ESIR: End-to-end Scene Text Recognition via Iterative Image Rectification

Fangneng Zhan
Nanyang Technological University
50 Nanyang Avenue, Singapore 639798
fnzhan@ntu.edu.sg

Shijian Lu
Nanyang Technological University
50 Nanyang Avenue, Singapore 639798
shijian.lu@ntu.edu.sg

Abstract

Automated recognition of texts in scenes has been a research challenge for years, largely due to the arbitrary variation of text appearances in perspective distortion, text line curvature, text styles and different types of imaging artifacts. The recent deep networks are capable of learning robust representations with respect to imaging artifacts and text style changes, but still face various problems while dealing with scene texts with perspective and curvature distortions. This paper presents an end-to-end trainable scene text recognition system (ESIR) that iteratively removes perspective distortion and text line curvature as driven by better scene text recognition performance. An innovative rectification network is developed which employs a novel line-fitting transformation to estimate the pose of text lines in scenes. In addition, an iterative rectification pipeline is developed where scene text distortions are corrected iteratively towards a fronto-parallel view. The ESIR is also robust to parameter initialization and the training needs only scene text images and word-level annotations as required by most scene text recognition systems. Extensive experiments over a number of public datasets show that the proposed ESIR is capable of rectifying scene text distortions accurately, achieving superior recognition performance for both normal scene text images and those suffering from perspective and curvature distortions.

1. Introduction

Texts in scenes contain high level semantic information that is very useful in many practical applications such as indoor and outdoor navigation, content-based image retrieval, etc. Accurate and robust recognition of scene texts by machines has been a research challenge for years, largely due to a huge amount of variations in text appearance, the complicated image background, imaging artifacts, etc. The advance of deep learning research and its successes in many computer vision tasks have pushed the boundary of scene text recognition greatly in recent years

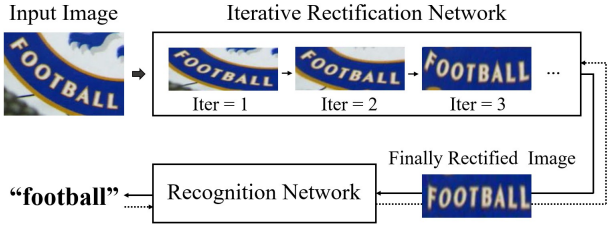

Figure 1. The proposed system consists of an Iterative Rectification Network that corrects scene text distortions iteratively and a Recognition Network that recognizes the finally rectified scene texts (the images within the Iterative Rectification Network illustrate the iterative scene text rectification process). It is end-to-end trainable as driven by better scene text recognition performance, where the dotted lines show the back propagation of gradients.

[29, 16, 35, 36, 40, 2, 23, 24, 25, 48, 45, 49, 50]. On the other hand, the deep learning based approach is still facing various problems while dealing with a large amount of scene texts that suffer from arbitrary perspective distortions and text line curvature.

We design an end-to-end trainable scene text recognition network via iterative rectification (ESIR). The ESIR employs an innovative rectification network that corrects perspective and curvature distortions of scene texts iteratively as illustrated in Fig. 1. The finally rectified scene text image is fed to a recognition network for recognition. The training of the iterative rectification network is driven by better scene text recognition as back-propagated from the recognition network, requiring no other annotations beyond the scene texts as used in most scene text recognition systems.

The proposed ESIR addresses two typical constraints in the scene text recognition problem. The first is *robust* distortion rectification for optimal scene text recognition. To address this challenge, we design a novel line-fitting transformation that is powerful and capable of modeling and correcting various scene text distortions reliably. The line-fitting transformation models the middle line of scene texts by using a polynomial which is able to estimate the pose of either straight or curved text lines flexibly as illustrated in Fig. 2. In addition, it employs line segments which are

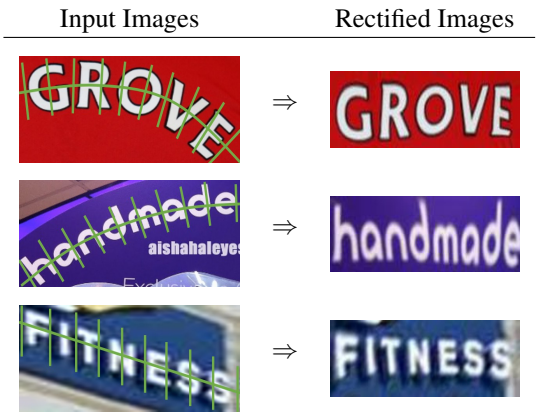

Figure 2. A novel line-fitting transformation is designed which employs a polynomial to model the middle line of scene texts in horizontal direction, and a set of line segments to estimate the orientation and the boundary of text lines in vertical direction. It helps to model and correct various scene text distortions accurately.

capable of estimating the orientation and the boundary of text lines in vertical direction reliably. The proposed rectification network is thus capable of correcting not only perspective distortions in straight text lines as in spatial transfer networks [18] and bag-of-keypoints recognizer [33], but also various curvatures in curved text lines in scenes.

The second is *accurate* rectification of perspective and curvature distortions of texts in scenes. To address this challenge, we develop an iterative rectification pipeline that employs multiple feed-forward rectification modules to estimate and correct scene text distortions iteratively. As illustrated in Fig. 2, each iteration takes the image rectified in the last iteration for further distortion estimation as driven by higher scene text recognition accuracy. The iterative rectification is thus capable of producing more accurate distortion correction compared with the state-of-the-art that just performs a single distortion estimation and correction [37, 38]. In addition, the iteratively rectified images lead to superior scene text recognition accuracy especially for datasets that contain a large amount of curved and/or perspectively distorted texts, to be described in **Experiments**.

The contributions of this work are threefold. First, it proposes a novel line-fitting transformation that is flexible and robust for scene text distortion modeling and correction. Second, it designs an iterative rectification framework that clearly improves the scene text rectification and recognition performance with no extra annotations. Third, it develops an end-to-end trainable system that is robust to parameter initialization and achieves superior scene text recognition performance across a number of public datasets.

2. Related Work

2.1. Scene Text Recognition

Existing scene text recognition work can be broadly grouped into two categories. One category adopts a bottom-up approach that first detects and recognizes individual characters. The other category takes a top-down approach that recognizes words or text lines directly without explicit detection and recognition of individual characters.

Most traditional scene text recognition systems follow a bottom-up approach that first detects and recognizes individual characters by using certain hand-crafted features and then links up the recognized characters into words or text lines using dynamic programming and language models. Different scene character detection and recognition methods have been reported by using sliding window [43, 42], connected components [31], extremal regions [32], Hough voting [47], co-occurrence histograms [41], etc., but most of them are constrained by the representation capacity of the hand-crafted features. With the advances of deep learning in recent years, various CNN architectures and frameworks have been designed for scene character recognition. For example, [4] adopts a fully connected network to recognize characters, [44] uses CNNs for feature extraction, and [16] uses CNNs for unconstrained character recognition. On the other hand, these deep network based methods require localization of individual characters which is resource-hungry and also prone to errors due to complex image background and heavy touching between adjacent characters.

To address the character localization issues, various top-down methods have been proposed which recognize an entire word or text line directly without detecting and recognizing individual characters. One approach is to treat a word as a unique object class and convert the scene text recognition into an image classification problem [17]. In addition, recurrent neural networks (RNNs) have been widely explored which encode a word or text line as a feature sequence and perform recognition without character segmentation. For example, [39, 40] extract histogram of oriented gradient features across a text sequence and use RNNs to convert them into a feature sequence. [13], [6] and [36] propose end-to-end systems that use RNNs for visual feature representation and CTC for sequence prediction. In recent years, visual attention has been incorporated which improves recognition by detecting more discriminative and informative image regions. For example, [21] learns broader contextual information and uses an attention based decoder for sequence generation. [7] proposes a focus mechanism to eliminate attention drift to improve the scene text recognition performance. [14] designs a novel character attention mechanism for end-to-end scene text spotting.

2.2. Recognition of Distorted Scene Texts

The state-of-the-art combining RNNs and attention has achieved great success while dealing with horizontal or slightly distorted texts in scenes. On the other hand, most existing methods still face various problems while dealing with many scene texts that suffer from either perspective distortions or text line curvatures or both.

Prior works dealing with perspective distortions and text line curvatures are limited but this problem has attracted increasing attention in recent years. The very early works [26, 9, 27] correct perspective distortions in document texts as captured by digital cameras for better recognition. [33] works with scene texts by using bag of key-points that are tolerant to perspective distortions. These early systems achieve limited successes as they use hand-crafted features and also require character-level information. The recent works [37, 38] also take an image rectification approach but explore spatial transformer networks for scene text distortion correction. Similarly, [3, 22] integrate the rectification and recognition into the same network. These recent systems exploit deep convolutional networks for rectification and RNNs for recognition, which require little manually crafted features or extra annotations and have shown very promising recognition performance.

Our proposed technique adopts a rectification approach for robust and accurate recognition of scene texts with perspective and curvature distortions. Different from existing rectification based works [37, 38, 3, 22], it corrects distortions in an iterative manner which helps to improve the rectification and recognition greatly. In addition, we propose a novel line-fitting transformation that is robust and flexible in scene text distortion estimation and correction.

Note some attempt has been reported in recent years which handles scene text perspectives and curvature distortions by managing deep network features. For example, [46] presents an auxiliary dense detector to encourage visual representation learning. [8] describes an arbitrary orientation network that extracts scene text features in four directions to deal with scene text distortions.

3. The Proposed Method

This section presents the proposed scene text recognition technique including iterative scene text rectification network, sequence recognition network and detailed description of network training.

3.1. Iterative Rectification Network

The proposed iterative rectification network employs a novel line-fitting transformation and an iterative rectification pipeline for optimal estimation and correction of perspective and curvature distortions of texts in scenes.

3.1.1 Line-Fitting Transformation

A novel line-fitting transformation is designed to model the pose of scene texts and correct perspective and curvature distortions for better scene text recognition. As illustrated in Fig. 2, the fitting lines consist of a polynomial that is employed to model the middle line of text lines in horizontal direction, and L line segments that are employed to estimate the orientation and the boundary of text lines in vertical direction. Since most text lines in scenes are either along a straight or smoothly curved line, a polynomial of a certain order is sufficient for the estimation of text line poses in the horizontal direction. In our implemented system, a polynomial of order 4 and a set of 20 line segments are employed for scene text pose estimation.

By setting the image center as the origin and normalizing the x-y coordinate of each pixel within scene text images, the middle line of text lines in scenes can be modeled by a polynomial of order K as follows:

$$y = a_K * x^K + a_{K-1} * x^{K-1} + \cdots + a_1 * x + a_0$$

The L line segments can be modeled by:

$$y = b_{1,l} * x + b_{0,l} \mid r_l, \quad l = 1, 2, \cdots, L$$

where r_l denotes the length of line segments on the two sides of the middle line of text lines in scenes which can be approximated as the same. We therefore have $3L$ parameters for estimating the L line segments. By including the middle line polynomial, the number of parameters to be estimated becomes $3L + K + 1$.

The proposed rectification network iteratively regresses to estimate the fitting-line parameters by employing a localization network together with image convolutions as illustrated in Fig. 3. Table 1 gives detailed structures of the localization network. It should be noted that the training of the localization network does not require any extra annotation of fitting lines but is completely driven by the gradients that are back-propagated from the recognition network. The underlying principle is that higher recognition performance is usually achieved when scene text distortions are better estimated and corrected.

Once the fitting line parameters are estimated, the coordinates of 2 endpoints of each of the L line segments $\{t_j | j = 1, \cdots, 2L\}$ can be determined. Scene text distortions can then be corrected by a thin plate spline transformation [5] that can be determined based on the estimated line segment endpoints and another $2L$ base points $\{t'_j | j = 1, \cdots, 2L\}$ which define the appearance of scene texts within the rectified scene text image:

$$t'_j = \begin{cases} [-0.5 + \frac{j-1}{L-1}, 0.5]^T, & 1 \le j \le L \\ [-0.5 + \frac{l-(L+1)}{L-1}, -0.5]^T, & L+1 \le j \le 2L \end{cases}$$

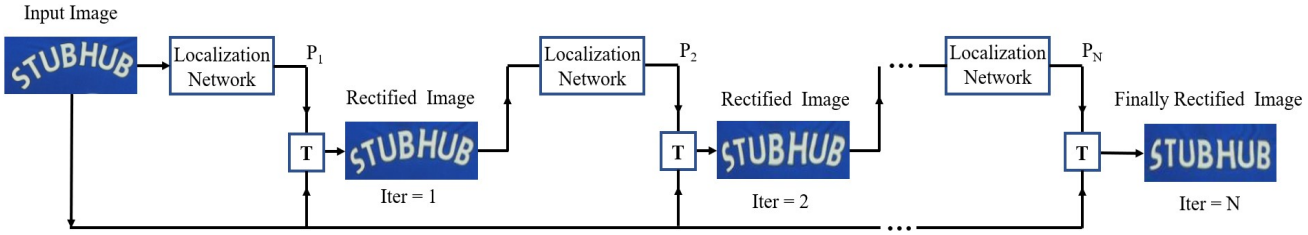

Figure 3. The iterative rectification process: T denotes a thin plate spline transformation, P_1, P_2, ... denote transformation parameters that are predicted by the localization network, Iter denotes the number of rectification iterations and N is the predefined iteration number.

Table 1. The structure of the Rectification Network in Fig. 1

Layers	Out Size	Configurations
Block1	16×50	3×3 $conv, 32, 2 \times 2$
Block2	8×25	3×3 $conv, 64, 2 \times 2$
Block3	4×13	3×3 $conv, 128, 2 \times 2$
FC1	512	-
FC2	$3L$+K+1	-

With $P = [t_1, t_2, \cdots t_{2L}]^T$ and $P' = [t'_1, t'_2, \cdots t'_{2L}]^T$, the transformation parameters can be determined by:

$$C = \begin{bmatrix} S & 1_{2L} & P \\ 1_{2L}^T & 0 & 0 \\ P^T & 0 & 0 \end{bmatrix}^{-1} \cdot \begin{bmatrix} P' \\ 0 \\ 0 \end{bmatrix}$$

where S=[U(t-t_1), U(t-t_2), $\cdots$, U(t-t_{2L})]^T and $U(r) = r^2 log r^2$. For every pixel t' within the rectified scene text image, the corresponding pixel t within the distorted scene text image can thus be determined as follows:

$$t = C \cdot t'$$

With the estimated pixels $\{t'_j | j = 1, \cdots, 2L\}$, a grid $G = \{t_j | j = 1, \cdots, 2L\}$ can be generated within the distorted scene text image for rectification. A sampler is implemented to produce the rectified scene text image by using the determined grid, where the value of the pixel t' is bilinearly interpolated from the pixels near t within the distorted scene text image. The sampler can back propagate the image gradients as it is fully differentiable.

3.1.2 Iterative Rectification

We develop an iterative rectification pipeline for optimal scene text rectification and recognition. At the first iteration, the rectification network takes the original scene text image as input and rectifies it to certain degrees by using the estimated transformation parameters as described in the last subsection. After that, the rectified scene text image is fed to the same rectification network for further parameter estimation and image rectification. This process repeats until a predefined number of rectification iterations is reached. The finally rectified image is then fed to the sequence recognition network for scene text recognition.

The iterative rectification improves the scene text image rectification performance greatly as to be described in **Experiments**. On the other hand, it often encounters a critical 'boundary effect' problem if the iterative rectification is performed directly without control. In particular, each rectification iteration discards image pixels lying outside of the image sampling region, which accumulates and could lead to the discarding of certain text pixels during the iterative image rectification process. In addition, direct image rectification iteratively often degrades the image clarity severely because of the multiple rounds of bilinear interpolations.

We deal with the 'boundary effects' and losing of image clarity by designing a novel network structure as illustrated in Fig. 3. In particular, the rectification network consists of localization networks for estimating rectification parameters, and a thin plate spline transformation T that employs the estimated rectification parameters to generate rectified scene text images. During the iterative image rectification process, the intermediately rectified scene text image is used for parameter estimation only, and the original instead of intermediately rectified scene text image is fed to the transformation module T for rectification consistently. With this new design, 'boundary effect' accumulation can be avoided and image clarity will not be degraded, both help to improve the scene text recognition performance greatly as to be presented in **Experiments**.

3.2. Recognition Network

The recognition network employs a sequence-to-sequence model with an attention mechanism. It consists of an encoder and a decoder. In the encoder, the input is a rectified scene text image that is re-sized to 32×100 pixels. A 53-layer residual network [12] is used to extract features, where each residual unit consists of a 1×1 convolution and a 3×3 convolution operations. A 2×2 stride convolution is implemented to down-sample feature maps in the first two

residual blocks. The convolution stride is then changed to 2×1 in all following residual blocks and this helps to reserve more information along the horizontal direction and is also very useful for distinguishing neighboring characters. The residual network is followed by two layers of Bidirectional long short-term memory (BiLSTM) each of which has 256 hidden units. The decoder adopts the LuongAttention mechanism [28] which consists of 2-layer attentional LSTMs with 256 hidden units and 256 attention units. During the inference stage, beam search is employed to decode components that maintain k candidates with the highest accumulative scores.

3.3. Network Training

The training of image rectification networks is never a simple task. The major issue is that image rectification networks are sensitive to parameter initialization as frequently encountered and shared in prior studies [37, 38]. In particular, random parameter initialization often leads to network convergence problems because it is liable to produce highly distorted scene text images that ruin the training of the recognition network and further the rectification network (note that the training of the rectification network is driven by the scene text recognition performance).

We address the network initialization problem by avoiding direction prediction of P as defined in **Line-Fitting Transformation**. Instead, we define an auxiliary P_0 that equals P' at the beginning and make $P = P_0 + \triangle P$, where $\triangle P$ is predicted by the rectification network iteratively. By assigning a small value to $\triangle P$, the initial P will have a similar value as P'. This approach avoids generating highly distorted scene text image at the beginning stage and improves the network convergence greatly. Additionally, the gradual learning of P via iterative estimation of $\triangle P$ (instead of direct prediction of P) makes the rectification network training smooth and stable.

4. Experiments

4.1. Datasets and Metrics

This section describes a list of datasets and evaluation metrics that are used in the experiments.

4.1.1 Datasets

All ESIR models to be evaluated are trained by using the Synth90K and SynthText, and there is no fine-tuning by using any third dataset. The ESIR models are evaluated over 6 scene text datasets including 3 normal datasets ICDAR2013, IIIT5K and SVT where most scene texts are almost horizontal, and 3 distorted datasets ICDAR2015, SVTP and CUTE80 where a large amount of scene texts suffer from perspective and curvature distortions. The 6 datasets are publicly accessible which have been widely used for evaluations in scene text recognition research.

Synth90K [15] contains 9 million synthetic text images with a lexicon of 90K, and it has been widely used for training scene text recognition models. It has no separation of training and test data and all images are used for training.

SynthText [11] is the synthetic image dataset that was created for scene text detection research. It has been widely used for scene text recognition research as well by cropping text image patches using the provided annotation boxes. State-of-the-art methods crop different amounts for evaluations, e.g. [8] crops 4 million, Shi [38] crops over 7 million, etc. We crop 4 million text image patches from this dataset which are at lower end for fair benchmarking.

ICDAR2013 [20] is used in the Robust Reading Competition in the International Conference on Document Analysis and Recognition (ICDAR) 2013. It contains 848 word images for model training and 1095 for testing.

ICDAR2015 [19] was used in the Robust Reading Competition under ICDAR 2015. It contains incidental scene text images that are captured without preparation before capturing. 2077 text image patches are cropped from this dataset, where a large amount of cropped scene texts suffer from perspective and curvature distortions.

IIIT5K [30] consists of 2000 training images and 3000 test images that are cropped from scene texts and born-digital images. Each word image in this dataset has a 50-word lexicon and a 1000-word lexicon, where each lexicon consists of a ground-truth word and a set of randomly picked words.

SVT [42] is collected from the Google Street View images that were used for scene text detection research. 647 words images are cropped from 249 street view images and words within most cropped word images are almost horizontal. Each word image has a 50-word lexicon.

SVTP [33] consists of 639 word images that are cropped from the SVT images. Most images in this dataset are heavily distorted by perspective distortions which are specifically picked for evaluation of scene text recognition under perspective views. Each word image has a 50-word lexicon as inherited from the SVT dataset.

CUTE [34] consists of 288 word images where most cropped scene texts are curved. All word images are cropped from the CUTE dataset which contains 80 high-resolution scene text images that are originally collected for the scene text detection research. No lexicon is provided for the 288 word images in this dataset.

4.1.2 Metrics

We follow the protocol and evaluation metrics that have been widely used in scene text recognition research [7, 38]. In particular, the recognition covers 68 characters includ-

Table 2. Scene text recognition performance over the datasets ICDAR2013, ICDAR2015, IIIT5K, SVT, SVTP and CUTE, where "50" and "1K" in the second row denote the lexicon size and "None" means no lexicon used. The used network backbone and training data are given in [] at the end of each method, where SK and ST denote the Synth90K and SynthText datasets, respectively.

Methods	ICDAR2013	ICDAR2015	IIIT5K			SVT		SVTP	CUTE
	None	None	50	1k	None	50	None	None	None
Wang [44] [-]	-	-	-	-	-	70.0	-	-	-
Bissacco [4] [-]	87.6	-	-	-	-	-	-	-	-
Yao [47] [-]	-	-	80.2	69.3	-	75.9	-	-	-
Almazan [1] [-]	-		91.2	82.1	-	89.2	-	-	-
Gordo [10] [-]	-		93.3	86.6	-	91.8	-	-	-
Jaderberg [16] [VGG, SK]	81.8	-	95.5	89.6	-	93.2	71.7	-	-
Jaderberg [17] [VGG, SK]	90.8	-	97.1	92.7	-	95.4	80.7	-	-
Shi [37] [VGG, SK]	88.6	-	96.2	93.8	81.9	95.5	81.9	71.8	59.2
Yang [46] [VGG, Private]	-	-	97.8	96.1	-	95.2	-	75.8	69.3
Cheng [7] [ResNet, SK+ST]	**93.3**	70.6	99.3	97.5	87.4	97.1	85.9	71.5	63.9
Cheng [8] [VGG, SK+ST]	-	68.2	99.6	98.1	87.0	96.0	82.8	73.0	76.8
Shi [38] [ResNet, SK+ST]	91.8	76.1	**99.6**	**98.8**	**93.4**	97.4	89.5	78.5	79.5
ESIR [VGG, SK]	87.4	68.4	95.8	92.9	81.3	96.7	84.5	73.8	68.4
ESIR [ResNet, SK]	89.1	70.1	97.8	96.1	82.9	97.1	85.9	75.8	72.1
ESIR [ResNet, SK+ST]	91.3	**76.9**	**99.6**	**98.8**	93.3	**97.4**	**90.2**	**79.6**	**83.3**

ing 10 digits, lower-case letters and 32 ASCII punctuation marks. In evaluation, only digits and letters are counted and the rest is directly discarded. If a lexicon is provided, the lexicon word that has the minimum edit distance with the predicted word is selected. In addition, evaluations are based on the correctly recognized words (CRW) which can be determined based on the ground truth transcription.

4.2. Implementation

The proposed scene text recognition network is implemented by using the Tensorflow framework. The ADADELTA is adopted as optimizer which employs adaptive learning rate and weighted cross-entropy in sequence loss calculation. The network is trained in 1 million iterations with a batch size of 64. In addition, the network training is performed on a workstation with one Intel Core i7-7700K CPU, one NVIDIA GeForce GTX 1080 Ti graphics card with 12GB memory and 32GB RAM.

Three ESIR models are trained for evaluations and benchmarking with state-of-the-art techniques. The first is a baseline model **ESIR [VGG, SK]** as shown in Table 2, which uses VGG as the network backbone and the Synth90 as the training data. The second model as denoted by **ESIR [ResNet, SK]** uses the same training data but ResNet as the network backbone. The third model as denoted by **ESIR [ResNet, SK+ST]** uses ResNet as the network backbone but a combination of the Synth90K and SynthText as training data, largely for benchmarking with state-of-the-art models such as ASTER and AON that also use a combination of the two datasets in training. All three ESIR models are trained under the same parameters setting: rectification iteration number: 5; number of line segments: 20; order of the middle line polynomial: 4.

4.3. Experimental Results

4.3.1 Rectification and Recognition

The proposed ESIR has been evaluated extensively over the 6 public datasets as described in **Dataset** that contain both normal scene text images and scene text images with a variety of perspective and curvature distortions. In addition, it has been benchmarked with a number of state-of-the-art scene text recognition techniques that employ rectification, feature learning techniques, etc. as described in **Related**

Figure 4. Illustration of scene text rectification and recognition by the proposed ESIR: For the distorted scene text images from the SVTP and CUTE80 in the first column, the second column shows the finally restored scene text images by the proposed rectification network and the third column shows the recognized texts with and without using the proposed rectification technique, respectively. It is clear that the proposed rectification helps to improve the scene text recognition performance greatly.

Work. Table 2 shows experimental results.

As Table 2 shows, the **ESIR [ResNet, SK]** consistently outperforms the **ESIR [VGG, SK]** across all 6 datasets evaluated due to the use of a more powerful network backbone. In addition, the **ESIR [ResNET, SK+ST]** consistently outperforms the **ESIR [VGG, SK]** across the six datasets, demonstrating the value of including more data in network training. By taking a second look at the datasets, we observe that Synth90K mainly consists of English words but few sample images with numbers and punctuation, whereas SynthText contains a large amount of sample images with numbers and punctuation. This could partially explain why the inclusion of the SynthText helps more for the datasets IIT5K and CUTE that contain a large amount of images with numbers and punctuation.

The proposed ESIR achieves superior scene text recognition performance across the 6 datasets as compared with state-of-the-art techniques. For the three distorted datasets, ESIR outperforms state-of-the-art techniques under all different settings, demonstrating the advantage of the proposed iterative rectification network. In particular, **ESIR [VGG, SK]** consistently outperforms the [37] over the SVTP and CUTE when similar network backbone and training data are used. **ESIR [ResNET, SK+ST]** also outperforms [7] and [38] consistently across the ICDAR2015, SVTP and CUTE under the same setting. For the three normal datasets, ESIR also achieves state-of-the-art performance. In particular, both **ESIR [VGG, SK]** and **ESIR [ResNET, SK+ST]** outperform state-of-the-art techniques over the SVT dataset that contains a large amount of low-quality scene texts from street view imagery. For the datasets ICDAR2013, [7] achieves the best accuracy but it requires character-level bounding box annotations. [17] and [37] outperform the **ESIR [VGG,SK]** slightly on the ICDAR2013 under the same setting but they only recognize words within a 90K dictionary. [38] achieves the best accuracy on the IIIT5K, but it crops 7.2 million training images from the SynthText whereas we only crop 4 million training images.

Fig. 4 illustrates the scene text rectification and recognition by the proposed ESIR, where the three columns show several sample images from the CUTE and SVTP, the rectified images by using the **ESIR [VGG, SK]**, and the recognized texts without (at the top) and with (at the bottom) the proposed iterative rectification (incorrectly recognized texts are highlighted in red color), respectively. As Fig. 4 shows, the proposed ESIR is capable of rectifying scene text images with various perspective and curvature distortions in most cases. For the last two severely distorted scene text images, the rectification could be further improved by employing a larger number of rectification iterations (beyond default 5 under the current setting). At the same time, we can see that the proposed ESIR does not degrade scene text images that do not suffer from perspective and curvature distortions as illustrated in the sixth sample image. Further, the proposed ESIR helps to improve the scene text recognition performance greatly as shown in the third column. Note that the recognition here does not use any lexicon, and the recognition performance can be greatly improved by including a lexicon or even a large dictionary, e.g. the mis-recognized 'bookstorf' and 'tacation' from the last two sample images could be corrected if a dictionary is used.

We conjecture that the ESIR's superior recognition performance especially over the three distorted datasets is largely due to our proposed iterative rectification network. Fig. 5 compares scene text rectifications by our proposed rectification network and two state-of-the-art rectification networks in RARE [37] and ASTER [38]. As Fig. 5 shows, the proposed network produces clearly better rectifications as compared with rectifications by RARE and ASTER. The better rectifications are largely due to the robust line-fitting transformation as well as the iterative rectification framework as described in **Proposed Method**.

Original By RARE By ASTER By ESIR

Figure 5. Visual comparison of different scene text rectification methods: For the four sample images in the first column, columns 2-4 show the rectified images by using the RARE, ASTER, and ESIR, respectively. The sample images are from SVTP and CUTE80 which suffer from perspective and curvature distortions as well as complex image background. The proposed ESIR performs clearly better in scene text distortion rectification.

4.3.2 Ablation Analysis

The performance of the proposed scene text rectification and recognition technique is heavily affected by two key parameters, namely, the number of rectification iterations and the number of line segments as illustrated in Fig. 2. We study these two key parameters separately by using the Synth90K and SynthText as training data consistently. Table 3 shows experimental results, where the first set of experiments fix the number of line segments at 20 but change the rectification iteration number from 1 to 5 and the second set of experiments fix the rectification iteration number at 5 but use 5, 10 and 15 line segments, respectively.

As Table 3 shows, the model performance improves consistently when a larger number of rectification iterations is implemented. In particular, the improvement is more significant at the early stage when the first and second iterations of rectification are implemented i.e. the number of iterations changes from 0 to 1 and from 1 to 2. This can be observed more clearly over the two highly distorted datasets SVTP and CUTE as shown in Table 3. In addition, using a larger number of line segments also helps to improve the scene text recognition performance though the improvement is not as significant as implementing a larger number of rectification iterations. We conjecture that using a larger number of line segments helps to produce better estimations of text line poses which further helps to improve the scene text rectification and recognition performance.

4.3.3 Computational Costs

Though the proposed ESIR performs multiple iterations of rectification, the overall computation costs just increase slightly as compared with state-of-the-art rectification techniques without using iterations [37, 38]. In particular, the proposed ESIR with 5 rectification iterations takes 3ms per image in training with a batch-size of 64 and 28ms per image in testing with a batch-size of 1. Under the similar network setting, the ASTER takes 2.4ms per image in training and 20ms per image in testing. The similar computational cost is largely due to the proposed rectification network as shown in Table 1 which is small and computational light as compared with the feature extraction network and the sequence recognition network.

Table 3. ESIR recognition accuracy under different parameter settings: N denotes the rectification iteration number, L denotes the number of line segments, and Standard Setting uses 5 rectification iterations and 20 line segments. When N or L changes, all other parameters keep the same as the Standard Setting (Synth90K and SynthText are used in training under all settings).

Methods	ICDAR2015	SVTP	CUTE
Standard Setting: N=5, L=20	76.9	79.6	83.3
N = 0	73.9	73.2	73.4
N = 1	75.8	77.3	78.8
N = 2	76.3	78.7	81.1
N = 3	76.7	79.3	82.7
N = 4	76.9	79.5	83.1
L = 5	75.8	78.0	81.7
L = 10	76.6	78.9	82.6
L = 15	76.9	79.3	83.0

5. Conclusions

This paper presents an end-to-end trainable scene text recognition network that is capable of recognizing distorted scene texts via iterative rectification. The proposed network estimates and corrects perspective distortion and text line curvature iteratively as driven by better scene text recognition performance. In particular, a novel line-fitting transformation is designed to estimate the pose of text lines in scenes, and an iterative rectification framework is developed for optimal scene text rectification and recognition. The proposed network is also robust to parameter initialization and does not require extra annotations. Experiments over a number of public datasets demonstrate its superior performance in scene text rectification and recognition.

The integration with a detection model to realize joint optimization and achieve an end-to-end scene text reading system will be further explored.

References

[1] J. Almazan, A. Gordo, A. Fornes, and E. Valveny. Word spotting and recognition with embedded attributes. *TPAMI*, 36(12):2552–2566, 2014. 6

[2] Fan Bai, Zhanzhan Cheng, Yi Niu, Shiliang Pu, and Shuigeng Zhou. Edit probability for scene text recognition. In *CVPR*, 2018. 1

[3] Christian Bartz, Haojin Yang, and Christoph Meinel. See: towards semi-supervised end-to-end scene text recognition. In *AAAI*, 2018. 3

[4] A. Bissacco, M. Cummins, Y. Netzer, and H. Neven. Photoocr: Reading text in uncontrolled conditions. In *ICCV*, 2013. 2, 6

[5] Fred L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *TPAMI*, 11(6), 1989. 3

[6] Michal Buta, Luk Neumann, and Jir Matas. Deep textspotter: An end-to-end trainable scene text localization and recognition framework. In *ICCV*, pages 2223–2231, 2017. 2

[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 5076–5084, 2017. 2, 5, 6, 7

[8] Zhanzhan Cheng, Yangliu Xu, Fan Bai, Yi Niu, Shiliang Pu, and Shuigeng Zhou. Aon: Towards arbitrarily-oriented text recognition. In *CVPR*, 2018. 3, 5, 6

[9] Paul Clark and Majid Mirmehdi. Recognising text in real scenes. In *IJDAR*, pages 243–257, 2002. 3

[10] A. Gordo. Supervised mid-level features for word image representation. In *CVPR*, 2015. 6

[11] A. Gupta, A. Vedaldi, and A. Zisserman. Synthetic data for text localisation in natural images. In *CVPR*, 2016. 5

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 4

[13] 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

[14] Tong He, Zhi Tian, Weilin Huang, Chunhua Shen, Yu Qiao, and Changming Sun. An end-to-end textspotter with explicit alignment and attention. In *CVPR*, pages 5020–5029, 2018. 2

[15] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. In *NIPS Deep Learning Workshop*, 2014. 5

[16] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep structured output learning for unconstrained text recognition. In *ICLR*, 2015. 1, 2, 6

[17] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Reading text in the wild with convolutional neural networks. *IJCV*, 2016. 2, 6, 7

[18] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and koray kavukcuoglu. Spatial transformer networks. In *NIPS*, 2015. 2

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

[20] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, S. R. Mestre, J. Mas, D. F. Mota, J. A. Almazan, L. P. de las Heras, and et al. Icdar 2013 robust reading competition. In *ICDAR*, pages 1484–1493, 2013. 5

[21] Chen-Yu Lee and Simon Osindero. Recursive recurrent nets with attention modeling for ocr in the wild. In *CVPR*, pages 2231–2239, 2016. 2

[22] Wei Liu, Chaofeng Chen, and Kwan-Yee K. Wong. Char-net: A character-aware neural network for distorted scene text. In *AAAI*, 2018. 3

[23] Yang Liu, Zhaowen Wang, Hailin Jin, and Ian Wassell. Synthetically supervised feature learning for scene text recognition. In *ECCV*, 2018. 1

[24] Zichuan Liu, Guosheng Lin, Sheng Yang, Jiashi Feng, Weisi Lin, and Wang Ling Goh. Learning markov clustering networks for scene text detection. In *CVPR*, 2018. 1

[25] Zichuan Liu, Guosheng Lin, Sheng Yang, Fayao Liu, Weisi Lin, and Wang Ling Goh. Towards robust curve text detection with conditional spatial expansion. In *CVPR*, 2019. 1

[26] Shijian Lu, Ben Mei Chen, and Chi Chung Ko. Perspective rectification of document images using fuzzy set and morphological operations. *Image and Vision Computing*, pages 541–553, 2005. 3

[27] Shijian Lu, Ben M Chen, and Chi Chung Ko. A partition approach for the restoration of camera images of planar and curled document. *Image and Vision Computing*, 24(8):837–848, 2006. 3

[28] Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In *EMNLP*, 2015. 5

[29] Anand Mishra, Karteek Alahari, and C.V. Jawahar. Scene text recognition using higher order language priors. In *BMVC*, 2012. 1

[30] Anand Mishra, Karteek Alahari, and C.V. Jawahar. Scene text recognition using higher order language priors. In *BMVC*, 2012. 5

[31] Lukas Neumann and Jiri Matas. Real-time scene text localization and recognition. In *CVPR*, 2012. 2

[32] Luk Neumann and Ji Matas. Real-time lexicon-free scene text localization and recognition. *TPAMI*, pages 1872–1885, 2016. 2

[33] Trung Quy Phan, Palaiahnakote Shivakumara, Shangxuan Tian, and Chew Lim Tan. Recognizing text with perspective distortion in natural scenes. In *ICCV*, 2013. 2, 3, 5

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

[35] J. A. Rodrguez-Serrano, A. Gordo, and F. Perronnin. Label embedding: A frugal baseline for text recognition. *IJCV*, 2015. 1

[36] B. Shi, X. Bai, and C. Yao. An end-to-end trainable neural network for image-based sequence recognition and its appli-

cation to scene text recognition. *TPAMI*, 39(11):2298–2304, 2017. 1, 2

[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. 2, 3, 5, 6, 7, 8

[38] Baoguang Shi, Mingkun Yang, Xinggang Wang, Pengyuan Lyu, Cong Yao, and Xiang Bai. Aster: An attentional scene text recognizer with flexible rectification. *TPAMI*, 2018. 2, 3, 5, 6, 7, 8

[39] Bolan. Su and Shijian. Lu. Accurate scene text recognition based on recurrent neural network. In *ACCV*, 2014. 2

[40] Bolan. Su and Shijian. Lu. Accurate recognition of words in scenes without character segmentation using recurrent neural network. *Pattern Recognition*, pages 397–405, 2017. 1, 2

[41] Shangxuan Tian, Ujjwal Bhattacharya, Shijian Lu, Bolan Su, Qingqing Wang, Xiaohua Wei, Yue Lu, and Chew Lim Tan. Multilingual scene character recognition with co-occurrence of histogram of oriented gradients. *Pattern Recognition*, pages 125–134, 2016. 2

[42] K. Wang, B. Babenko, and S. Belongie. End-to-end scene text recognition. In *ICCV*, 2011. 2, 5

[43] Kai Wang and Serge Belongie. Word spotting in the wild. In *ECCV*, 2010. 2

[44] 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, 6

[45] Chuhui Xue, Shijian Lu, and Fangneng Zhan. Accurate scene text detection through border semantics awareness and bootstrapping. In *ECCV*, pages 355–372, 2018. 1

[46] 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. 3, 6

[47] Cong Yao, Xiang Bai, Baoguang Shi, and Wenyu Liu. Strokelets: A learned multi-scale representation for scene text recognition. In *CVPR*, 2014. 2, 6

[48] Fangneng Zhan, Shijian Lu, and Chuhui Xue. Verisimilar image synthesis for accurate detection and recognition of texts in scenes. In *ECCV*, pages 249–266, 2018. 1

[49] Fangneng Zhan, Hongyuan Zhu, and Shijian Lu. Scene text synthesis for efficient and effective deep network training. *arXiv:1901.09193*, 2019. 1

[50] Fangneng Zhan, Hongyuan Zhu, and Shijian Lu. Spatial fusion gan for image synthesis. In *CVPR*, 2019. 1