Learn to Augment: Joint Data Augmentation and Network Optimization for Text Recognition

Canjie Luo[1], Yuanzhi Zhu[1], Lianwen Jin[1]*, Yongpan Wang[2]
[1]South China University of Technology, [2]Alibaba Group
{canjie.luo, zzz.yuanzhi, lianwen.jin}@gmail.com, yongpan@taobao.com

Abstract

Handwritten text and scene text suffer from various shapes and distorted patterns. Thus training a robust recognition model requires a large amount of data to cover diversity as much as possible. In contrast to data collection and annotation, data augmentation is a low cost way. In this paper, we propose a new method for text image augmentation. Different from traditional augmentation methods such as rotation, scaling and perspective transformation, our proposed augmentation method is designed to learn proper and efficient data augmentation which is more effective and specific for training a robust recognizer. By using a set of custom fiducial points, the proposed augmentation method is flexible and controllable. Furthermore, we bridge the gap between the isolated processes of data augmentation and network optimization by joint learning. An agent network learns from the output of the recognition network and controls the fiducial points to generate more proper training samples for the recognition network. Extensive experiments on various benchmarks, including regular scene text, irregular scene text and handwritten text, show that the proposed augmentation and the joint learning methods significantly boost the performance of the recognition networks. A general toolkit for geometric augmentation is available[1].

1. Introduction

The last decade witnessed the tremendous progress brought by the deep neural network in the computer vision community [3, 11, 14, 21]. Limited data is not sufficient to train a robust deep neural network, because the network may overfit to the training data and produce poor generalization on the test set [5]. However, data collection and annotation require a lot of resources. Different from single object classification task [21], the annotation work of text string is more tough, because there are multiple characters in a text image. This is also a reason why most state-of-the-art scene text recognition methods [23, 28, 38] only used synthetic samples [13, 17] for training. The data limitation also effects handwritten text recognition. There exists a wide variety of writing styles. Collecting large scale annotated handwritten text image is high-cost and cannot cover all diversities [47]. It is also challenging to generate synthetic data for handwritten text, because it is difficult to imitate various writing styles.

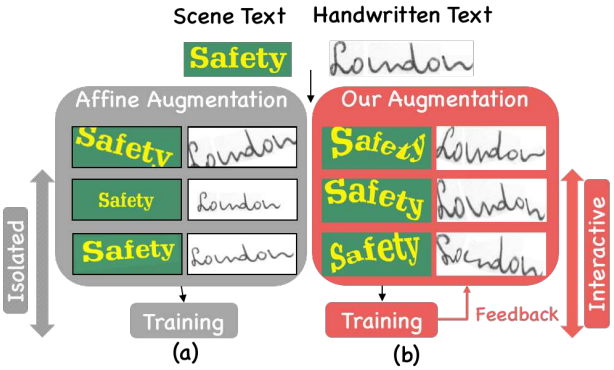

Figure 1. (a) Existing geometric augmentation, including rotation, scaling and perspective transformation; (b) Our proposed flexible augmentation. Moreover, a joint learning method bridges the isolated processes of data augmentation and network training.

To obtain more training samples, it is possible to apply random augmentation to the existing data [9]. Handwritten text with varying writing styles, and scene text with different shapes, such as perspective and curved text, are still very challenging to be recognized [5, 28, 38]. Therefore, geometric augmentation is an important way to gain robustness for recognition methods. As shown in Figure 1 (a), the common geometric transformations are rotation, scaling and perspective transformation. Multiple characters in an image are regarded as one entity, and a global augmentation is performed on the image. However, the diversity of each character should be taken into account. Given a text image, the augmentation goal is to increase the diver-

*Corresponding author.
[1]https://github.com/Canjie-Luo/Text-Image-Augmentation

sity of every character in the text string. Therefore, existing augmentation is limited to the over-simple transformations, which are inefficient for training.

In addition, the effective training samples that contribute to the robustness of the network may still be rare because of the long-tail distribution [31], which is another reason that causes inefficient training. The strategy of random augmentation is the same for every training sample, neglecting the difference among the samples and the optimization procedure of the network. Under the manually controlled static distribution, the augmentation may produce many "easy" samples which are useless for the training. Therefore, random augmentation under the static distribution can hardly meet the requirement of the dynamic optimization. Simultaneously, the manually designed best augmentation strategy on a dataset, usually cannot be transferred to another dataset as expected. Our goal is to study a learnable augmentation method that can automatically adapt to other tasks without any manual modification.

In this paper, we propose a new data augmentation method for text recognition, which is designed for sequence-like characters [36] augmentation. Our augmentation method focuses on the spatial transformation of images. We first initialize a set of fiducial points on the image and then move the points to generate a new image. The moving state, which represents the movement of the points to create "harder" training samples, is sampled from the predicted distribution of the agent network. Then the augmentation module takes the moving state and image as input, and generates a new image. We adopt similarity transformation based on moving least squares [35] for image generation. Besides, a random moving state is also fed to the augmentation module to generate a randomly augmented image. Finally, the agent learns from the moving state that increases recognition difficulty. The difficulty is measured under the metric of edit distance, which is highly relevant to the recognition performance.

To summarize, our contributions are as follows:

- We propose a data augmentation method for text images that contain multiple characters. To the best of our knowledge, this may be the first augmentation method specially designed for sequence-like characters.
- We propose a framework that jointly optimizes the data augmentation and the recognition model. The augmented samples are generated through an automatic learning process, and are thus more effective and useful for the model training. The proposed framework is end-to-end trainable without any fine-tuning.
- Extensive experiments conducted on various benchmarks, including scene text and handwritten text, show that the proposed augmentation and joint learning methods remarkably boost the performance of the recognizers, especially on small training dataset.

2. Related Work

Scene Text Recognition As an essential process in computer vision tasks, scene text recognition has attracted much research interest [22, 23, 28, 38]. There are multiple characters in a scene text image. Thus the text string recognition task is more difficult than single character recognition. Typically, scene text recognition approaches can be divided into two types: localization-based and segmentation-free.

The former attempts to localize the position of characters, recognize them and group all the characters as a text string [41, 42]. The latter benefits from the success of deep neural network and models the text recognition as a sequence recognition problem. For instance, He et al. [15] and Shi et al. [36] applied recurrent neural networks (RNNs) on the top of convolutional neural networks (CNNs) for spatial dependencies of sequence-like objects. Furthermore, the sequence-to-sequence mapping issue was addressed by attention mechanism [38].

The great progress in regular text recognition led the community to irregular text recognition. Luo et al. [28] and Shi et al. [38] proposed rectification networks to remove distortion and decrease recognition difficulty. Zhan and Lu [46] iteratively removed perspective distortion and text line curvature. Yang et al. [43] gave an accurate description of text shape by using more geometric constraints and supervisions for every character. Though the methods above made a notable step forward, irregular scene text recognition still remains a challenging problem.

Handwritten Text Recognition Due to various writing styles, handwritten text recognition is still a challenging field [5]. Early methods used hybrid hidden Markov model [10] and embedded both word images and text strings in a common vectorial subspace to cast recognition tasks as nearest neighbor problems [1].

In the deep learning era, Sueiras et al. [39] and Sun et al. [40] extracted feature by using CNNs followed by RNNs, and obtained superior results. Zhang et al. [47] addressed handwriting style diversity problem by proposing a sequence-to-sequence domain adaptation Network. Bhunia et al. [5] adversarially warped the intermediate feature-space to alleviate the lack of variations in some sparse training datasets. While great progress has been made, handwritten text recognition remains an open and challenging problem because of various writing styles.

Data Augmentation Data augmentation is critical to avoid overfitting in the training of deep neural networks [9, 16, 31]. Nevertheless, few research addresses the augmentation issue for text images. Common geometric augmentations including flipping, rotation, scaling and perspective transformation, are typically useful for single object recognition [21]. However, a text image contains multiple characters. Existing over-simple transformations do not significantly contribute to the diversity of text appearance.

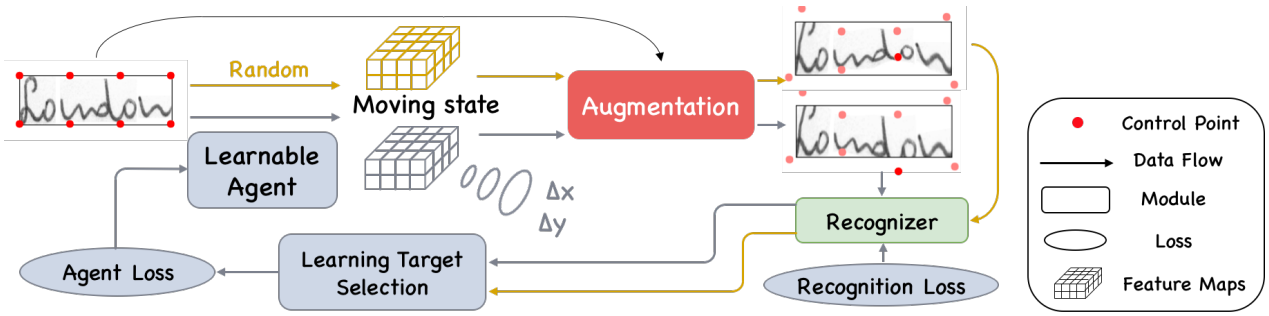

Figure 2. Overview of the proposed framework. First, the learnable agent predicts a distribution of the moving state aiming to create a harder training sample. Then the augmentation module generates augmented samples based on the random and predicted moving state, respectively. The difficulty of the pair of samples is measured by the recognition network. Finally, the agent takes the moving state that increases difficulty as guidance and updates itself. The unified framework is end-to-end trainable.

Simultaneously, the static augmentation policy does not meet the dynamic requirement of optimization. Cubuk et al. [9] searched the policy for augmentation by using reinforcement learning. Ho et al. [16] generated flexible augmentation policy schedules to speed up the searching procedure (5000 GPU hours to 5 GPU hours on CIFAR-10). Peng et al. [31] augmented samples by adversarial learning with pre-training processes.

With respect to text recognition, the training of the recognizer requires much data. The widely used synthetic datasets [13, 17] provide more than 10 million samples. However, Li et al. [22] additionally used approximately 50k public real datasets for training and significantly improved recognition performance, which suggests that the recognition models are still data-hungry. As for handwritten text, existing training data can hardly cover various writing styles and generating synthetic handwritten data is also challenging. Unlike scene text synthesis, there is few font in writing style to render on a canvas.

Our method is proposed for multiple characters augmentation in an automatic manner. An agent network searches hard training samples online. Moreover, the framework is end-to-end trainable without any fine-tuning.

3. Methodology

3.1. Overall Framework

As illustrated in Figure 2, the proposed framework consists of three main modules: an agent network, an augmentation module and a recognition network. First, we initialize a set of custom fiducial points on the image. A moving state predicted by the agent network and a randomly generated moving state are fed to the augmentation module. The moving state indicates the movement of a set of custom fiducial points. Then the augmentation module takes the image as input, and applies transformation based on the moving states respectively. The recognizer predicts text strings on the augmented images. Finally, we measure the recognition difficulty of the augmented images under the metric of edit distance. The agent learns from the moving state that increases difficulty, and explores the weakness of the recognizer. As a result, the recognizer gains robustness from the hard training samples.

As we only use the prediction of the recognition network and the difficulty is measured by edit distance rather than other loss functions, the recognition network can be replaced by recent advanced methods [36, 38], which we will demonstrate in the section 4. In this section, we describe the augmentation module and the joint training scheme of the proposed framework.

3.2. Text Augmentation

Given a text image, the augmentation goal is to increase the diversity of every character in the text string. This motivates us to use more custom fiducial points for transformation. As shown in Figure 3, we averagely divide the image into N patches and initialize $2(N+1)$ fiducial points p along the top and bottom image borders. After that, we augment images by following a certain distribution and randomly moving the fiducial points to q within the radius R.

To generate an augmented image, we apply similarity deformation based on moving least squares [35] on the input image. Given a point u in the image, the transformation for u is

$$T(u) = (u - p_*)\,\mathbf{M} + q_*, \tag{1}$$

where $\mathbf{M} \in \mathbb{R}^{2\times 2}$ is a linear transformation matrix that is constrained to have the property $M^{\mathrm{T}}M = \lambda^2 I$ for some scalar λ. Here p_* and q_* are the weighted centroids of initialized fiducial points p and moved fiducial points q, respectively:

$$p_* = \frac{\sum_{i=1}^{2(N+1)} w_i p_i}{\sum_{i=1}^{2(N+1)} w_i}, q_* = \frac{\sum_{i=1}^{2(N+1)} w_i q_i}{\sum_{i=1}^{2(N+1)} w_i}. \tag{2}$$

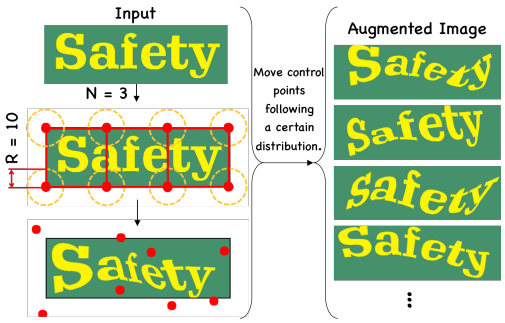

Figure 3. Text augmentation. The image is divided into three patches ($N = 3$) and the moving radius is limited to ten ($R = 10$). The red points denote control points.

The weight w_i for point u has the form

$$w_i = \frac{1}{|p_i - u|^{2\alpha}}, u \neq p_i. \tag{3}$$

Note that as u approaches p_i, the weight w_i increases. This means that u mostly depends on the movement of the nearest fiducial point. The w_i is bounded. If $u = p_i$, then $T(u) = u$. Here we set $\alpha = 1$.

The best transformation $T(u)$ is obtained by minimizing

$$\sum_{i=1}^{2(N+1)} w_i \left| T_u\left(p_i\right) - q_i \right|^2, \tag{4}$$

to yield the unique minimizer [35].

Discussion Though Thin Plate Spline Transformation (TPS) [6] has achieved success in shape rectification [38] and feature-level adversarial learning [5], it is reported that TPS appears non-uniform scaling and shearing, which is undesirable in many applications [35]. One possible reason why previous work used TPS may be all the operators in TPS are differentiable and can be found in most mainstream deep learning libraries. As the learning of our augmentation is free of backward calculation of recognition loss, and our goal is to setup a general augmentation, we choose similarity deformation based on moving least squares as our transformation strategy. Besides, we also compare similarity transformation with rigid transformation [35], which is regarded as the most realistic transformation for general object. As illustrated in Figure 4, the rigid transformation retains relative shape (realistic for general object), but the similarity transformation is more suitable for text image augmentation, because it provides more flexible deformation for every character. Further analysis is given in Section 4.4 and Table 2.

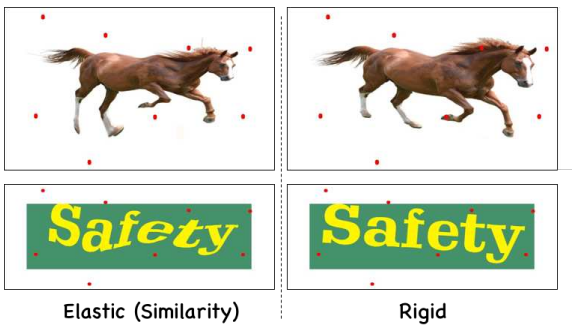

Figure 4. Comparison of the elastic (similarity) and rigid transformation. The movements of the fiducial points on all the images are the same. The rigid transformation retains relative shape (realistic for general object), but text image augmentation requires more flexible deformation for every character. Therefore, the elastic (similarity) transformation is more suitable for text image augmentation.

3.3. Learnable Agent

Different from the previous smart augmentation method [9] that used reinforcement learning to search for best policies, we solve the learning problem in a faster and more efficient fashion. Inspired by heuristic algorithms, we find solutions among all possible ones. As the training procedure is dynamic, approximate solutions are sufficient and exact solutions are computationally expensive. For every step in the training procedure, we generate a variation of the predicted moving state. It serves as a candidate of learning target. If the random moving state increases recognition difficulty, then the agent learns from the moving state. In contrast, we reverse the learning target if the moving state decreases recognition difficulty.

We formulate the problem of finding harder distorted sample as a movement learning problem. As illustrated in Figure 3, given an image, we randomly move the fiducial points to warp the image. The moving operation $(\Delta x, \Delta y)$ for every fiducial point is associated with two factors: 1) the direction of movement, namely, the signs of $(\Delta x, \Delta y)$; 2) the distance of movement, namely, $(|\Delta x|, |\Delta y|)$. In our practice, the learning of distance fails to converge. It is hard for the agent network to precisely learn the distance of the movement. Another interesting observation is that the failed agent network always predicts maximum moving distance to create excessive distorted samples, which reduced the stability of recognizer training. Therefore, we limit the learning space to the direction of movement. Based on the moving direction, the moving distance is randomly generated within the range of radius. It avoids tedious movement predicted by the agent network, because the randomness introduces uncertainties in the augmentation. Moreover, the agent network can be designed as a lightweight architecture. As shown in Table 1, the agent network consists of only six convolutional layers and a fully connected layer. The storage requirement of the agent network is less than 1.5M.

The learning scheme of the agent network is shown in Algorithm 1. First, the learnable agent predicts a moving state distribution aiming to create a harder training sample.

Algorithm 1 Joint Learning Scheme

Input image I_{in} and Ground truth GT;
Patch number N and Moving radius R;
Initialized fiducial points p.

1: Sample moving state as S from predicted distribution:
$$S = \textbf{Agent}(I_{in}).$$
2: Generate random moving state S' (randomly select one point in S and switch to the opposite direction).
3: Both S and S' contain direction for movement.
4: Within the range of R, randomly move p based on S and S' to obtain q and q', respectively.
$$I_{Aug} = \textbf{Augment}(I_{in}, p, q),$$
$$I'_{Aug} = \textbf{Augment}(I_{in}, p, q').$$
5: Recognize I_{Aug} and I'_{Aug}:
$$Reg = \textbf{Recognizer}(I_{Aug}),$$
$$Reg' = \textbf{Recognizer}(I'_{Aug}).$$
6: Update **Recognizer** using I_{Aug}.
7: Measure difficulty by edit distance **ED**$(\cdot)$:
8: **if** **ED**$(Reg, GT) \leq$ **ED**(Reg', GT) **then**
S' increases recognition difficulty.
Update **Agent** network with S' by minimizing:
$$Loss = -\sum_{i=1}^{2(N+1)} \log\left(P(S'_i|I_{in})\right) \quad (5)$$
9: **else**
Update **Agent** network with reversed direction $-S'$ by minimizing:
$$Loss = -\sum_{i=1}^{2(N+1)} \log\left(P(-S'_i|I_{in})\right) \quad (6)$$

A random moving state is also fed to the augmentation module. Then the augmentation module generates augmented samples based on the two moving state, respectively. After that, the recognition network takes the augmented samples as input and predicts text strings. The difficulty of the pair of samples is measured by the edit distance between the ground truth and predicted text strings. Finally, the agent takes the moving state that increase difficulty as guidance and updates itself. The unified framework is end-to-end trainable.

4. Experiments

In this section, we conduct extensive experiments on various benchmarks, including regular and irregular scene text, and handwritten text. We first conduct ablation studies to analyze the impact of the size of training data, the number of divided patches N and the moving radius R on performance. Our method is also compared to existing affine and rigid transformations. Then we integrate state-of-the-art recognition models with our method to show the effectiveness of our learnable data augmentation. Finally, we combine our method with the feature-level adversarial learning method [5] to further boost the recognition performance, which suggests that our method is flexible and can be applied in other augmentation systems.

4.1. Scene Text Datasets

The widely used synthetic datasets [17] and [13] contain 9-million and 8-million synthetic words respectively. We randomly sample 10k, 100k and 1 million images (refered to as **Syn-10k**, **Syn-100k** and **Syn-1m respectively**) for ablation studies.

Real-50k is collected by Li et al. [22] from all the public real datasets, containing approximately 50k samples.

IIIT 5K-Words [30] (**IIIT5K**) contains 3000 cropped word images for testing.

Street View Text [41] (**SVT**) consists of 647 word images for testing. Many images are severely corrupted by noise and blur.

ICDAR 2003 [27] (**IC03**) contains 867 cropped images after discarding images that contained non-alphanumeric characters or had fewer than three characters [41].

ICDAR 2013 [20] (**IC13**) inherits most of its samples from IC03. It contains 1015 cropped images.

Street View Text Perspective [33] (**SVT-P**) contains 645 cropped images for testing. Most of them are perspective distorted.

CUTE80 [34] (**CT80**) contains 80 high-resolution images taken in natural scenes. It was specifically collected to evaluate the performance of curved text recognition. It contains 288 cropped natural images.

ICDAR 2015 [19] (**IC15**) is obtained by cropping the words using the ground truth word bounding boxes and includes more than 200 irregular text images.

Table 1. Architecture of the agent network. "AP" denotes 2×2 average pooling. "BN" represents batch normalization. The kernel size, stride and padding size of all the convolutional layers are 3, 1 and 1, respectively. The output size means 2(N + 1) points, two coordinates and two moving directions.

Type	Size
Input	$1 \times 32 \times 100$
Conv-16, ReLU, AP	$16 \times 16 \times 50$
Conv-64, ReLU, AP	$64 \times 8 \times 25$
Conv-128, BN, ReLU	$128 \times 8 \times 25$
Conv-128, ReLU, AP	$128 \times 4 \times 12$
Conv-64, BN, ReLU	$64 \times 4 \times 12$
Conv-16, BN, ReLU, AP	$16 \times 2 \times 6$
FC-8(N+1)	$8(N+1)$
Reshape	$2(N+1) \times 2 \times 2$

4.2. Handwritten Text Datasets

IAM [29] contains more than 13,000 lines and 115,000 words written by 657 different writers.

RIMES [2] contains more than 60,000 words written in French by over 1000 authors.

4.3. Implementation Details

Network The architecture of the agent network is detailed in Table 1, which is a lightweight network (less than 1.5M) consisting of six convolutional layers and a fully connected layer. The output size means $2(N+1)$ points, two coordinates and two moving directions. As we use the edit distance as the metric of difficulty, the framework is independent of various recognition losses. For instance, Shi et al. [36] adopted CTC loss [12] for convolutional recurrent neural network and the attentional decoders [28, 38] are guided by the cross-entropy loss. Therefore, our framework is friendly to different recognizers. We show the flexibility of our method in the following experiments.

Optimization In the ablation study, we use ADADELTA [45] with default learning rate as the optimizer. The batch size is set to 64. All the images are resized to $(32, 100)$. When our method is integrated with recent state-of-the-art recognizers, the experiment settings, including optimizer, learning rate, image size, and training and testing datasets, are the same as those of the recognizers for the sake of fair comparison.

Environment All experiments are conducted on NVIDIA 1080Ti GPUs. The augmentation module takes less than 2ms to generate a $(32, 100)$ image on a 2.0GHz CPU. It is possible to take advantage of multi-threaded acceleration. For every iteration, the end-to-end training with learnable augmentation takes less than 1.5 times of the training time of the single recognizer. If it is trained with random augmentation, there is nearly no extra time consumption.

4.4. Ablation Study

In this section, we perform a series of ablation studies. As the released scene text datasets [13, 17] provide tens of millions of training samples, it is possible to sample small datasets with three orders of scales. Therefore, we conduct ablation studies on scene text datasets. The training datasets are Real-50k, Syn-10k, Syn-100k and Syn-1m. We use ADADELTA [45] with default learning rate as the optimizer. The batch size is set to 64. All the images are resize to $(32, 100)$. In Table 2, we combine all the scene text testing sets as a unified large dataset for evaluation.

As the attentional recognizer is the most cutting-edge method, we choose the network equipped with ResNet and attentional decoder in [38] as the recognizer. The recognizer trained without any augmentation serves as a baseline. Following the widely used evaluation metric [28, 38], the performance is measured by word accuracy in Table 2-4. To ensure that the training is sufficient, we train the models 10 more epochs after they achieve highest accuracy.

Table 2. Ablation studies on the size of training data and transformation with the settings of $N = 3$ and $R = 10$. "Aug." denotes our augmentation method under a randomly initialized distribution for direction sampling.

Method	Real-50k	Syn-10k	Syn-100k	Syn-1m
baseline	54.1	7.7	39.5	60.9
Affine	58.6	16.9	43.9	61.7
Rigid	58.7	17.5	44.9	63.9
Aug.	63.4	20.1	48.6	65.9
Aug.+Agent	**66.5**	**21.7**	**51.2**	**67.4**

Size of Training data As shown in Table 2, the recognizer using our learnable augmentation method outperforms the baseline by a large margin. For instance, the largest margin of 14.0% is on the Syn-10k dataset. This suggests that our proposed method greatly improves the generalization of recognizer in small-data settings. With the increase of the dataset size, the gap reduces. But there is still a significant accuracy increase of 6.5% on the one million training data Syn-1m.

Transformation Affine transformation [18] including rotation, scaling and translation, is compared with our augmentation method in Table 2. The results show that the recognizer using affine augmentation outperforms the baseline but still falls behind the recognizer that uses our augmentation method, because the affine transformation is limited to designed geometric deformations, which are unable to cover the diversity of text appearance. We also conduct an experiment to study the effectiveness of the rigid transformation. As discussed in Section 3.2, although the rigid transformation is realistic for general object [35], the similarity transformation is more suitable for text image augmentation.

Learnable Agent In Table 2, the agent network further boosts the performance by jointly learning data augmentation and recognizer training. In particular, it achieves an

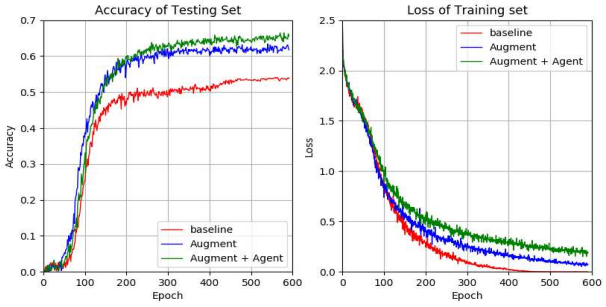

Figure 5. Training loss on Real-50k and testing accuracy on the large evaluation dataset.

Table 3. Ablation studies on the number of patches. R is set to 10.

N	IIIT5K	SVT	IC03	IC13	SVT-P	CT80	IC15
1	23.5	6.6	19.6	22.3	6.0	10.4	10.6
2	**29.8**	10.5	**29.3**	29.3	8.2	14.6	**14.3**
3	29.4	**10.8**	27.2	**29.6**	**9.1**	**16.3**	**14.3**
4	26.5	7.3	22.6	25.6	5.8	11.5	11.0
5	26.1	7.4	22.6	26.9	6.0	13.5	11.2

Table 4. Ablation studies on the moving radius. N is set to 3.

R	IIIT5K	SVT	IC03	IC13	SVT-P	CT80	IC15
0	10.9	2.3	9.0	13.0	1.8	5.2	3.6
2	13.4	2.2	9.8	14.2	2.0	5.2	4.3
5	20.3	4.6	17.0	20.4	4.2	9.0	7.8
10	**29.4**	**10.8**	**27.2**	**29.6**	**9.1**	**16.3**	**14.3**
15	28.8	8.3	26.1	27.8	6.3	13.2	12.2

accuracy increase of 3.1% when the recognizer is trained using Real-50k. The curves of training loss on Real-50k and testing accuracy on the large evaluation dataset are illustrated in Figure 5. An interesting observation is that the loss of the recognizer with learnable agent decreases slower than others, which suggests that the agent network explores the weakness of the recognizer and generates harder samples for training. Thus the recognizer keeps learning and gains robustness. In contrast, the traditional recognizer stops learning when the loss is close to zero.

Patch Number and Moving Radius We study two key parameters N and R respectively. The training dataset is Syn-10k. Table 3 and Table 4 show the experiment results. We find that for regular text, to achieve the best performance, the patch number N can be set to 2 or 3. As for irregular text (SVT-P, CT80 and IC15), it is better to set N to 3, because under this setting, numerous curve text images are generated for training. The recognizer thus gains robustness. We further illustrate the effectiveness of the variance of moving radius R in Table 4. The best setting for a $(32, 100)$ image is $R = 10$. In the following experiments, we use the best setting for N and R for further studies.

4.5. Integration with State-of-the-art Methods

In this section, we integrate our proposed method with state-of-the-art recognizers. The augmented samples for different tasks are shown in Figure 6. We first show the improvement of attention-based recognizer [38] on irregular scene text benchmarks. Then we validate the generalization of our method by using CTC-based recognizer [5] and conducting experiments on handwritten text. Note that our method automatically adapt to general text recognition tasks without any manual modification. Moreover, we show that our method is flexible and can be integrated with other augmentation systems to further boost the performance.

Irregular Scene Text Recognition Irregular shape is one of the challenges for scene text recognition. ASTER proposed by Shi et al. [38] is an attention-based recognizer equipped with rectification network. We study the robustness of the recognizer by augmenting training samples and increasing the diversity of text appearance. The experiment settings, including optimizer, learning rate, image size, and training datasets, are the same as ASTER [38].

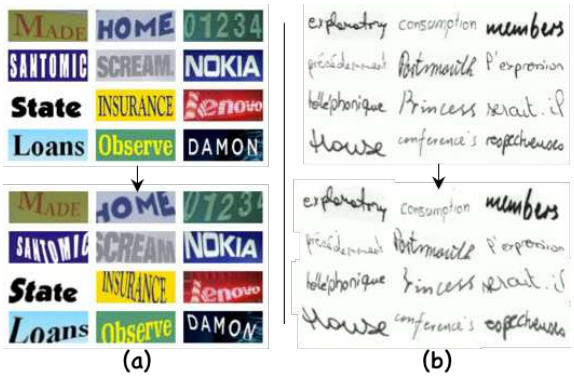

Figure 6. Visualization of augmented samples on (a) scene text and (b) handwritten text.

The performance improved by our method is compared to state-of-the-art methods. Although using real samples [22] and character-level geometric constraints [43] to train the recognizer can significantly improve the performance, we follow the setting of most methods for fair comparison. As Zhan and Lu [46] rectified images for several times and Shi et al. [38] only performed rectification once, we choose the result with one rectification iteration reported in the paper. The performance of scene text recognizers is measured

Table 5. Word accuracy on irregular text. "*" denotes the result is from one rectification iteration for fair comparison.

Method	Irregular Text		
	SVT-P	CT80	IC15
Shi, Bai, and Yao [36]	66.8	54.9	-
Shi et al. [37]	71.8	59.2	-
Liu et al. [25]	73.5	-	-
Yang et al. [44]	75.8	69.3	-
Cheng et al. [7]	71.5	63.9	70.6
Liu, Chen, and Wong [24]	-	-	60.0
Cheng et al. [8]	73.0	76.8	68.2
Bai et al. [4]	-	-	73.9
Liu et al. [26]	73.9	62.5	-
Luo, Jin, and Sun [28]	76.1	77.4	68.8
Liao et al. [23]	-	78.1	-
Shi et al. [38]	78.5	79.5	**76.1**
Zhan and Lu [46]*	77.3	78.8	75.8
baseline (ASTER [38])	77.7	79.9	75.8
+ Ours	**79.2**	**84.4**	**76.1**

by word accuracy.

As shown in Table 5, we first reproduce the same recognizer as ASTER [38], which serves as a baseline. The results of the reimplemented ASTER are comparable to the results in the original paper. Then we integrate our method with the recognizer. A significant accuracy gain occurs on CT80 (4.5%). It is noteworthy that there is still a notable improvement (1.5%) on SVT-P, which contains images with noise, blur and low-resolution. Though abundant synthetic samples may cover a lot of variation of text appearance, our augmentation shows reasonable improvement on irregular text recognition. The result is competitive with recent state-of-the-art methods.

Handwritten Text Recognition As the diversity of handwriting styles is the main challenge of handwritten text recognition [1] and limited training data is difficult to cover all handwriting styles, we evaluate our model on two popular datasets IAM [29] and RIMES [2] to validate the effectiveness of our method. We use Character Error Rate (CER) and Word Error Rate (WER) as metrics for handwritten text recognition. The CER measures the Levenshtein distance normalized by the length of the ground truth. The WER denotes the ratio of the mistakes at the word level, among all words of the ground truth.

We compare our method to state-of-the-art methods in the Table 6 and Table 7. Besides, a comparison with previous augmentation method of Bhunia et al. [5] is conducted. For fair comparisons, our experiment settings are the same with [5].

We apply the same CTC-based recognition network as [5]. The baseline shown in Table 6 and Table 7 is the reproduced result. Further, we reproduce Adversarial Feature Deformation Module (AFDM) [5] in the recognition network. The AFDM is the key module proposed by Bhunia et al. [5] for smart augmentation. The accuracy increases as expected. Note that our reproduced results are better than most of the results (7 of 8) in the original paper, which verifies the effectiveness of our implementations and experiments. We find that our augmentation greatly contributes to the robustness of the recognizer. It improves the performance by a large margin (5.08% unconstrained WER reduction on IAM) and significantly performs better than AFDM. The recognizer trained using our method also outperforms all the state-of-the-art methods.

Table 6. Comparison with previous methods on IAM. AFDM is the key module of [5].

Method	Unconstrained		Lexicon	
	WER	CER	WER	CER
Bosquera et al. [10]	-	-	20.01	11.27
Almazán et al. [1]	-	-	15.50	6.90
Sun et al. [40]	-	-	11.51	-
Sueiras et al. [39]	23.80	8.80	19.70	9.50
Ptucha et al. [32]	-	-	8.22	4.70
Zhang et al. [47]	22.20	8.50	-	-
Bhunia et al. [5]	17.19	8.41	8.87	5.94
baseline	19.12	7.39	10.07	5.41
+ Ours	14.04	5.34	7.52	3.82
+ AFDM [5]	16.40	6.40	8.77	4.67
+ Ours + AFDM [5]	**13.35**	**5.13**	**7.29**	**3.75**

Table 7. Comparison with previous methods on RIMES. AFDM is the key module of [5].

Method	Unconstrained		Lexicon	
	WER	CER	WER	CER
Sueiras et al. [39]	15.90	4.80	13.10	5.70
Ptucha et al. [32]	-	-	5.68	2.46
Bhunia et al. [5]	10.47	6.44	6.31	3.17
baseline	13.83	3.93	4.94	2.02
+ Ours	9.23	2.57	4.41	1.49
+ AFDM [5]	11.81	3.33	4.85	1.92
+ Ours + AFDM [5]	**8.67**	**2.42**	**3.90**	**1.37**

Finally, we use both AFDM and our method for training and further boost the performance of the recognizer by a notable accuracy increase. This suggests that our method is a meta framework, which can be applied in other augmentation systems.

5. Conclusion

In this paper, we propose a learnable augmentation method for the training of text recognizer. Our method may be the first geometric augmentation method specifically designed for sequence-like characters. Furthermore, our method bridges the gap between the data augmentation and network optimization by joint learning. The proposed method is simple yet effective. It is able to automatically adapt to general text recognition tasks without any manual modification. Extensive experiments show that our method boosts the performance of the recognizers for both scene text and handwritten text. Moreover, our method is a meta framework that potentially can be incorporated into other augmentation systems. In future, we will extend our method for more general applications in multiple object detection and recognition.

Acknowledgement

This research is supported in part by NSFC (Grant No.: 61936003), the National Key Research and Development Program of China (No. 2016YFB1001405), and GD-NSF (no.2017A030312006).

References

[1] Jon Almazán, Albert Gordo, Alicia Fornés, and Ernest Valveny. Word spotting and recognition with embedded attributes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(12):2552–2566, 2014. 2, 8

[2] Emmanuel Augustin, Matthieu Carré, Emmanuèle Grosicki, J-M Brodin, Edouard Geoffrois, and Françoise Prêteux. Rimes evaluation campaign for handwritten mail processing. In *IWFHR*, pages 231–235, 2006. 6, 8

[3] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2015. 1

[4] Fan Bai, Zhanzhan Cheng, Yi Niu, Shiliang Pu, and Shuigeng Zhou. Edit probability for scene text recognition. In *CVPR*, pages 1508–1516, 2018. 7

[5] Ayan Kumar Bhunia, Abhirup Das, Ankan Kumar Bhunia, Perla Sai Raj Kishore, and Partha Pratim Roy. Handwriting recognition in low-resource scripts using adversarial learning. In *CVPR*, pages 4767–4776, 2019. 1, 2, 4, 5, 7, 8

[6] Fred L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(6):567–585, 1989. 4

[7] Zhanzhan Cheng, Fan Bai, Yunlu Xu, Gang Zheng, Shiliang Pu, and Shuigeng Zhou. Focusing attention: Towards accurate text recognition in natural images. In *ICCV*, pages 5086–5094, 2017. 7

[8] Zhanzhan Cheng, Yangliu Xu, Fan Bai, Yi Niu, Shiliang Pu, and Shuigeng Zhou. AON: Towards arbitrarily-oriented text recognition. In *CVPR*, pages 5571–5579, 2018. 7

[9] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. AutoAugment: Learning augmentation strategies from data. In *CVPR*, pages 113–123, June 2019. 1, 2, 3, 4

[10] Salvador Espana-Boquera, Maria Jose Castro-Bleda, Jorge Gorbe-Moya, and Francisco Zamora-Martinez. Improving offline handwritten text recognition with hybrid hmm/ann models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(4):767–779, 2010. 2, 8

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, pages 2672–2680, 2014. 1

[12] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *ICML*, pages 369–376, 2006. 6

[13] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. In *CVPR*, pages 2315–2324, 2016. 1, 3, 5, 6

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1

[15] Pan He, Weilin Huang, Yu Qiao, Chen Change Loy, and Xiaoou Tang. Reading scene text in deep convolutional sequences. In *AAAI*, pages 3501–3508, 2016. 2

[16] Daniel Ho, Eric Liang, Xi Chen, Ion Stoica, and Pieter Abbeel. Population based augmentation: Efficient learning of augmentation policy schedules. In *ICML*, pages 2731–2741, 09–15 Jun 2019. 2, 3

[17] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Reading Text in the Wild with Convolutional Neural Networks. *Int. J. Comp. Vis.*, 116(1):1–20, May 2015. 1, 3, 5, 6

[18] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *NeurIPS*, pages 2017–2025, 2015. 6

[19] Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, et al. ICDAR 2015 competition on robust reading. In *ICDAR*, pages 1156–1160, 2015. 5

[20] Dimosthenis Karatzas, Faisal Shafait, Seiichi Uchida, Masakazu Iwamura, Lluis Gomez i Bigorda, Sergi Robles Mestre, Joan Mas, David Fernandez Mota, Jon Almazan Almazan, and Lluis Pere De Las Heras. ICDAR 2013 robust reading competition. In *ICDAR*, pages 1484–1493, 2013. 5

[21] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *NeurIPS*, volume 2, pages 1097–1105, 2012. 1, 2

[22] Hui Li, Peng Wang, Chunhua Shen, and Guyu Zhang. Show, attend and read: A simple and strong baseline for irregular text recognition. In *AAAI*, volume 33, pages 8610–8617, 2019. 2, 3, 5, 7

[23] Minghui Liao, Jian Zhang, Zhaoyi Wan, Fengming Xie, Jiajun Liang, Pengyuan Lyu, Cong Yao, and Xiang Bai. Scene text recognition from two-dimensional perspective. In *AAAI*, volume 33, pages 8714–8721, 2019. 1, 2, 7

[24] Wei Liu, Chaofeng Chen, and Kwan-Yee K Wong. Charnet: A character-aware neural network for distorted scene text recognition. In *AAAI*, 2018. 7

[25] Wei Liu, Chaofeng Chen, Kwan-Yee K Wong, Zhizhong Su, and Junyu Han. STAR-Net: A spatial attention residue network for scene text recognition. In *BMVC*, pages 7–7, 2016. 7

[26] Yang Liu, Zhaowen Wang, Hailin Jin, and Ian Wassell. Synthetically supervised feature learning for scene text recognition. In *ECCV*, pages 435–451, 2018. 7

[27] Simon M Lucas, Alex Panaretos, Luis Sosa, Anthony Tang, Shirley Wong, and Robert Young. ICDAR 2003 robust reading competitions. In *ICDAR*, pages 682–687, 2003. 5

[28] Canjie Luo, Lianwen Jin, and Zenghui Sun. MORAN: A multi-object rectified attention network for scene text recognition. *Patt. Recogn.*, 90:109–118, 2019. 1, 2, 6, 7

[29] U-V Marti and Horst Bunke. The iam-database: an english sentence database for offline handwriting recognition. *Int. J. Doc. Anal. Recogn.*, 5(1):39–46, 2002. 6, 8

[30] Anand Mishra, Karteek Alahari, and CV Jawahar. Scene text recognition using higher order language priors. In *BMVC*, pages 1–11, 2012. 5

[31] Xi Peng, Zhiqiang Tang, Fei Yang, Rogerio S. Feris, and Dimitris Metaxas. Jointly optimize data augmentation and network training: Adversarial data augmentation in human pose estimation. In *CVPR*, pages 2226–2234, June 2018. 2, 3

[32] Raymond Ptucha, Felipe Petroski Such, Suhas Pillai, Frank Brockler, Vatsala Singh, and Paul Hutkowski. Intelligent character recognition using fully convolutional neural networks. *Patt. Recogn.*, 88:604–613, 2019. 8

[33] Trung Quy Phan, Palaiahnakote Shivakumara, Shangxuan Tian, and Chew Lim Tan. Recognizing text with perspective distortion in natural scenes. In *ICCV*, pages 569–576, 2013. 5

[34] Anhar Risnumawan, Palaiahankote Shivakumara, Chee Seng Chan, and Chew Lim Tan. A robust arbitrary text detection system for natural scene images. *Expert Systems with Applications*, 41(18):8027–8048, 2014. 5

[35] Scott Schaefer, Travis McPhail, and Joe Warren. Image deformation using Moving Least Squares. In *ACM Transactions on Graphics*, pages 533–540, July 2006. 2, 3, 4, 6

[36] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(11):2298–2304, 2017. 2, 3, 6, 7

[37] Baoguang Shi, Xinggang Wang, Pengyuan Lyu, Cong Yao, and Xiang Bai. Robust scene text recognition with automatic rectification. In *CVPR*, pages 4168–4176, 2016. 7

[38] Baoguang Shi, Mingkun Yang, Xinggang Wang, Pengyuan Lyu, Cong Yao, and Xiang Bai. ASTER: An attentional scene text recognizer with flexible rectification. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(9):2035–2048, 2019. 1, 2, 3, 4, 6, 7, 8

[39] Jorge Sueiras, Victoria Ruiz, Angel Sanchez, and Jose F Velez. Offline continuous handwriting recognition using sequence to sequence neural networks. *Neurocomputing*, 289:119–128, 2018. 2, 8

[40] Zenghui Sun, Lianwen Jin, Zecheng Xie, Ziyong Feng, and Shuye Zhang. Convolutional multi-directional recurrent network for offline handwritten text recognition. In *ICFHR*, pages 240–245. IEEE, 2016. 2, 8

[41] Kai Wang, Boris Babenko, and Serge Belongie. End-to-end scene text recognition. In *ICCV*, pages 1457–1464, 2011. 2, 5

[42] Tao Wang, David J Wu, Adam Coates, and Andrew Y Ng. End-to-end text recognition with convolutional neural networks. In *ICPR*, pages 3304–3308, 2012. 2

[43] Mingkun Yang, Yushuo Guan, Minghui Liao, Xin He, Kaigui Bian, Song Bai, Cong Yao, and Xiang Bai. Symmetry-constrained rectification network for scene text recognition. In *ICCV*, pages 9147–9156, 2019. 2, 7

[44] Xiao Yang, Dafang He, Zihan Zhou, Daniel Kifer, and C Lee Giles. Learning to read irregular text with attention mechanisms. In *IJCAI*, pages 3280–3286, 2017. 7

[45] Matthew D Zeiler. ADADELTA: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012. 6

[46] Fangneng Zhan and Shijian Lu. ESIR: End-to-end Scene Text Recognition via Iterative Image Rectification. In *CVPR*, pages 2059–2068, 2019. 2, 7

[47] Yaping Zhang, Shuai Nie, Wenju Liu, Xing Xu, Dongxiang Zhang, and Heng Tao Shen. Sequence-to-sequence domain adaptation network for robust text image recognition. In *CVPR*, pages 2740–2749, 2019. 1, 2, 8