IBN-STR: A Robust Text Recognizer for Irregular Text in Natural Scenes

Xiaoqian Li[*1,2], Jie Liu[*1], Guixuan Zhang[†1] , Shuwu Zhang[1]
[1]Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
Email: {lixiaoqian2015, jie.liu, guixuan.zhang, shuwu.zhang}@ia.ac.cn

***Abstract*—Although text recognition methods based on deep neural networks have promising performance, there are still challenges due to the variety of text styles, perspective distortion, text with large curvature, and so on. To obtain a robust text recognizer, we have improved the performance from two aspects: data aspect and feature representation aspect. In terms of data, we transform the input images into S-shape distorted images in order to increase the diversity of training data. Besides, we explore the effects of different training data. In terms of feature representation, the combination of instance normalization and batch normalization improves the model's capacity and generalization ability. This paper proposes a robust scene text recognizer IBN-STR, which is an attention-based model. Through extensive experiments, the model analysis and comparison have been carried out from the aspects of data and feature representation, and the effectiveness of IBN-STR on both regular and irregular text instances has been verified. Furthermore, IBN-STR is an end-to-end recognition system that can achieve state-of-the-art performance.**

I. Introduction

Text is a vital cue in natural scene images. Text recognition is a branch of computer vision, which can help people understand the content of images and can be widely used as auxiliary aids in intelligent transportation, auxiliary translation, image retrieval, and so on. Research on text recognition has a long history. Traditional pattern classification provides many solutions for text recognition [1]–[3], and due to the development of deep learning and computational power, text recognition has recently achieved great breakthroughs.

Although text recognition methods [4]–[6] based deep learning perform well, there are still challenges in text recognition due to the large curvature of text instances, the variety of text styles, similar characters, occlusion, uneven illumination, and shooting environments. Therefore, a robust text recognizer is of great significance.

Most text recognizers [4]–[7] are trained on synthetic data and evaluated on real data. As we can see in Figure 1, the curvature of text in commonly used synthetic datasets is less volatile, but the curvature of text in real images is greater and the appearance of the text will be more variable. This means that there is a gap between the distribution of training data and test data. In view of the above problems, we first consider improving from the data aspect. It is necessary to use data augmentation to increase the diversity of training data.

(a) Real text (b) Synthetic text

Fig. 1. Various text.

Secondly, for the feature representation aspect, we attempt to mine a robust and efficient module to extract features.

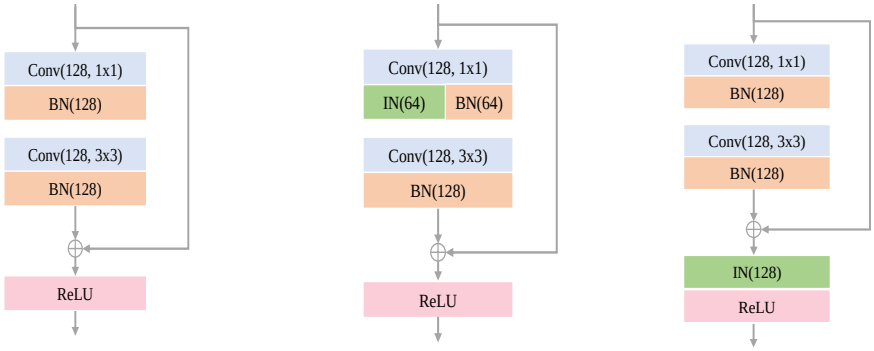
For the data aspect, we utilize S-shape distortion to enrich the text curvature of the training data. Considering the contribution of instance normalization (IN) in style transfer task [8], [9], IN can introduce appearance invariance, and batch normalization (BN) can preserve content information. For the feature representation aspect, a robust instance-batch normalization (IBN) module is proposed to introduce text appearance invariance and improve performance.

In this paper, we propose a Scene Text Recognizer with Instance-Batch Normalization module (named IBN-STR) to achieve regular text and irregular text recognition in natural scenes. The contributions of this paper are as follows:

- In terms of data, we demonstrate the impact of data augmentation and different training data on text recognition. The input images will be S-shape distorted to increase the diversity of training data and further improve performance.
- In terms of feature representation, instance normalization is introduced into text recognition for the first time to improve the model's capacity and generalization ability. The IBN module combines instance normalization with batch normalization, which helps the model extract more effective feature maps and it is effective for both regular text and irregular text.
- With the rectification network, we propose an IBN-STR model for text recognition and achieve state-of-the-art performance.

The remainder of this paper is organized as follows. Section

* Authors contributed equally as first author.
† Corresponding author: Guixuan Zhang.



(a) origianal BN (b) IBN_a module (c) IBN_b module

Fig. 2. Instance-batch normalization (IBN) module.

II introduces the related works of text recognition. Section III illustrates the proposed method. Section IV demonstrates the experimental results to verify the effectiveness of the proposed method. Section V will make a summary of this paper.

II. Related Works

Traditional text recognition methods mainly rely on manual design features such as connected components [10], stroke width transform [11]–[13]. Recently, methods based on deep neural networks have shown advantages in text recognition. Jaderberg *et al.* [14] proposed a model combining convolutional neural network (CNN) and conditional random field, which is optimized by structured output loss. Most CNN-based methods tend to treat text recognition as a sequence recognition task. Inspired by speech recognition, CRNN [8] introduced CTC loss into text recognition. CRNN is an end-to-end system that utilizes CNN and recurrent neural network (RNN) to generate character sequence features. Vanilla CTC can only deal with 1D probability distributions, while 2D-CTC [13] can compute the conditional probability of labels from 2D distributions, which is suitable for irregular text. With the popularity of attention mechanism, attention-based text recognition methods are proposed [5], [6], [15]. RARE [5], ASTER [6], MORAN [7], and RCN [16] transformed irregular text images into rectified images, and then used the attention-based recognition network of an encoder-decoder framework to achieve text recognition. In the absence of a rectification network, Lyu *et al.* [17] proposed a relation attention module and a parallel attention module to transform text images into character feature sequences, which can be workable for irregular text recognition.

Our method is based on the attentional sequence-to-sequence (seq2seq) model. Different from previous methods, the proposed IBN-STR introduces IN for the first time to improve the capacity and generalization ability of text recognizer.

III. Method

As shown in Figure 3, the proposed IBN-STR model consists of a rectification network and a text recognition network. The rectification network is based on the spatial transformer

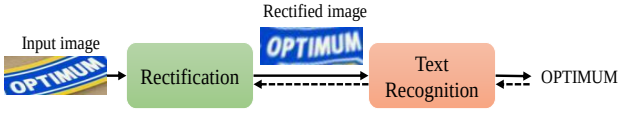


Fig. 3. Overview of text recognizer. The dashed lines mean the direction of gradient propagation.

network and generates rectified images. The text recognition network follows the encoder-decoder framework which is widely used in seq2seq text recognition. It consists of a CNN-BLSTM encoder and an attention-based decoder. The encoder first extracts stacked convolutional features of input images and utilizes bidirectional long short-term memory (BLSTM) to convert the image features into feature sequences. The decoder is a sequence-to-sequence model that translates the feature sequence into a character sequence. The IBN module is embedded in the stacked convolutional modules to improve the capacity and generalization ability of text recognizer.

A. IBN Module

Batch normalization [18] is proposed to normalize data and preserve the representations of data. BN enables the model less sensitive to parameters and converges faster by limiting the input data to a certain range through the mean and variance.

Given the feature map $x \in \mathbb{R}^{N\times C\times H\times W}$ with N samples, C channels, H height, and W width. The normalized data can be formulated as

$$x^{'} = \gamma(\frac{x-\mu(x)}{\sigma(x)}) + \beta. \tag{1}$$

where γ, β are scaling and shift factors. BN retains the channel dimension when calculating the mean and variance:

$$\mu_c(x) = \frac{1}{NHW}\sum_{n=1}^{N}\sum_{h=1}^{H}\sum_{w=1}^{W} x_{nchw}, \tag{2}$$

$$\sigma_c(x) = \sqrt{\frac{1}{NHW}\sum_{n=1}^{N}\sum_{h=1}^{H}\sum_{w=1}^{W}(x_{nchw}-\mu_c(x))^2 + \epsilon}. \tag{3}$$

In the training phase, the mean/variance is calculated base on samples of the mini-batch. All means and variance of mini-batches during training will be saved for testing. In the test phase, the mean/variance of each mini-batch from the training phase will be weighted average and obtain the estimated value.

Instance normalization [19] is mainly used in the field of style transfer [8], [9] because IN can learn features that are invariant to styles or appearance. For style transfer, each image should be regarded as a domain. In order to maintain the independence between different image instances, IN retains the dimensions of N and C, and only operates of averaging and standard deviation for H and W within the channel. The mean and variance can be formulated as:

$$\mu_{nc}(x) = \frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W} x_{nchw}, \tag{4}$$

$$\sigma_{nc}(x) = \sqrt{\frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W}(x_{nchw} - \mu_{nc}(x))^2 + \epsilon}. \tag{5}$$

Texts in natural scenes are very variable in appearances. Inspired by the contribution of Instance normalization to style transfer tasks [8], [9], we introduce IN to obtain a robust text recognizer in natural scenes. As shown in Figure 2, two types of IBN modules are provided. In the IBN_a module, the feature maps are divided into two parts and sent to IN and BN respectively, and the outputs of IN and BN will be concatenated and sent to the next convolutional layer. IBN_a module can integrate appearance invariant features and content related information to improve the performance. To explore the generalization ability of the different types of IBN modules, the IBN_b module is proposed. The IN layer will be placed before the residual block output.

The experimental results in section IV verify the effectiveness of the proposed IBN module. Particularly, most of the time IBN_a module performs better than the IBN_b module in text recognition and improves regular and irregular text recognition.

TABLE I
ARCHITECTURE OF TEXT RECOGNITION NETWORK. BLSTM MEANS BIDIRECTIONAL LONG SHORT-TERM MEMORY LAYER.

	Layers	Configurations		Outsize
encoder	Block 0	3×3 $conv, s$ $1 \times 1, bn$		$32 \times 32 \times 100$
	Block 1	$\begin{matrix}1 \times 1\ conv, 32, bn \\ 3 \times 3\ conv, 32, bn\end{matrix}$	$\times 3, s$ 2×2	$32 \times 16 \times 50$
	Block 2	$\begin{matrix}1 \times 1\ conv, 64, ibn \\ 3 \times 3\ conv, 64, bn\end{matrix}$	$\times 4, s$ 2×2	$64 \times 8 \times 25$
	Block 3	$\begin{matrix}1 \times 1\ conv, 128, ibn \\ 3 \times 3\ conv, 128, bn\end{matrix}$	$\times 6, s$ 2×1	$128 \times 4 \times 25$
	Block 4	$\begin{matrix}1 \times 1\ conv, 256, ibn \\ 3 \times 3\ conv, 256, bn\end{matrix}$	$\times 6, s$ 2×1	$256 \times 2 \times 25$
	Block 5	$\begin{matrix}1 \times 1\ conv, 512, bn \\ 3 \times 3\ conv, 512, bn\end{matrix}$	$\times 3, s$ 2×1	$512 \times 1 \times 25$
	BLSTM1	256 hidden units		25×256
	BLSTM2	256 hidden units		25×256
decoder	GRU	256 hidden units		25×256

B. Encoder

The encoder aims to extract rich and discriminative features. As illustrated in Table I, the main structure of the encoder is the CNN-BLSTM framework. The encoder first extracts spatial feature maps from the input image through stacked convolutional layers with residual connections [20]. The proposed IBN module can be employed in the shallow layers to obtain strong spatial features. Based on ResNet45 [21], the proposed IBN-STR model uses the IBN module in the residual Block 2 to residual Block 4.

The CNN of encoder aims to capture the features of local regions. To capture the long-range dependencies of characters, a multi-layer bidirectional long short-term memory [22] is introduced. BLSTM can encode feature sequences bidirectionally, capture long-range dependencies in both directions, and model global context information, thereby leveraging richer context and improving performance.

C. Decoder

The decoder is attention-based which can achieve sequence to sequence prediction. As shown in Table I, GRU cell [23] is utilized to decode output dependencies. Through T iterations, the decoder generates a predicted symbol sequence $(y_1, ..., y_T)$, where T is the number of characters. To generate a variable-length sequence, a special end-of-sequence symbol (EOS) is inserted at the end of the target sequence. At step t, the decoder produces a predicted output y_t and the probability of y_t is $p(y_t)$:

$$\begin{aligned} p(y_t) &= Softmax(W_{out}s_t + b_{out}), \\ y_t &\sim p(y_t), \end{aligned} \tag{6}$$

where s_t is the hidden state at the current time, and W_{out} and b_{out} are trainable parameters. In this paper, the embedding vectors and s_{t-1} (the hidden state at the previous time) will be fed into GRU to update s_t:

$$s_t = GRU(s_{t-1}, (g_t, f(y_{t-1}))), \tag{7}$$

$$g_t = \sum_{i=1}^{L}(\alpha_{t,i}h_i), \tag{8}$$

where $(g_t, f(y_{t-1}))$ is the concatenation of g_t glimpse vectors and $f(y_{t-1})$ embedding vectors of the previous output y_{t-1}. The glimpse vectors focus on a small part of the whole context. In the formula 8, L is the length of feature maps; $\alpha_{t,i}$ is the attentional weights vector and it can be generated by

$$\alpha_{t,i} = exp(e_{t,i}) / \sum_{j=1}^{L}(exp(e_{t,j})), \tag{9}$$

$$e_{t,i} = Tanh(Ws_{t-1} + Vh_i + b), \tag{10}$$

where W, V and b are trainable parameters.

Given the predicted symbol sequence, the recognition loss can be formulated as

$$L_{rec} = -\frac{1}{T}\sum_{t=1}^{T}(logp_{l2r}(y_t) + logp_{r2l}(y_t)), \tag{11}$$

where $p_{l2r}(y_t)$ and $p_{r2l}(y_t)$ are the probabilities of the sequence from left to right and from right to left.

D. Rectification

The rectification network is based on the spatial transformer network [24], similar to RARE [5]. First, fiducial points are predicted by the localization network, and then thin-plate-spline transformation [25] matrices will be calculated to generate the sampling grid. Finally, the sampler uses bilinear interpolation to obtain the rectified image. Table II illustrates the architecture of the localization network. The input image is scaled to 32×100 and fed into convolutional layers and pooling layers. Each convolution layer is followed by a batch normalization layer and a ReLU layer. The adaptive average pooling layer is used to generate feature vectors and then feature vectors pass through 2 fully connected layers to generate predicted fiducial points.

E. Data augmentation

To enrich the diversity of training data, it is necessary to adopt data augmentation for input images. Here, we utilize the trigonometric function to generate S-shape distortion transformation. Given the position of original image (i, j) and the position of rectified image $(i^{'}, j^{'})$, the correspondences of between (i, j) and $(i^{'}, j^{'})$ are as follows:

$$\begin{aligned} i^{'} &= a_1 i + a_2 Sin(\theta, j) + a_3, \\ j^{'} &= j, \end{aligned} \tag{12}$$

where a_1, a_2, a_3 are scaling and shifting parameter. θ determines the distortion mode for the entire image. In this paper, the original image is S-shape distorted with a probability of 0.4. As shown in Figure 4, there are 16 distortion modes, one of which will be randomly selected as the input image. The experimental results demonstrate the effectiveness of S-shape distortion.

The proposed method focuses on alphanumeric character recognition, but non-alphanumeric characters occur frequently in text images of natural scenes. Therefore, this paper also discusses whether to use non-alphanumeric text images in section IV.

TABLE II
ARCHITECTURE OF THE LOCALIZATION NETWORK. CONV MEANS CONVOLUTION LAYER, MP MEANS MAXPOOLING LAYER AND ADAPAVGP MEANS ADAPTIVE AVERAGE POOLING LAYER.

Layers	Configurations	Outsize
Conv 1	3×3 $conv, 64, s$ 1×1	$64 \times 32 \times 100$
MP 1	2×2	$64 \times 16 \times 50$
Conv 2	3×3 $conv, 128, s$ 1×1	$128 \times 16 \times 50$
MP 2	2×2	$128 \times 8 \times 25$
Conv 3	3×3 $conv, 256, s$ 1×1	$256 \times 8 \times 25$
MP 3	2×2	$256 \times 4 \times 12$
Conv 4	3×3 $conv, 512, s$ 1×1	$512 \times 4 \times 12$
AdapAvgP	1×1	$512 \times 1 \times 1$
Linear 1	$512, 256$	256
Linear 2	$256, 2K$	$2K$

(a) origin image (b) distorted images

Fig. 4. S-shape distortion.

IV. EXPERIMENTS

In this section, we conduct extensive experiments to verify the effectiveness of the proposed method. The performances of all the methods are measured by word accuracy.

A. Benchmark Datasets

- **Street View Text (SVT).** Street View Text dataset [1] has 350 images collected from Google street view. The dataset has 647 word instances, and each instance has a 50-word lexicon.
- **IIIT5K-Words (IIIT5K).** The IIIT5K-word dataset [26] has 3,000 cropped word instances for testing. The dataset provides a 50-word and a 1k-word lexicons for each word instance.
- **ICDAR 2013 (IC13).** ICDAR 2013 [27] has 1,095 word instances cropped from 233 scene images. After filtering words with non-alphanumeric characters, 1,015 cropped word instances are obtained for evaluation.
- **ICDAR 2015 (IC15).** ICDAR 2015 [28] provides 2,077 word instances in multi-oriented for text recognition. The word instances are cropped from test scene images. Remove non-alphanumeric characters, words with a length less than 3 and irregular text will obtain 1,811 word instances.
- **SVT-Perspective (SVT-P).** SVT-Perspective dataset [29] has 639 perspective text instances and a 50-lexicon is provided for each instance.
- **CUTE80 (CUTE).** CUTE80 dataset [30] has 288 word instances cropped from 80 high-resolution images taken in natural scenes. The dataset has many examples of curved text.
- **Total-Text.** Total-Text [31] has 300 test images. The word instances are arbitrary shape text, including flipped text. 2,204 word instances are obtained after filtering words with non-alphanumeric characters.

The benchmarks consist of regular texts and irregular text. There are 4,662 regular text instances from SVT, IIIT5K and IC13 datasets, and 5,214 irregular text instances from IC15, SVT-P, CUTE, and Total-text datasets. The total number of text instances is 9,876.

B. Implementation Details

In this paper, we utilize Synth90k [32] and SynthText [33] as training data and evaluate the standard benchmarks. Synth90k dataset (denoted as **SK**) contains approximately 8.9 million synthetic word images and SynthText dataset (denoted as **ST**) has 6.9 million training data, including 1.4 million non-alphanumeric instances. For SynthText, 5.5 million word instances (denoted as **ST_a**) will be obtained by filtering words with non-alphanumeric characters, while 1.4 million non-alphanumeric word instances will be denoted as **ST_e**. The proposed model is trained using only synthetic data, without fine-tuning. The model only recognizes the alphanumeric characters including 26 letters, 10 digits, a symbol for non-alphanumeric characters, and a symbol standing for 'EOS'.

The model is trained from scratch and optimized by Adam optimizer with a learning rate of 5e-4. Iteration stops after 10 epochs. All the input images are resized to 32×100. The experiments are conducted with two NVIDIA Tesla K40 GPUs and the batch size is 1024.

TABLE III
THE RESULTS OF DATA AUGMENTATION.

Method	Regular	Irregular	Total
Base(BO+37)	90.20	72.61	80.91
Base-stn(BO+37)	90.78	75.76	82.85
Base(TO+38)	92.32	74.41	82.87
Base-stn(TO+38)	93.07	76.89	84.53
Data-base(BO+37)	90.35	72.75	81.06
Data-base-stn(BO+37)	91.30	75.93	83.18
Data-base(TO+37)	92.94	75.07	83.51
Data-base-stn(TO+37)	**93.35**	77.94	**85.22**
Data-base(TO+38)	92.53	75.49	83.54
Data-base-stn(TO+38)	93.22	**78.02**	85.20
Improvement	Regular	Irregular	Total
S-shape(BO+37)	+0.15	+0.13	+0.15
S-shape-stn(BO+37)	+0.52	+0.17	+0.33
S-shape(TO+38)	+0.21	+1.07	+0.67
S-shape-stn(TO+38)	+0.15	+1.13	+0.67
Data(TO37-BO37)	+2.59	+2.32	+2.45
Data-stn(TO37-BO37)	+2.06	+2.01	+2.04
Char(TO38-TO37)	-0.41	+0.42	+0.03
Char-stn(TO38-TO37)	-0.13	+0.08	-0.02

C. Ablation Study

1) Data Augmentation: Here we attempt to display the effects of the different training datasets, S-shape distortion, and the outputs (including a symbol for non-alphanumeric characters). As shown in Table III, BO means using SK + ST_a datasets, while TO means using SK+ST (SK + ST_a + ST_e) datasets. 37 indicates an output without considering non-alphanumeric characters, while 38 indicates an output including a symbol for non-alphanumeric characters. Base-* model is trained by images without S-shape distortion, but the inputs of Data-Base-* are S-shape distorted. All the models with *-stn are trained without the rectification network.

TABLE IV
THE RESULTS OF DIFFERENT IBN MODULES.

Method	Regular	Irregular	Total
Base	92.32	74.41	82.87
Base-stn	93.07	76.89	84.53
Base-ibn-a	$92.40^{+0.08}$	$74.90^{+0.48}$	$83.16^{+0.29}$
Base-ibn-b	$92.15^{-0.17}$	$73.80^{-0.61}$	$82.84^{-0.41}$
Basestn-ibn-a	$92.96^{-0.11}$	$77.67^{+0.78}$	$84.89^{+0.36}$
Basestn-ibn-b	$92.60^{-0.47}$	$76.37^{-0.52}$	$84.03^{-0.50}$
DataBase	92.53	75.49	83.54
DataBase-stn	93.22	**78.02**	85.20
DataBase-ibn-a	$92.65^{+0.11}$	$76.39^{+0.90}$	$84.06^{+0.52}$
DataBase-ibn-b	$92.60^{+0.07}$	$75.72^{+0.23}$	$83.69^{+0.15}$
DataBasestn-ibn-a	$93.16^{-0.06}$	$77.50^{-0.52}$	$84.89^{-0.31}$
DataBasestn-ibn-b	$\mathbf{93.48}^{+0.25}$	$77.87^{-0.16}$	$\mathbf{85.24}^{+0.04}$

TABLE V
THE RESULTS OF DIFFERENT NUMBER OF IBN LAYERS.

Method	Regular	Irregular	Total
BN	92.53	75.49	83.54
IBN_a, 2	$92.66^{+0.13}$	$76.04^{+0.55}$	$83.89^{+0.35}$
IBN_a, 1-2	$\mathbf{93.03}^{+0.50}$	$75.87^{+0.38}$	$83.97^{+0.43}$
IBN_a, 2-3	$92.90^{+0.37}$	$75.85^{+0.36}$	$83.90^{+0.36}$
IBN_a, 2-4	$92.92^{+0.39}$	$\mathbf{76.97}^{+1.48}$	$\mathbf{84.50}^{+0.96}$
IBN_a, 1-4	$92.65^{+0.12}$	$76.39^{+0.90}$	$84.06^{+0.52}$

The top half of Table III demonstrates the results of the regular text and irregular text recognition, and the bottom half shows the performance improvement. Obviously, the S-shape distortion and ST_e dataset greatly promote performance. Outputting non-alphanumeric symbol has a relatively small impact on the performance of overall data. The output of 38 classes will damage the text recognition of regular datasets and promote the text recognition of irregular datasets. According to the above analysis, we take the Base(TO+38) model as the base model in the following.

2) IBN Module: We discuss the effects of different versions of the IBN module and the number of IBN module layers on text recognizer. We utilize ResNet45 [21] as the backbone which consists of 5 residual modules with batch normalization. According to [9], the batch normalization layers are replaced by IBN in the shallow layers.

When we compare the effects of two IBN modules (IBN_a module and IBN_b module), the first 4 residual blocks with batch normalization layer are replaced by IBN modules. As illustrated in Table IV, we use the model with only batch normalization as the baseline (denoted by Base, Base-stn, DataBase, and DataBase-stn). All the models are trained by SK and ST datasets. Without S-shape distortion, the IBN_a module can always help improve performance, while the IBN_b module degrades the performance. With S-shape distortion, IBN_a module can help improve the performance of the DataBase-ibn-a model but make Databasestn-ibn-a model slightly worse. As for the IBN_b module, it can help the DataBase*-ibn-b model to improve the performance on overall data.

In addition, we also compare the impact of the number of

TABLE VI
COMPARISON OF OTHER TEXT RECOGNITION METHODS. * MEANS USING 1,811 IMAGES.

Method	Data	Regular						Irregular					Total
		IC13	SVT		IIIT5K			IC15	SVT-P		CUTE	Total-text	
		None	None	50	None	50	1k	None	None	50	None	None	
CRNN [4]	SK	89.6	82.7	97.5	81.2	97.8	95.0	-	-	-	-	-	-
GCRNN [34]	SK	-	81.5	96.3	80.8	98.0	95.6	-	-	-	-	-	-
R2AM [15]	SK	90.0	80.7	96.3	78.4	96.8	94.4	-	-	-	-	-	-
Liao et.al [35]	ST	91.4	82.1	98.5	92.0	99.8	98.9	-	-	-	78.1	-	
Aster [6]	ST+SK	91.8	**93.6**	**99.2**	93.4	99.6	98.8	76.1*	78.5	-	79.5	-	-
2D CTC [36]	ST+SK	93.9	90.6	97.2	**94.7**	**99.8**	98.9	75.2*	79.2	-	81.3	63.0	-
RCN [16]	ST+SK	93.2	88.6	97.7	94.0	99.6	98.9	77.1	80.6	**95.0**	**88.5**	-	-
MORAN [7]	ST+SK	92.4	88.3	96.6	91.2	97.9	96.2	68.8	76.1	94.3	77.4	-	-
Lyu et.al [17]	ST+SK	92.7	90.1	97.2	94.0	**99.8**	**99.1**	76.3	82.3	-	86.8	-	-
IBN-STR(base)	ST+SK	93.8	90.0	97.3	93.3	99.5	98.7	77.8	83.6	**95.0**	84.4	73.3	84.5
IBN-STR(stn)	ST+SK	**94.7**	91.0	98.0	94.0	**99.8**	98.6	**79.1**	**85.1**	94.6	85.4	**74.8**	85.6

IBN layers. As shown in Table V, BN is the configurations for the DataBase model. Different IBN_a module layers all promote performance.

According to the above analysis, the proposed IBN module can improve text recognition. And overall IBN_a module is better than IBN_b module in performance improvement. Performance improvement in irregular text is more than that in regular text. In addition, IBN_a module does not increase computational cost.

D. Comparisons with the State-of-the-arts

The proposed IBN-STR model is trained by ST + SK datasets, and the outputs are 38 classes, including a non-alphanumeric symbol recognition. The input images will be S-shape distorted and fed into the IBN-STR model. In the final model IBN-STR, the IBN_a modules are utilized in Block 2 to Block 4 of Table I. We compare the performance of our model and other state-of-the-arts in Table VI. IBN-STR(base) is trained without the rectification network, and IBN-STR(stn) is trained with the rectification network. Compared with rectification-based methods Aster [6] and MORAN [7], our method performs better on IC13, IIIT5K, IC15, SVT-P and CUTE datasets. In addition, on Total-text with complex text instances, the performance of our model is significantly improved, which is 10.3%-11.8% higher than the previous model.

V. CONCLUSION

In this paper, we consider the data aspect and feature representation aspect to improve the generalization of the model. S-shape distortion is utilized to enrich the diversity of training data and the effect of data augmentation on text recognition is analyzed. In addition, the combination of instance normalization and batch normalization improves the model's capacity and generalization ability. The IBN-STR model is proposed to achieve text recognition and can compete with the state-of-the-arts. Experimental results show the effectiveness of the proposed method. Although the proposed method can perform well on the regular text and irregular text recognition, our method cannot handle vertical or flipped text instances. Future research will focus on a flexible text recognizer that can process text from various perspectives.

ACKNOWLEDGMENT

This work is supported by the National Key R&D Program of China (2018YFB1403900) and the Science and Technology Program of Beijing (Z201100001820002). It was also the research achievement of the Key Laboratory of Digital Rights Services, which is one of the National Science and Standardization Key Labs for Press and Publication Industry.

REFERENCES

[1] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *2011 International Conference on Computer Vision*. IEEE, 2011, pp. 1457–1464.

[2] A. Bissacco, M. Cummins, Y. Netzer, and H. Neven, "Photoocr: Reading text in uncontrolled conditions," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 785–792.

[3] L. Neumann and J. Matas, "Real-time scene text localization and recognition," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 3538–3545.

[4] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 11, pp. 2298–2304, 2016.

[5] B. Shi, X. Wang, P. Lyu, C. Yao, and X. Bai, "Robust scene text recognition with automatic rectification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4168–4176.

[6] B. Shi, M. Yang, X. Wang, P. Lyu, C. Yao, and X. Bai, "Aster: An attentional scene text recognizer with flexible rectification," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 9, pp. 2035–2048, 2018.

[7] C. Luo, L. Jin, and Z. Sun, "Moran: A multi-object rectified attention network for scene text recognition," *Pattern Recognition*, vol. 90, pp. 109–118, 2019.

[8] H. Nam and H.-E. Kim, "Batch-instance normalization for adaptively style-invariant neural networks," in *Advances in Neural Information Processing Systems*, 2018, pp. 2558–2567.

[9] X. Pan, P. Luo, J. Shi, and X. Tang, "Two at once: Enhancing learning and generalization capacities via ibn-net," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 464–479.

[10] L. Neumann and J. Matas, "Real-time scene text localization and recognition," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3538–3545.

[11] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2963–2970.

[12] C. Yao, X. Bai, and W. Liu, "A unified framework for multioriented text detection and recognition," *IEEE Transactions on Image Processing*, vol. 23, no. 11, pp. 4737–4749, 2014.

[13] C. Yao, X. Bai, W. Liu, Y. Ma, and Z. Tu, "Detecting texts of arbitrary orientations in natural images," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 1083–1090.

[14] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep structured output learning for unconstrained text recognition," *arXiv preprint arXiv:1412.5903*, 2014.
[15] C.-Y. Lee and S. Osindero, "Recursive recurrent nets with attention modeling for ocr in the wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2231–2239.
[16] Y. Gao, Y. Chen, J. Wang, Z. Lei, X.-Y. Zhang, and H. Lu, "Recurrent calibration network for irregular text recognition," *arXiv preprint arXiv:1812.07145*, 2018.
[17] P. Lyu, Z. Yang, X. Leng, X. Wu, R. Li, and X. Shen, "2d attentional irregular scene text recognizer," *arXiv preprint arXiv:1906.05708*, 2019.
[18] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.
[19] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6924–6932.
[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
[21] A. Mishra, K. Alahari, and C. Jawahar, "Enhancing energy minimization framework for scene text recognition with top-down cues," *Computer Vision and Image Understanding*, vol. 145, pp. 30–42, 2016.
[22] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 5, pp. 855–868, 2008.
[23] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.
[24] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *Advances in neural information processing systems*, 2015, pp. 2017–2025.
[25] F. L. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 11, no. 6, pp. 567–585, 1989.
[26] A. Mishra, K. Alahari, and C. V. Jawahar, "Scene text recognition using higher order language priors," in *BMVC*, 2012.
[27] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. G. i Bigorda, S. R. Mestre, J. Mas, D. F. Mota, J. A. Almazan, and L. P. De Las Heras, "Icdar 2013 robust reading competition," in *2013 12th International Conference on Document Analysis and Recognition*. IEEE, 2013, pp. 1484–1493.
[28] D. Karatzas, L. Gomez-Bigorda, A. Nicolaou, S. Ghosh, A. Bagdanov, M. Iwamura, J. Matas, L. Neumann, V. R. Chandrasekhar, S. Lu *et al.*, "Icdar 2015 competition on robust reading," in *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 2015, pp. 1156–1160.
[29] T. Quy Phan, P. Shivakumara, S. Tian, and C. Lim Tan, "Recognizing text with perspective distortion in natural scenes," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 569–576.
[30] A. Risnumawan, P. Shivakumara, C. S. Chan, and C. L. Tan, "A robust arbitrary text detection system for natural scene images," *Expert Systems with Applications*, vol. 41, no. 18, pp. 8027–8048, 2014.
[31] C. K. Ch'ng and C. S. Chan, "Total-text: A comprehensive dataset for scene text detection and recognition," in *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, vol. 1. IEEE, 2017, pp. 935–942.
[32] M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman, "Synthetic data and artificial neural networks for natural scene text recognition," *arXiv preprint arXiv:1406.2227*, 2014.
[33] A. Gupta, A. Vedaldi, and A. Zisserman, "Synthetic data for text localisation in natural images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2315–2324.
[34] J. Wang and X. Hu, "Gated recurrent convolution neural network for ocr," in *Advances in Neural Information Processing Systems*, 2017, pp. 335–344.
[35] M. Liao, J. Zhang, Z. Wan, F. Xie, J. Liang, P. Lyu, C. Yao, and X. Bai, "Scene text recognition from two-dimensional perspective," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 8714–8721.
[36] Z. Wan, F. Xie, Y. Liu, X. Bai, and C. Yao, "2d-ctc for scene text recognition," *arXiv preprint arXiv:1907.09705*, 2019.