robust scene text recognition. In: Proc. AAAI (2020)
43. Wang, K., Babenko, B., Belongie, S.: End-to-end scene text recognition. In: Proc. ICCV. pp. 1457–1464 (2011)
44. Wang, Q., Huang, Y., Jia, W., He, X., Blumenstein, M., Lyu, S., Lu, Y.: Faclstm: Convlstm with focused attention for scene text recognition. Science China Information Sciences **63**(2), 1–14 (2020)
45. Wang, T., Zhu, Y., Jin, L., Luo, C., Chen, X., Wu, Y., Wang, Q., Cai, M.: Decoupled attention network for text recognition. In: Proc. AAAI. pp. 12216–12224 (2020)
46. Wang, W., Xie, E., Liu, X., Wang, W., Liang, D., Shen, C., Bai, X.: Scene text image super-resolution in the wild. In: Proc. ECCV. pp. 650–666 (2020)
47. Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: Cbam: Convolutional block attention module. In: Proc. ECCV. pp. 3–19 (2018)
48. Xie, Y., Chen, X., Sun, L., Lu, Y.: Dg-font: Deformable generative networks for unsupervised font generation. In: Proc. CVPR. pp. 5130–5140 (2021)
49. Xie, Z., Huang, Y., Zhu, Y., Jin, L., Liu, Y., Xie, L.: Aggregation cross-entropy for sequence recognition. In: Proc. CVPR. pp. 6538–6547 (2019)
50. Xing, L., Tian, Z., Huang, W., Scott, M.R.: Convolutional character networks. In: Proc. ICCV. pp. 9126–9136 (2019)
51. Yan, R., Peng, L., Xiao, S., Yao, G.: Primitive representation learning for scene text recognition. In: Proc. CVPR. pp. 284–293 (2021)
52. 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: Proc. ICCV. pp. 9147–9156 (2019)
53. Yang, S., Wang, Z., Wang, Z., Xu, N., Liu, J., Guo, Z.: Controllable artistic text style transfer via shape-matching gan. In: Proc. ICCV. pp. 4442–4451 (2019)
54. Yue, X., Kuang, Z., Lin, C., Sun, H., Zhang, W.: Robustscanner: Dynamically enhancing positional clues for robust text recognition. In: Proc. ECCV. pp. 135–151 (2020)
55. Zhan, F., Lu, S.: Esir: End-to-end scene text recognition via iterative image rectification. In: Proc. CVPR. pp. 2059–2068 (2019)
56. Zhang, C., Gupta, A., Zisserman, A.: Adaptive text recognition through visual matching. In: Proc. ECCV. pp. 51–67 (2020)
57. 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: Proc. CVPR. pp. 2740–2749 (2019)
58. Zhou, W., Ge, T., Xu, K., Wei, F., Zhou, M.: Self-adversarial learning with comparative discrimination for text generation. arXiv preprint arXiv:2001.11691 (2020)
59. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proc. ICCV. pp. 2223–2232 (2017)