Representative Batch Normalization for Scene Text Recognition

Yajie Sun[1,2*], Xiaoling Cao[1] and Yingying Sun[1]
[1]School of Computer Science, Nanjing University of Information Science & Technology, Nanjing, China
[2]Engineering research center of digital forensics ministry of education, Nanjing University of Information Science & Technology, Nanjing, China
[Email: yanacn@163.com]
*Corresponding Author: Yajie Sun

Received January 19, 2022; revised May 31, 2022; accepted July 3, 2022; published July 31, 2022

Abstract

Scene text recognition has important application value and attracted the interest of plenty of researchers. At present, many methods have achieved good results, but most of the existing approaches attempt to improve the performance of scene text recognition from the image level. They have a good effect on reading regular scene texts. However, there are still many obstacles to recognizing text on low-quality images such as curved, occlusion, and blur. This exacerbates the difficulty of feature extraction because the image quality is uneven. In addition, the results of model testing are highly dependent on training data, so there is still room for improvement in scene text recognition methods. In this work, we present a natural scene text recognizer to improve the recognition performance from the feature level, which contains feature representation and feature enhancement. In terms of feature representation, we propose an efficient feature extractor combined with Representative Batch Normalization and ResNet. It reduces the dependence of the model on training data and improves the feature representation ability of different instances. In terms of feature enhancement, we use a feature enhancement network to expand the receptive field of feature maps, so that feature maps contain rich feature information. Enhanced feature representation capability helps to improve the recognition performance of the model. We conducted experiments on 7 benchmarks, which shows that this method is highly competitive in recognizing both regular and irregular texts. The method achieved top1 recognition accuracy on four benchmarks of IC03, IC13, IC15, and SVTP .

Keywords: Scene text recognition, deep learning, Representative Batch Normalization, Feature representation, Feature enhancement.

The code and datasets are available at github

http://doi.org/10.3837/tiis.2022.07.015 **ISSN : 1976-7277**

1. Introduction

Text is an important tool for computers to recognize and understand the world, and many researchers are engaged in the research of text-related topics, such as text emotion classification[1], text document security[2], and scene text recognition[3], etc. Scene Text Recognition (STR) refers to recognizing text in different natural environments, such as billboards, road signs, trademarks, etc. STR is widely used in artificial intelligence applications, which contains autonomous driving, image retrieval, and intelligent translation. STR can recognize characters in real natural environments, conversely, Optical Character Recognition (OCR) is used to recognize characters in documents with a neat background. STR is more complicated than OCR because of the diversity of backgrounds in the real environment, camera angles, and lighting conditions, which will affect the quality of natural images. An example of some recognition difficulties (e.g., curved, occlusion and blurred, etc.) is shown in **Fig. 1**.

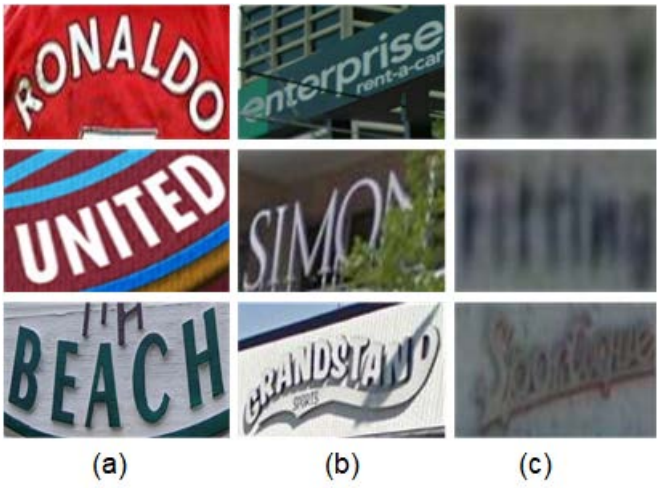

Fig. 1. Three kinds of low-quality text images (a) curved. (b) occlusion. (c) blurred font.

STR is divided into text detection and text recognition. In this paper, the main research is text recognition. During previous work, the traditional method[3] is used to recognize text on a character by character in a bottom-up manner. To some extent, the above method cannot take advantage of the sequence relationship between characters, which will weaken the recognition effect of the text. Currently, the problem of text recognition is transformed into sequence prediction in [3-6]. In [7] the original image was rectified to a horizontal text image by the image transformation module, which reduces the difficulty of processing irregular text. Before recognition, a pluggable super-resolution module was introduced to process low-resolution images without introducing additional computing time [8].

Although the above methods have made great contributions in the field of scene text recognition, all of them have an inherent challenge. On the one hand, using ResNet as a feature extractor will lead to the testing effect being worse than expected because the feature representation of the model had a strong dependence on the training data. On the other hand, using ResNet to extract features on low-quality images will easily lead to incorrect recognition results due to the lack of detailed features. In general, the visual information of scene text images is not fully learned and exploited.

Our motivation is to explore the valuable feature presentation from two aspects to improve the performance of the feature, which can reduce the accuracy difference between models on different test and training sets, as well as be able to improve the accuracy of models on multiple types of data sets. For the feature extract aspect, inspired by [9] that introduced [10] to extract robust features. We introduce a Representation Batch Normalization (RBN) to address the variability of feature representation among different instances and embed RBN into ResNet as the basic framework of the feature extractor. For the feature enhancement aspect, the U-shape network [9-10] can improve the recognition by merging multi-scale features. So, we propose a feature enhancement Network (FEN) to refine the feature representation, which contains low-level and high-level semantic information. In this paper, we propose a robust scene text recognizer based on representation batch normalization (RBN-STR). In addition to enhancing the representation of feature maps, this method also contacts contextual features and visual features [11-13] for recognition. In summary, the main contributions of this paper include three aspects as follows:

- First, we combine ResNet and RBN to propose a robust and efficient feature extractor. This can alleviate the dependence of the model on training data and help the model extract effective features on test sets.
- Second, we propose a feature enhancement module, which incorporates multi-scale visual feature maps with different resolutions. As a result, more refined feature information is contained in feature maps and feature representation is enhanced.
- Third, by contacting visual features and contextual information, our experimental results achieve the accuracy of top-1 on four benchmarks.

2. Related work

Scene text recognition is an important research topic in the field of computer vision. [14] is a comprehensive discussion of scene text detection and recognition, which divides STR into regular scene text recognition and irregular scene text recognition. This section will review the popular research approaches from two different categories.

2.1 Regular scene text recognition

In recent years, with the rapid development of deep learning, the research of scene text recognition has made significant progress. Some methods regard scene text recognition as sequence prediction of text. Such as [15-18] integrated the advantages of both Convolutional Neural Network(CNN) as a feature extractor to obtain the spatial features of images and Recurrent Neural Network(RNN) for obtaining contextual features. CRNN was built by Shi et al [4] which was the first model to combine CNN and RNN for scene text recognition. And they employed CTC for transcription. The advantage of CRNN is that it can recognize variable-length character sequences and learn directly from sequence labels without additional character annotations. Liang et al [6] analyzed and concluded that the feature extractor of CRNN could not extract the advanced features of the image. Therefore, they employed ResNet to extract effective features and introduced a rectangular convolution stride to expand the receptive field of feature extraction at the same time. Gao et al. [5] introduced DSAN as a mutually reinforcing supervision mechanism, which is combined with context-level modeling and supervision enhancement.

The above methods use decoders based on CTC, which can solve the problem of unaligned characters during training. [19] introduced an RNN to capture longer context dependencies, and used a decoder based on the attention mechanism to capture the target

sequence. Based on the Encoder-Decoder framework, a new decoder combining CTC and attention mechanism was proposed in [18]. Cheng et al. [17] found that using an attention mechanism would generate attention drift when images with complex backgrounds or low quality. They proposed FAN to automatically align the offset feature region. To solve the attention alignment problem, DAN was pointed out by Wang et al [20] that decoupled the alignment operation from the historical decoding information. Thus, the problem of misalignment caused by decoding error was solved.

2.2 Irregular scene text recognition

The state-of-the-art methods have achieved high recognition accuracy on regular scene text images, but they encounter difficulties in dealing with irregular scene text (e.g., curved and perspective text). Baek et al. [21] introduced a unified four-stage STR framework that most existing STR models fit into. Shi et al [22] proposed RARE, an irregular scene recognition method that includes a Spatial Transformer Network (STN) and a recognition network. In RARE, STN with predicted TPS transform can rectify original text into normal text for the following recognition network. Zhan et al [23] have achieved considerable success in correcting images because they improved the rectification pipeline that applies a line fitting transform to iteratively correct images. Luo et al [24] tackled rotated, scaled, and stretched characters with MORN, which is free of geometric constraints. AON was described by Cheng et al [25], which only required image and text-level annotations for training. Li et al. [26] designed an encoder-decoder framework based on 2D attention which has the advantage of not requiring text correction. STAN was proposed in [27], which divides the image into multiple non-overlapping regions and then performs the transformation for each region separately. And the sequential transformation is designed to achieve the smooth connection of adjacent regions.

In summary, the methods mentioned above either ignored the dependency of feature extraction on the training data or the importance of detailed features for recognition. In order to improve the generalization ability of the recognizer and increase the capacity of the feature map, we propose a robust scene text recognizer by exploiting RBN and a feature enhancement network.

3. Method

For STR, we propose RBN-STR, which consists of five components, as shown in **Fig. 2**; 1) Image Transformation: input image is corrected to a regular image. 2) Feature Extractor: rectified image is used to extract multi-scale visual feature maps. 3) Feature Enhancement: multi-scale feature maps are enhanced and fused into feature maps of the same size (H/4×W/4×512). 4) Sequence Modeling: it was used to extract the contextual information of the text, and contact the contextual information and the feature map to form a new feature space. 5) Decoder: the feature space is decoded into predictive texts. In this section, we describe the framework of RBN-STR in detail.

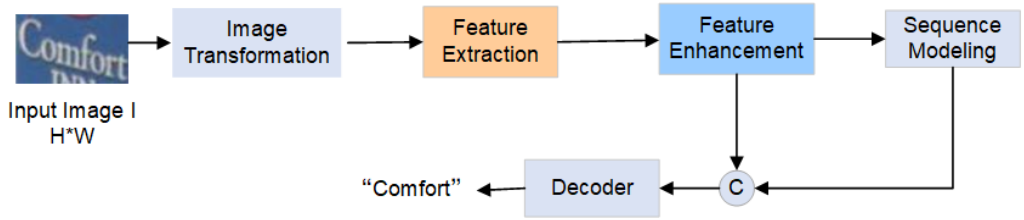

Fig. 2. The proposed architecture of RBN-STR.

3.1 Image Transformation

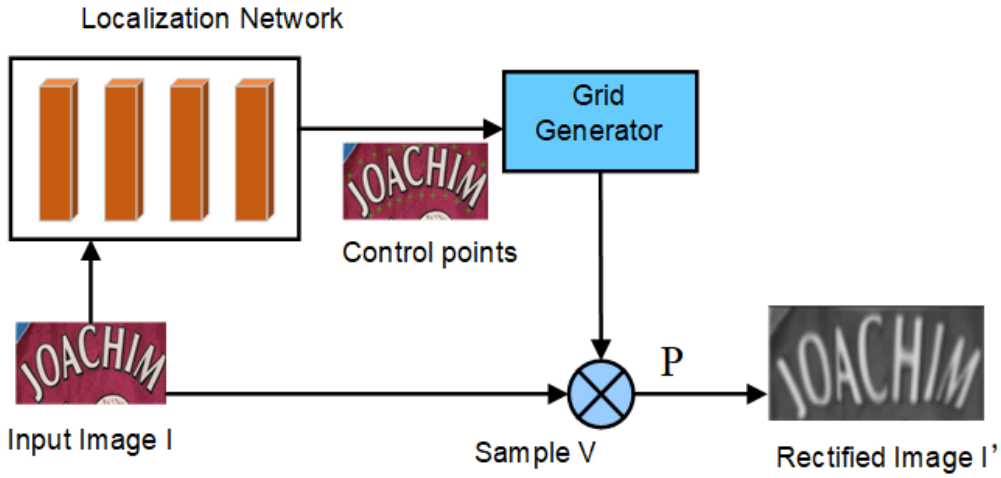

Fig. 3. The architecture of STN.

The structure of the Image transform uses the spatial transformation network (STN) to rectify the image, as same as Aster [7]. STN includes three parts: localization network, grid generator, and sampler, as shown in **Fig. 3**. The size of the STN input image I is 100 × 32. The rectification process is as follows. Firstly, a set of control points is predicted on the image I by a localization network. Then, the TPS transformation parameters are calculated in the grid generator using the control points, and the sampling grid P is generated at the image I. Finally, the sampling grid P and the image I are simultaneously fed into the sampler V, and the corrected image I' is obtained by sampling on the grid. The size of I' as same as the input.

3.2 Feature Extraction

3.2.1 BatchNorm

The feature mapping is $X \in R^{N\times C\times H\times W}$, where N, C, H, and W represent the batch size, the number of channels, and the height and width of the input features, respectively. The operation of BatchNorm for X is performed as follows that X denotes feature maps. First, X_c is centralized feature map that is obtained by X centralization, represented as follows

$$X_c = X - E(X) \tag{1}$$

Where *E(X)* denotes the mean value used for centering, which is represented as follows.

$$E(X) \leftarrow mE(X) + (1-m)\mu_B \tag{2}$$

Where m denotes accumulation momentum, μ_B represents the mean value of the mini-batch in the training phase, expressed as follows.

$$\mu_B = \frac{1}{NHW}\sum_{n=1}^{N}\sum_{h=1}^{H}\sum_{w=1}^{W} X \tag{3}$$

Then, executing the scaling operation on the feature map X_c, we get the scaled feature map X_s, represented as follows

$$X_s=\frac{X_c}{\sqrt{Var(X)+\epsilon}} \tag{4}$$

Where ϵ is used to avoid zero variance, *Var(X)* denotes the variance in the scaling operation, which is represented as follows.

$$Var(X)\leftarrow mVar(X)+(1-m)\sigma_B^2 \tag{5}$$

Where σ_B^2 is the variance value of the mini-batch in the training phase, represented as follows

$$\sigma_B^2=\frac{1}{NHW}\sum_{n=1}^{N}\sum_{h=1}^{H}\sum_{w=1}^{W}(X-\mu_B)^2 \tag{6}$$

Finally, the feature map X_s is affine transformed to obtain the radiative transformed feature Y with the following expression.

$$Y=X_s\gamma+\beta \tag{7}$$

Where γ and β are the scaling and translation parameters, respectively.

However, the test data usually has a large variability with the training data, which reduces the generalization ability of the network when the data distribution is different. On the one hand, inappropriate running mean values in the testing phase can make the centering features contain extra noise after activation or lose feature representations. On the other hand, inappropriate running variance can produce some scaling features with too much/too little intensity and result in an unstable feature distribution between channels, which leads to a weaker feature representation in the test phase. In this regard, we propose an RBN to enhance the representative feature representation of different instances and produce a more stable feature distribution.

3.2.2 Representation Batch Normalization

Centering and scaling are easy to ignore the feature differences between data individuals. So, we introduce the calibration mechanism into BatchNorm, which is enriching the feature representations on the test sets. According to [28], the BatchNorm is replaced by RBN after each convolutional layer as a novel residual block, the structure is shown in **Fig. 4**.

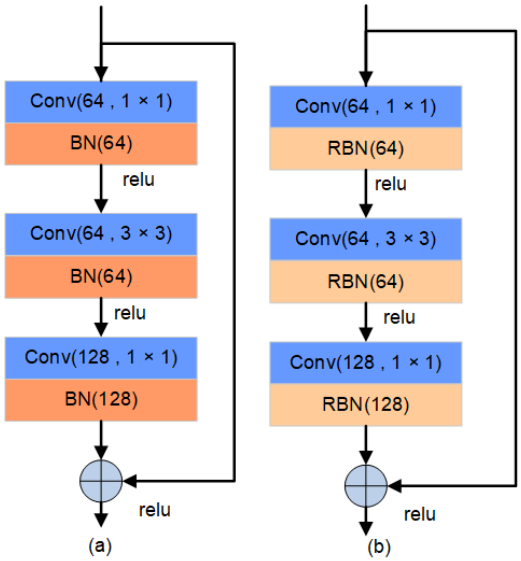

Fig. 4. Residual blocks. (a) denotes the structure with BatchNorm, (b) denotes the structure with RBN.

RBN proposes centering calibration and scaling calibration based on the original BatchNorm. Centering calibration enhances valid feature information and reduces noisy features by using instance-specific statistics to move features around. The scaling calibration accordingly adjusts the intensity of the features based on the statistics of the instances to produce a more stable distribution of features. The calibration mechanism improves the feature representation ability of individual data while maintaining the mini-batch advantage of BatchNorm. Unlike BatchNorm, we calculate the mean value (μ_c) and variance (σ_c^2) of the feature map from the channel dimension, denoted as follows.

$$\mu_c = \frac{1}{HW} \sum_{h=1}^{H} \sum_{w=1}^{W} X \tag{8}$$

$$\sigma_c^2 = \frac{1}{HW} \sum_{h=1}^{H} \sum_{w=1}^{W} (X - \mu_c)^2 \tag{9}$$

To make the centering operation not dependent on the running mean, we perform a centering calibration operation on feature *X* before the feature centering operation, the formula is defined as follows:

$$X_{cm} = X + \omega_m . K_m \tag{10}$$

where ω_m denotes the weight and K_m denotes the statistics of the *X*.

The scaling operation affects the intensity of features when the affine transformation is unchanged, and scaling features using inaccurate running variance can lead to instability in feature intensity. We calibrate the feature intensity by using a scaling calibration after the original scaling operation, the expression is defined as follows:

$$X_{cs} = X_s \cdot R(\omega_v \cdot K_s + \omega_b) \tag{11}$$

where ω_v and ω_b denote trainable parameters and K_s denotes the statistics of the feature map X_s; R () is a calibration function used to compress the eigenvalues outside the distribution to make the feature distribution more stable. It is represented by Sigmoid function. Compared with BatchNorm, RBN reduces noise and improves feature representation. We can directly introduce the RBN in ResNet to propose a new encoder. The regular text image is used to encode. Generally, the low resolution of the image obtained by Image transformation will affect the recognition performance. Our proposed encoder improves the feature extraction of low-quality images. The four stages of the encoder output four feature maps of different sizes, respectively. These feature maps are 1/4, 1/8, 1/16, and 1/32 of the input image, respectively. The used encoder to extract feature maps with different resolutions, which contain more detailed and accurate semantic information. Finally, we use a 1×1 convolutional layer to change the number of channels of each feature map to 128, and obtain a feature pyramid F_r.

3.3 Feature Enhancement Network

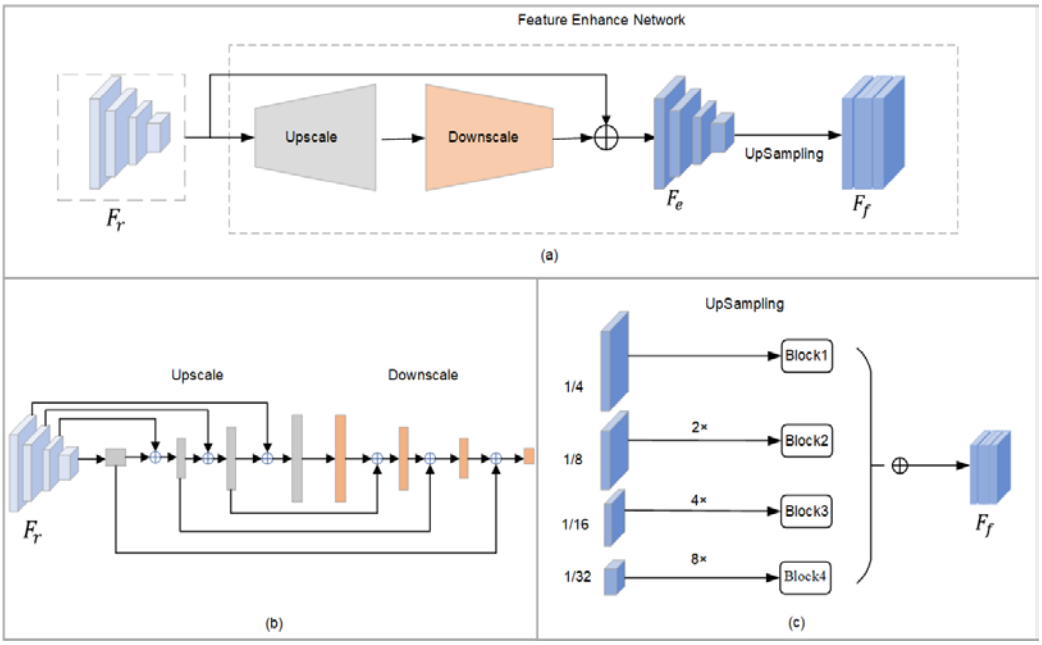

Fig. 5. (a) denotes the structure of the feature enhancement Network. (b) denotes the process of upscale convolutional layers and downscale convolutional layers. (c) denotes the up-sampling process of different scale feature maps.

Due to the limited acceptance domain of feature maps, we propose a low computational feature enhancement (FEN) network to further improve the ability of feature representation. FEN is a U-shaped network, which integrates low-level and high-level information to enhance the characteristics of different scales. The collected low-level and high-level semantic information enriches the details of the feature map and is helpful for text prediction. After obtaining the feature pyramid F_r the FEN is applied to it. As shown in **Fig. 5**, FEN includes three stages upscale convolutional layers, downscale convolutional layers, and up-sampling. The feature pyramid F_r is entered into upscale convolutional layers, sequentially convolving in the stride of 32, 16, 8, and 4 to output an upscale enhanced feature pyramid. In the downscale convolution stage, the output of each layer is added to the output of the same size in the upscaling convolutional layers as the input of the next layer. The downscale convolution stage is successively performed in the stride of 4, 8, 16, and 32, and the output feature pyramid is connected with F_r to obtain the final feature pyramid F_e. Finally, to unify the scale of the feature maps, all feature maps of F_e are upsampled to 1/4 of the original image and integrated into the enhanced feature map F_f, the size is H/4×W/4×512.

Feature maps with different resolutions from low to high are combined by FEN, which is covering a larger receptive field and containing more diversity of semantic information. FEN not only deepens the network structure but also effectively refines the features. Thereby, the ability of feature representation is enhanced. In addition, the contact function in feature enhancement is referred to as separable convolution, which improves computational efficiency.

3.4 Sequence Modeling

LSTM[4] is an RNN unit that can be used to learn contextual cues but solves the problem of vanishing gradients during training. LSTMs help to classify, analyze and evaluate time series

related data. In sentiment classification research[1], stacking multiple LSTM layers is used for sequence classification. However, this paper uses BiLSTM[6] to capture long-range dependencies on feature maps that consist of a forward and a backward LSTM.

We use BiLSTM to capture contextual information in character sequences to derive semantic information. The scene text image has high semantic information, which can assist features to predict text sequences. However, in feature extraction stage only pays attention to the visual features of the image and ignores the semantic information of the text. Therefore, the model may be hindered in processing low-quality image recognition problems such as blur, occlusion, and incomplete characters. To mitigate the impact of the lack of semantic information on the recognition effect, we combine the visual and semantic information of images to deal with the recognition problem of scene text.

The feature map F_f is fed into BiLSTM to get a sequence of contextual features $V = Seq(F_f)$with the same length. Then, we contact the visual feature map and the contextual sequence to obtain a new feature space $N = (F_f, V)$.

3.5 Decoder

In the prediction stage, the feature sequence is predicted as the target string sequence. The common methods are connectionist time classification (CTC) and attention mechanism. However, the effect of using the attention mechanism to achieve prediction is better than using CTC. Attention-based LSTM refers to the combination of attention mechanism and LSTM structure as a prediction module. Attention-based LSTM can automatically capture the information flow in the input sequence and predict the output sequence. In the decoding stage, an attention-based LSTM is used to decode, which aligns the attention region with the corresponding truth labels. At moment *t*, the output predicted by the decoder is y_t:

$$y_t = \text{softmax}(W_y s_t + b_y) \tag{12}$$

Where *softmax* is activation function, W_y and b_y are trainable parameters and s_t is the hidden state of the LSTM at moment *t*. The expressions are as follows.

$$s_t = \text{LSTM}(y_{t-1}, g_t, s_{t-1}) \tag{13}$$

Where g_t is the glimpse vector, which is the weighted sum of the feature space vector $C = (c_1, c_2, \cdots, c_T)$,

$$g_t = \sum_{j=1}^{T} \alpha_{t,j}\, n_j \tag{14}$$

Where $\alpha_{t,j}$ is the attention weight, calculated as follows:

$$\alpha_{t,j} = \frac{\exp(e_{t,j})}{\sum_j^T \exp(e_{t,j})} \tag{15}$$

$$e_{t,j} = v^T \tanh(W s_{t-1} + V n_j + b) \tag{16}$$

where v, W, V and b are trainable parameters and the dimension of the LSTM hidden layer is 512.

4. Experiment

In this section, the feasibility of RBN-STR is verified by extensive experiments. First, the training and test datasets are briefly introduced in 4.1 and the detailed implementation process is described in 4.2. Then, RBN-STR is analyzed and compared with the advanced methods in 4.3. Finally, in order to better present the innovative points of this paper, we perform ablation experiments and analyze the impact of our main contributions in 4.4.

4.1 Datasets

Firstly, the model is trained on the two synthetic datasets Synth Text and MJSynth, which contain 9 million and 8 million synthetic word images respectively. Then, the model is tested on four regular datasets (IIIT5K, SVT, ICDAR2003, ICDAR2013) and three irregular datasets (ICDAR2015, SVTP, and CUTE).

MJSynth (MJ) [29] is a synthetic data set published by Jaderberg in 2014. There are 9 million images covering 90K words. SynthText (ST) [30] is a composite training set containing 8 million images. Each image has about ten-word instances, annotated with character and word-level bound boxes.

IIIT 5K-Words (IIIT5K) [31] contains 3000 regular images for testing, which are cropped from Internet images. The text sample of images is almost horizontal. Street View Text (SVT) [32] contains 647 images with word-level axis-aligned bounding boxes, in the test dataset, which are collected from Google Street View. These images suffer from noise, blur, or have low resolutions. ICDAR2003 (IC03) [33] contains 251 scene images and 867 cropped images for testing, which are annotated with text bounding boxes.

ICDAR2013 (IC13) [34] contains 1015 cropped word images from signboards, books, and posters. Most of them are inherited by IC03. ICDAR2015 (IC15) [35] contains 4468 images for training and 2077 images for testing. Most images contain irregular text (oriented, perspective, or curved) because they are cropped under arbitrary angles. Street View Text Perspective (SVTP) [36] is collected from side-view angle snapshots in Google Street View and includes 639 images for testing. Therefore, most of them are subjected to critically perspective distortions. CUTE80 (CT80) [37] is a curved text data set, containing 288 images that are cropped from natural scenes. The dataset of the images with high resolution is annotated by words, and the dataset is used to test.

4.2 Implementation Details

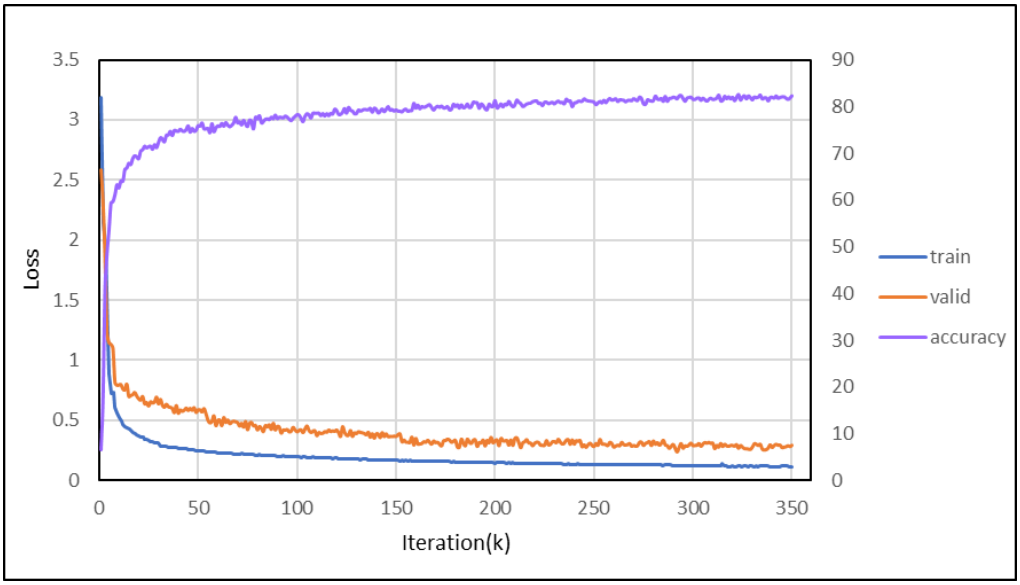

Fig. 6. Training loss on train datasets and testing accuracy on the validation dataset.

The model we proposed is based on Python3.6. The hardware environment for experimental training and testing is a Tesla V100 GPU with 32G memory. We use MJSynth and SynthText as training dataset s, and there are about 17 million synthetic images in total. We use only

synthetic data for training and do not need to fine-tune any of the datasets. In the model, the size of the image is 32 × 100 and we optimize the training model using AdaDelta optimizer. We set the initial learning rate λ to 1 and the batch size to 128. The model can recognize 36 types of characters, including 26 case-insensitive letters and 10 digits. The accuracy and loss curves are displayed in **Fig. 6.**

4.3 Results on regular and irregular datasets

To verify the validity of our proposed method, we evaluate and compare our method with the previous state-of-the-arts on the above-mentioned benchmarks. Following the previous methods, we measure the recognition performance with word recognition accuracy (*WRA*). *WRA* is defined by $WRA = W_c/W_t$, where W_t represents the total number of words, and W_c represents the number of correctly recognized words.

We compare the *WAR* of our method and previous outstanding methods on four regular datasets, as shown in **Table 1**. Our method achieves state-of-the-art performance on IC03 and IC13. There are partially curved or oriented images in IIIT5k and SVT, thus causing our accuracy to be slightly lower than [5,17,24]. The performance on the regular scene text dataset shows that the method has great competitiveness. Likewise, we evaluate the performance of our method on three irregular text datasets. The result is displayed in Table 1. RBN-STR achieves excellent performance on IC15 and SVTP, which is more accurate than other methods. However, the performance on CUTE is not satisfactory. Images in CUTE. Some images in CUTE contain multiple deformations and complex text distortion, while STN correct the shape of the text by rotating, scaling, and translation them. The image include additional noise after image transform and feature enhancement reinforce the characteristics of some of the noise, which is an important reason why the model performs poorly on CUTE.

Table 1. Recognition accuracies (%) with different methods on 7 benchmarks. The best accuracy is shown in red font.

Method	Regular				Irregular		
	IIIT5K	SVT	IC03	IC13	IC15	SVTP	CUTE
MORAN[24]	91.2	88.3	95.0	92.4	68.8	76.1	77.4
DAN[20]	94.3	89.2	95.0	93.9	74.5	80.0	84.4
STAN[27]	94.1	90.6	95.1	92.8	76.7	82.2	83.3
RNRT[6]	84.7	80.0	90.6	90.1	-	70.9	62.6
SAR[26]	95.0	91.2	-	94.0	78.8	79.2	81.3
ASTER[7]	93.4	93.6	94.5	91.8	76.1	78.5	79.5
DMDAN[31]	92.3	86.9	93.3	92.6	75.4	78.2	83.3
FAN[17]	87.4	85.9	94.2	93.3	81.3	-	-
ESIR[13]	93.3	90.2	90.2	91.3	76.9	79.9	83.3
Ours	93.0	90.5	95.5	94.1	79.9	82.3	72.2

4.4 Inference Speed

Table 2. Comparison of inference speed

Methods	ESIR	MORAN	ASTER	RBN-STR
Time(ms)	28.0	3.50	24.16	2.71

To explore the efficiency of RBN-STR, we evaluate the average inference time per image and compare it with state-of-the-art rectification methods. For comparison, we test all methods with the same running environment. **Table 2** present the inference speed of each method. On the same hardware, the RBN-STR has the fastest inference speed with test time of only 2.71ms. They used two layers of BiLSTM and ESIR with 5 rectification iterations. Rectification and BiLSTM were computationally intensive and time consuming. But we proposed feature extraction network is computationally light, and the proposed FEN does not introduce additional computation.

4.5 Comparison with ASTER

In our experiments, we use ASTER [7] as the baseline, which does not contain the RBN and feature enhancement network. As shown in **Fig. 7**, the accuracy of RBN-STR on the two regular text data sets and the two irregular text datasets are higher than ASTER. In detail, our model improves by 1% on IC03, 2.3% on IC13, up to 3.8% on IC15, and 3.8% on SVTP compared to ASTER. But the accuracy of RBN-STR on IIITK, SVT, and CUTE is slightly lower than ASTER. We randomly select some curved, blurred, and occluded low-quality images from the test set of irregular scene text, which is used to verify the recognition performance of the two different methods. **Fig. 8** shows the visual recognition results of some images.

These examples in **Fig. 8** point out that our proposed model is excellent for text recognition in partially irregular scenes. Both ASTER and RBN-STR correctly recognize curved and perspective text because they both use the same STN module, which transforms the input image into a horizontal text image. For recognizing images with complex backgrounds and images with blurred and obscured characters, such as "World", "Kitchen", "PARK", and "SIMON", ASTER is difficult to recognize, but RBN-STR can recognize them effectively. This is because RBN-STR uses richer features covering a wide range of receptive fields, and uses semantic information to assist text prediction in the decoding stage. Despite the disadvantage in a few cases, the recognition performance of RBN-STR has also achieved convincing progress.

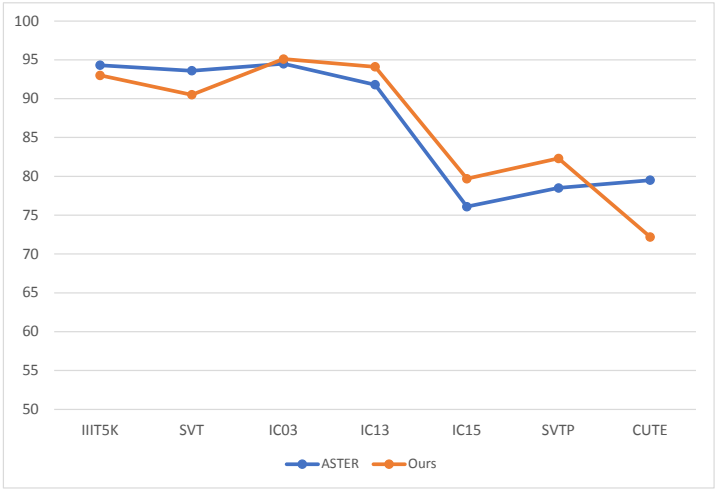

Fig. 7. Comparative evaluation of ASTER and our proposed method.

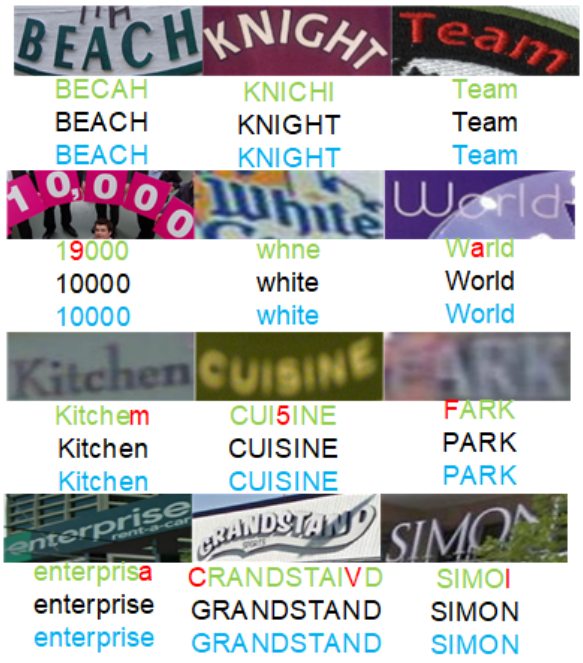

Fig. 8. Visual recognition results in two models. ASTER, RBN-STR, and Ground Truth are represented by green, black, and blue fonts, respectively. The error results are noted by red fonts.

4.6 Ablation Study

To verify the effectiveness of RBN and FEN, we did a series of ablation experiments. All experiments were trained using MJ and ST, and the accuracy of the models was compared on four datasets, IC03, IC13, IC15, and SVTP.

Table 3. Comparison of two regular benchmarks and two irregular benchmarks with different strategies. RBN means to use Representation BatchNorm instead of general BatchNorm. FEN represents embedding the feature enhancement network into the model.

Method	RBN	FEN	IC03	IC13	IC15	SVTP
Baseline	×	×	94.5	91.8	76.1	78.5
with RBN	√	×	94.8	92.5	75.8	79.5
with FEN	×	√	95.2	92.0	76.6	81.3
RBN-STR	√	√	95.5	94.1	79.9	82.3

To verify the function of RBN, we use ResNet50 as the backbone network and discuss the effect of different BatchNorm on recognition. From **Table 3** we observe that compared to Baseline, the accuracy of the model using RBN improved on all four datasets, by 0.3% on IC03, 0.7% on IC13, and 0.7% on SVTP, however, reduce by 0.6% on IC15. The results show that using RBN can alleviate the instability of feature extraction caused by instance differences.

To verify the effectiveness of the FEN, we conducted a series of ablation experiments with or without FEN, as shown in **Table 3**. Compared with Baseline, the model with the addition of FEN improves 0.7%, 0.2%, 0.5%, and 2.8% on the four datasets, respectively. The results show that the FFN improves feature representation by fusing semantic information of different resolutions, which affects the recognition performance.

4.7 Discussion

RBN-STR has stable feature representation in the face of irregular scene texts, showing its stability and robustness. Experiments demonstrated that RBN-STR can recognize scene text flexibly in most cases. But there are still difficulties in recognizing certain images, such as complex curvature, perspective, and low resolution. The quality of the image affects the process of image correction and feature extraction, so it is an important cause of recognition failure. Therefore, we may consider reducing the difficulty of STR from the aspect of image preprocessing, thereby improving the recognition performance of the algorithm.

5. Conclusion

In this paper, we consider how to improve the recognition accuracy of low-quality scene text from the feature level. We propose a scene text recognizer based on representation batch normalization, referred to as RBN-STR. Firstly, representation batch normalization is used to enhance the feature representation of the instance, so that the model obtains a stable feature distribution in different test sets. Then, we also analyze the key role of different levels of feature resolution on text recognition, using the feature enhancement module combined with feature maps of different resolutions to refine the feature representation capability. Finally, our proposed RBN-STR has strong competitiveness in scene text recognition, especially on IC03, IC13, IC15, and SVTP. It is experimentally demonstrated that adding the representation batch normalization and feature enhancement modules effectively enhances the text recognition effect.

We consider two promising directions for future work. First, our model does not handle vertical text and text with large distortion well. Therefore, we will continue to investigate the vertical text recognizer in the future. Second, we prepare to integrate text detection into the model and propose an end-to-end scene text recognition method.

References

[1] Jamal, Nasir, Chen Xianqiao, Fadi Al-Turjman, and Farhan Ullah, "A Deep Learning–based App roach for Emotions Classification in Big Corpus of Imbalanced Tweets," *Transactions on Asian and Low-Resource Language Information Processing*, vol.20, no.3, pp.1-16, 2021. Article (CrossRef Link)

[2] Khadam, Umair, Muhammad Munwar Iqbal, Leonardo Mostarda, and Farhan Ullah, "An Efficie nt Framework for Text Document Security and Privacy," in *Proc. of International Symposium o n Security and Privacy in Social Networks and Big Data*, vol.1298, pp. 132-140, 2020. Article (CrossRef Link)

[3] K. Wang and S. Belongie, "Word Spotting in the Wild," in *Proc. of Computer Vision - ECCV*, pp. 591–604, 2013. Article (CrossRef Link)

[4] Shi, X. Bai and C. Yao, "An End-to-End Trainable Neural Network for Image-Based Sequence R ecognition and Its Application to Scene Text Recognition," *IEEE Transactions on Pattern Analys is and Machine Intelligence*, vol. 39, no. 11, pp. 2298-2304, 1 Nov., 2017. Article (CrossRef Link)

[5] Y. Gao, Z. Huang, Y. Dai, C. Xu, K. Chen, and J. Guo, "DSAN: Double Supervised Network with Attention Mechanism for Scene Text Recognition," in *Proc. of 2019 IEEE Int. Conf. Vis. Commun. Image Process. VCIP 2019*, pp. 1–4, 2019. Article (CrossRef Link)

[6] Q. Liang, S. Xiang, Y. Wang, W. Sun, and D. Zhang, "RNTR-Net: A Robust Natural Text Recognition Network," *IEEE Access*, vol. 8, no. 1, pp. 7719–7730, 2020. Article (CrossRef Link)

[7] B. Shi, M. Yang, X. Wang, P. Lyu, C. Yao and X. Bai, "ASTER: An Attentional Scene Text Rec ognizer with Flexible Rectification," *IEEE Transactions on Pattern Analysis and Machine Intelli gence*, vol. 41, no. 9, pp. 2035-2048, 2019. Article (CrossRef Link)

[8] Y. Mou et al., "PlugNet: Degradation Aware Scene Text Recognition Supervised by a Pluggable Super-Resolution Unit," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12360 LNCS, pp. 158–174, 2020. Article (CrossRef Link)

[9] X. Li, J. Liu, G. Zhang, and S. Zhang, "IBN-STR: A robust text recognizer for irregular text in natural scenes," in *Proc. of Int. Conf. Pattern Recognit.*, pp. 9522–9528, 2020. Article (CrossRef Link)

[10] X. Pan, P. Luo, J. Shi, and X. Tang, "Two at Once: Enhancing Learning and Generalization Capacities via IBN-Net," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11208 LNCS, pp. 484–500, 2018. Article (CrossRef Link)

[11] Y. Sun, C. Zhang, Z. Huang, J. Liu, J. Han, and E. Ding, "TextNet: Irregular Text Reading from Images with an End-to-End Trainable Network," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11363 LNCS, pp. 83–99, 2019. Article (CrossRef Link)

[12] W. Wang et al., "PAN++: Towards Efficient and Accurate End-to-End Spotting of Arbitrarily-Shaped Text," *IEEE Trans. Pattern Anal. Mach. Intell.*, no. d, pp. 1–18, 2021. Article (CrossRef Link)

[13] R. Litman, O. Anschel, S. Tsiper, R. Litman, S. Mazor, and R. Manmatha, "Scatter: Selective context attentional scene text recognizer," in *Proc. of IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 11959–11969, 2020. Article (CrossRef Link)

[14] Chen, X., et al, "Text Recognition in the Wild: A Survey," Vol.54, no.2, pp:1-35, 2020. Article (CrossRef Link)

[15] Z. Qiao, Y. Zhou, D. Yang, Y. Zhou, and W. Wang, "SEED: Semantics enhanced encoder-decoder framework for scene text recognition," in *Proc. of IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 13525–13534, 2020. Article (CrossRef Link)

[16] D. Yu et al., "Towards accurate scene text recognition with semantic reasoning networks," in *Proc. of IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2, pp. 12110–12119, 2020. Article (CrossRef Link)
[17] Z. Cheng, F. Bai, Y. Xu, G. Zheng, S. Pu, and S. Zhou, "Focusing Attention: Towards Accurate Text Recognition in Natural Images," in *Proc. of IEEE Int. Conf. Comput. Vis.*, vol. 2017-Octob, pp. 5086–5094, 2017. Article (CrossRef Link)
[18] L. Q. Zuo, H. M. Sun, Q. C. Mao, R. Qi, and R. S. Jia, "Natural Scene Text Recognition Based on Encoder-Decoder Framework," *IEEE Access*, vol. 7, pp. 62616–62623, 2019. Article (CrossRef Link)
[19] C. Y. Lee and S. Osindero, "Recursive Recurrent Nets with Attention Modeling for OCR in the Wild," in *Proc. of IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, no. 3, pp. 2231–2239, 2016.Article (CrossRef Link)
[20] T. Wang et al., "Decoupled attention network for text recognition," in *Proc. of AAAI 2020 - 34th AAAI Conf. Artif. Intell.*, vol. 34, no. 07, pp. 12216–12224, 2020. Article (CrossRef Link)
[21] J. Baek et al., "What is wrong with scene text recognition model comparisons? dataset and model analysis," in *Proc. of IEEE Int. Conf. Comput. Vis.*, vol. 2019-Octob, pp. 4714–4722, 2019. Article (CrossRef Link)
[22] R. R. Litman et al., "Robust Scene Text Recognition with Automatic Rectification," in *Proc. of IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, no. 1, pp. 4168–4176, Oct. 2019. Article (CrossRef Link)
[23] F. Zhan and S. Lu, "ESIR: End-to-end scene text recognition via iterative image rectification," in *Proc. of IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2019-June, pp. 2054–2063, 2019. Article (CrossRef Link)
[24] C. Luo, L. Jin, and Z. Sun, "MORAN: A Multi-Object Rectified Attention Network for scene text recognitiosn," *Pattern Recognit.*, vol. 90, pp. 109–118, 2019. Article (CrossRef Link)
[25] Z. Cheng, Y. Xu, F. Bai, Y. Niu, S. Pu and S. Zhou, "AON: Towards Arbitrarily-Oriented Text Recognition," in *Proc. of 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5571-5579, 2018. Article (CrossRef Link)
[26] H. Li, P. Wang, C. Shen, and G. Zhang, "Show, attend and read: A simple and strong baseline for irregular text recognition," in *Proc. of 33rd AAAI Conf. Artif. Intell. AAAI 2019, 31st Innov. Appl. Artif. Intell. Conf. IAAI 2019 9th AAAI Symp. Educ. Adv. Artif. Intell. EAAI 2019*, pp. 8610–8617, 2019.Article (CrossRef Link)
[27] Q. Lin, C. Luo, L. Jin, and S. Lai, "STAN: A sequential transformation attention-based network for scene text recognition," *Pattern Recognit.*, vol. 111, p. 107692, 2021. Article (CrossRef Link)
[28] Q. H. Shang-Hua Gao, "Representative Batch Normalization with Feature Calibration," *Cvpr*, no. Cvpr, pp. 8669–8679, 2021. Article (CrossRef Link)
[29] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic Data and Artificial Neural Networks for Natural Scene Text Recognition," *Eprint Arxiv*, pp. 1–10, 2014. Article (CrossRef Link)
[30] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic Data for Text Localisation in Natural Images," in *Proc. of IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 2315–2324, 2016. Article (CrossRef Link)
[31] Y. Huang and W. Fang, "Deformable Mixed Domain Attention Network for Scene Text Recognition," in *Proc. of 2020 IEEE 11th International Conference on Software Engineering and Service Science (ICSESS)*, vol. 2020-Octob, pp. 142–145, Oct. 2020. Article (CrossRef Link)
[32] M. Opitz, M. Diem, S. Fiel, F. Kleber, and R. Sablatnig, "End-to-end text recognition using local ternary patterns, MSER and deep convolutional nets," in *Proc. of 11th IAPR Int. Work. Doc. Anal. Syst. DAS 2014*, pp. 186–190, 2014. Article (CrossRef Link)
[33] S. M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, and R. Young, "ICDAR 2003 ro-bust read ing competitions," in *Proc. of Int. Conf. Doc. Anal. Recognition, ICDAR*, vol. 2003-Janua, pp. 6 82–687, 2003. Article (CrossRef Link)
[34] D. Karatzas et al., "ICDAR 2013 robust reading competition," in *Proc. of Int. Conf. Doc. Anal. Recognition, ICDAR*, pp. 1484–1493, 2013. Article (CrossRef Link)

[35] D. Karatzas et al., "ICDAR 2015 competition on Robust Reading," in *Proc. of Int. Conf. Doc. Anal. Recognition, ICDAR*, vol. 2015-Novem, pp. 1156–1160, 2015. Article (CrossRef Link)
[36] T. Q. Phan, P. Shivakumara, S. Tian, and C. L. Tan, "Recognizing text with perspective distortion in natural scenes," in *Proc. of IEEE Int. Conf. Comput. Vis.*, pp. 569–576, 2013. Article (CrossRef Link)
[37] A. Risnumawan, P. Shivakumara, C. S. Chan, and C. L. Tan, "A robust arbitrary text detection system for natural scene images," *Expert Syst. Appl.*, vol. 41, no. 18, pp. 8027–8048, 2014. Article (CrossRef Link)

Yajie Sun is currently an associate professor in Nanjing University of Information Science and Technology, Nanjing, China. She received Ph.D. degree in test measurement technology and instrument engineering from Nanjing University of Aeronautics and Astronautics. Her research interests include structural health monitoring, signal processing and big data.

Xiaoling Cao received the B.S. degree in internet of things engineering from Chengdu Technological University, China, in 2020. She is now pursuing a M.S. degree in electronic Information at Nanjing University of Information Science & Technology, Nanjing, JiangSu, China. Her research interest is Image Processing and Pattern Recognition.

Yingying Sun received the B.S. degree in Computer Science and Technology from Binjiang College Nanjing University of Information Science and Technology, China, in 2020. She is now pursuing a M.S. degree in electronic Information at Nanjing University of Information Science & Technology, Nanjing, JiangSu, China. Her research interests are Image Processing and Deep Learning.