PlugNet: Degradation Aware Scene Text Recognition Supervised by a Pluggable Super-Resolution Unit

Yongqiang Mou[1]✉*, Lei Tan[2]*, Hui Yang[1], Jingying Chen[2], Leyuan Liu[2], Rui Yan[1], and Yaohong Huang[1]

[1] AI-Labs, GuangZhou Image Data Technology Co., Ltd., China
yongqiang.mou@gmail.com, huiyang865@hotmail.com, reeyree@163.com, hyh362@me.com
[2] Nercel, Central China Normal University, China
lei.tan@mails.ccnu.edu.cn,{chenjy,lyliu}@mail.ccnu.edu.cn

Abstract. In this paper, we address the problem of recognizing degradation images that are suffering from high blur or low-resolution. We propose a novel degradation aware scene text recognizer with a pluggable super-resolution unit (PlugNet) to recognize low-quality scene text to solve this task from the feature-level. The whole networks can be trained end-to-end with a pluggable super-resolution unit (PSU) and the PSU will be removed after training so that it brings no extra computation. The PSU aims to obtain a more robust feature representation for recognizing low-quality text images. Moreover, to further improve the feature quality, we introduce two types of feature enhancement strategies: Feature Squeeze Module (FSM) which aims to reduce the loss of spatial acuity and Feature Enhance Module (FEM) which combines the feature maps from low to high to provide diversity semantics. As a consequence, the PlugNet achieves state-of-the-art performance on various widely used text recognition benchmarks like IIIT5K, SVT, SVTP, ICDAR15 and etc.

Keywords: Scene Text Recognition, Neural Network, Feature Learning

1 Introduction

Scene text recognition, where the task is aiming to recognize the text in the scene images, is a long-standing computer vision issue that could be widely used in the majority of applications like driverless vehicles, product recognition, handwriting recognition, and visual recognition. Different from the Optical Character Recognition (OCR) which has been well-solved before, scene text recognition is still a challenging task owing to the variable scene conditions as occlusion, illumination, curvature distortions, perspective, etc.

* Equal Contribution

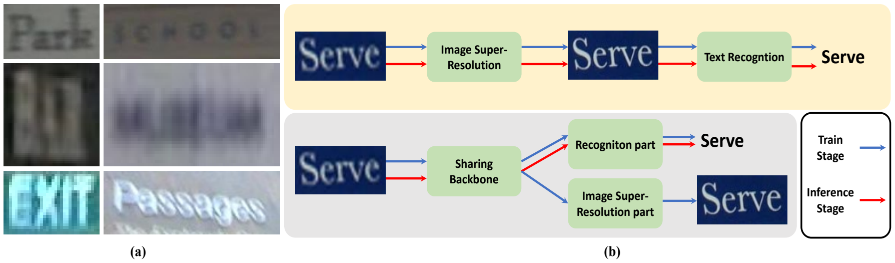


Fig. 1. (a)Low-quality images that are suffering from low-resolution, blur or shake will make a great challenge for text recognition. (b)Two types of strategies to solve the degrade images: the image-level solution and feature-level solution.

Over recent years, inspired by the practical and research value, scene text recognition attracts growing attention [33,1,45,42,18]. We have witnessed the great improvement in this area due to the powerful feature extractor like deep convolution networks like Resnet [10] and more specific methods as Aster [33]. State-of-the-art scene text recognizers nowadays are base on two types of categories: the bottom-up approaches that recognize the text by each character and top-down approaches that recognize the text by the whole image. However, both of the categories are facing a condition that using the lexicon or not will make a great gap in the recognition result. Part of the above problem is caused by the low-quality images which are suffering the noising, blurred or low-resolution as shown in Fig. 1 (a). Due to the lack of sufficient details, those images easily cause the wrong result.

Generally, when facing low-quality images in other computer vision tasks, previous works prefer to solve this problem from the image-level. Embedding a super-resolution module in the ordinary model seems like a paradigm [3]. Following this trend, TextSR[39] training an ESRGAN[40]-Aster recognition network. Unfortunately, although this method shows better visual quality than original images, it improves limited in the recognition result. Especially when considers its efficiency, this structure is far from satisfied. For example, we train the ESRGAN-Aster with the Synth90K dataset [11] with a single NVIDIA 2080Ti, it needs nearly 30 days each epoch.

Motivated by this condition, we attempt to explore a more reasonable way to solve low-quality images. Different from general methods, we attempt to solve those degradation images from the feature-level as shown in Fig. 1 (b). Base on this idea, we proposed an end-to-end trainable scene text recognizer together with a pluggable super-resolution unit (PlugNet) for auxiliary training. As shown in Fig. 2, the PlugNet can be divided into four parts: rectification network, CNN backbone, recognition network, and pluggable super-resolution unit. Specifically, the PlugNet only takes a light-weight pluggable super-resolution unit (PSU) that constructed by upsampling layers and few convolution layers to improve the feature quality in the training stage. During the inference stage, the PSU will be removed which means no extra computation.

As the most popular text recognition framework, the CNN-LSTM shows a great performance nowadays. Owing to the special structure, the input for LSTM should be a one-dimension vector. So, in most the previous works tend to use deep CNN to squeeze the height-level features maps to generate one-dimension vectors. However, CNN shows limited performance to cope with spatial-level issues like rotation[14], shift[46]. Due to the loss of spatial acuity makes it difficult for both the recognition part and the rectified part to get effective learning. Therefore, in our work, we proposed a Feature Squeeze Module (FSM) trying to maintain more spatial information in the final one-dimension vectors. Specifically, we remove the down-sampling convolutional layers in the last three blocks to maintain the feature resolution and use a 1×1 convolution layer together with a reshape layer to generate the same one-dimension vectors straightly from the feature maps. Surprisingly, by adding more spatial information into the features, recognition performance improved significantly in all of the datasets. Additionally, maintain more feature-resolution helps the PSU could easily be attached to the CNN backbone.

Affected by the above observation and Feature Pyramid Networks [21], we suppose those low-level semantics will also enhance the final sharing feature maps. We designed a Feature Enhance Module (FEM) to further combine those semantics from low to high levels.

The proposed PlugNet is compared against several state-of-the-art scene text recognition method (as [42,25,19]) on various challenging text recognition datasets like SVT [37] ,ICDAR2015 [15], SVTP [29] and etc, to demonstrate its significant advantages.

In summary, the main contributions of this paper are as follows:

- We proposed an end-to-end trainable scene text recognizer (PlugNet), which combined with a light-weight pluggable super-resolution unit to handle degradation images from the feature-level with no extra computation during the inference stage.
- Observed the importance of feature resolution in the text recognition issue, we introduced a feature squeeze module (FSM) that offers a better way to connect the CNN-based backbone and the LSTM-based recognition model. It could also be used as a fresh baseline for top-down text recognition method.
- A feature enhance module is designed to combine those semantics from low to high levels which further strengthen the sharing feature maps.
- Experimental results have demonstrated the effectiveness of the proposed PlugNet, PlugNet achieves the state-of-the-art performance on several wildly-used text recognition benchmarks.

2 Related works

Text recognition has made great progress in the last decades. Most of the text recognition methods can be divided into two categories: The bottom-up approaches which recognize the text by each character and top-down approaches which recognize the text by the whole image.

Traditional text recognition methods tend to use the bottom-up approach, in which the characters will be detected firstly by hand-crafted features and then follow with some subsequent steps like non-text component filtering, text line construction, and text line verification. These methods depend heavily on the result of character detection which using sliding-window [38,37], connected components [28] and Hough voting [43]. These methods often extract low-level hand-crafted features like HOG [43], SWT[44] to separate characters and backgrounds. Although those methods or relating occupy the major status before the deep learning era, it is still a challenge for text recognizers to reach satisfactory performance in the wild. As the most powerful extractor and classifier, we have witnessed lots of deep neural network-based frameworks further improved the performance. Bissacco *et al.* [4] utilizes the fully connected network with 5 hidden layers to extract the characters' features, then using the n-gram language model to recognition. Jaderberg *et al.* [12] introduced a CNN-based method to solve unconstrained. Charnet [22] trained a character-aware neural network for distorted scene text. Liao *et al.* [19] combined the text recognition methods with a semantic segmentation network for recognizing the text of arbitrary shapes.

On the other hand, the top-down approach has an advantage in character localization for using the whole image instead of individual characters. Jaderberg *et al.* [13] regards each word as an object and converts the text recognition to classification by training a classifier with a large number of classes. Inspired by speech recognition and Natural Language Processing [26,8], recurrent neural networks (RNN) are widely used in recent text recognition models in recent years. These methods solved this problem by converting the text recognition to sequence recognition which has high similarity. Su *et al.* [34] extracts the HOG features of the word and generates the character sequence with RNN. Busta *et al.* and Shi *et al.* [5,31] introduces an end to end model paradigm which using CNN for extracted the feature maps and using RNN to the decoder the feature maps. In recent years, the attention mechanism has inspired several works like Focus Attention[6], Moran[25].

Also, some special problems in text recognition have been proposed and solved well in the past three years. Aster, ESIR and Liao *et al.* [33,45,19] designed a rectification network to transform those irregular text. Liu *et al.* [23] proposed a data augmentation method to improve image feature learning for scene text recognition.

3 Approach

3.1 Overall Framework

The overall framework of our PlugNet is shown in Fig. 2. In order to solve the blur and low-resolution cases, we adopt the pluggable super-resolution unit (PSU) for auxiliary training to assist the recognition network. Hence, our PlugNet can be divided into four parts: rectification network, sharing CNN backbone, PSU, and recognition part.

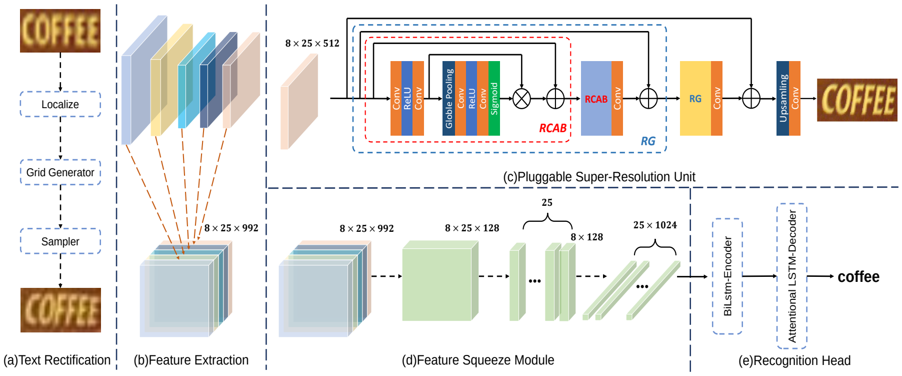


Fig. 2. Overall framework of our PlugNet. The pipeline of our method includes four parts: rectification network, sharing CNN backbone, recognition network and pluggable super-resolution unit.

Rectification Network: The rectification network aims to rectify those irregular scene text. Our method employs the same strategy as Aster[33], which shows robust performance in irregular scene text recognition. The Rectification Network is composed of three parts: localization network, grid generator, and sampler. Localization network using a CNN-based network to localize the borders of text in the input image by n control points. The grid generator will utilize the localization result and compute the transformation matrix for each pixel by Thin-Plate-Spline (TPS). Finally, the sampler is attached to generate the rectified images.

Sharing CNN Backbone: As shown in Fig. 2, the sharing CNN backbone employed the Resnet-based structure to extract the feature maps. In this work, we keep a similar structure as Aster does. We describe the detailed structure in Table. 1. Compared with the Aster, for achieving better expandability and retain more spatial information, we removed the down-sampling layers in last three CNN blocks so that the feature maps after the backbone have the dimension as $\frac{W}{4} \times \frac{H}{4} \times C$ where the W, H means the width and height of input image.

Recognition Part: Following the success of previous works like ESIR, Aster, we employed the LSTM-based method for text recognition for its advantage performance in solving whole sequences. The structure of the recognition part is shown in Fig. 2 (d)(e). Those features after the sharing CNN backbone will be used to the Feature Squeeze Module to generate the one-dimension vectors. After that, we employed a sequence-to-sequence model with attention mechanism which composed by two-layer Bidirectional LSTM (BiLSTM) with 256 hidden unit as an encoder and a two-layer attentional LSTM as a decoder. In detail, suppose the input sequence V_s whose shape can be denoted as $W \times (H \times C)$, a two-layer BiLSTM is attached to capture the long-range dependencies in both directions, obtaining a robust new sequence H_s of the same length as V_s. Next, a two-layer attentional LSTM is adopted to translate sequence H_s to a out-

Table 1. Structure of CNN Blocks in Fig. 2. Herein, the 's' means the stride of the first convolutional layer of each block.

Layers	Output size	Configurations
Block 0	32×100	$3 \times 3conv, s = 1$
Block 1	16×50	$\begin{bmatrix} 1 \times 1conv, 32 \\ 3 \times 3conv, 32 \end{bmatrix} \times 3, s = 2$
Block 2	8×25	$\begin{bmatrix} 1 \times 1conv, 64 \\ 3 \times 3conv, 64 \end{bmatrix} \times 3, s = 2$
Block 3	8×25	$\begin{bmatrix} 1 \times 1conv, 128 \\ 3 \times 3conv, 128 \end{bmatrix} \times 3, s = 1$
Block 4	8×25	$\begin{bmatrix} 1 \times 1conv, 256 \\ 3 \times 3conv, 256 \end{bmatrix} \times 3, s = 1$
Block 5	8×25	$\begin{bmatrix} 1 \times 1conv, 512 \\ 3 \times 3conv, 512 \end{bmatrix} \times 3, s = 1$

put sequence Y_s. Herein, to confirm the length of sequences, an end-of-sequence symbol (EOS) will be attached as a rest.

Pluggable SR Unit: Benefit from the FSM, the sharing CNN backbone could keep the image resolution. It helps the PSU which based-on the image super-resolution method can easily be attached in the whole network. In our work, the PSU is used to build the super-resolution images from the high-level features. This part will be detailed in Sec 3.2.

3.2 Pluggable Super-resolution Unit

As we have mentioned before, the PSU is designed to solve the degradation images from the feature level. Most of the recognition methods tend to embed the super-resolution network in the original recognition network. Limited by the efficiency, the SR-Recognition framework that solves those degradation images from the image-level shows an obvious bottleneck. Inspired by the success of multi-task learning, we utilized the PSU to help the sharing CNN backbone better represent the features of degradation images. We employed the RCAN [47] structure to build the PSU. As shown in Fig. 2 (c), we use two Residual Channel Attention Block(RCAB) to construct each Residual Group(RG). Then, two RGs are used to build the final PSU. After training, the PSU will be removed which means no extra computation in the inference stage.

3.3 Feature Enhancement

Feature Squeeze Module: As shown in Fig. 2 (d), we replaced those down-sampling convolution layers by the Feature Squeeze Module (FSM) to maintain more resolution information in the final one-dimension vectors. FSM only contains a 1×1 convolutional layer for the channel reduction and a reshape layer to generate the one-dimension vectors from the feature maps which means FSM adds few extra computations when compared to the baseline method. Based on the FSM, not only the CNN-LSTM text recognition framework has improved a lot, but also the PSU could benefit from high-resolution features which influenced a lot in the super-resolution issue.

Feature Enhance Module: Affected by the Feature Pyramid Networks [21] and the success of FSM, to further combine those semantics from low to high levels, we designed a Feature Enhance Module as shown in Fig. 2 (b). For the first two blocks, we use a down-sampling layer to transform their shape to $\frac{W}{4} \times \frac{H}{4}$. Then all of the features maps from low to high will be concatenated as the enhanced feature.

3.4 Training and Inference

Training Dataset Our model is trained on the Synth90K (**90k**) [11] and SynthText (**ST**) [9]. The Synth90K includes 9 million synthetic text images generated from 90k words lexicon. Similarly, the synthetic is also synthetic dataset (SynthText). It is generated for text detection research, so the images should be cropped to a single text. We cropped 4 million text images for training our model which keeps the same size as [45] but less than [33] who cropped 7 million text images. When training our model, we do not separate the train data and test data, all images in these two datasets are used for training.

Training Super-resolution Unit Owing to the text recognition dataset has no separation of high-resolution and low-resolution images, training the super-resolution is not an easy task. To achieve this task, we adopted two strategies as Gaussian Blur and down-up sampling to generate low-quality images. Herein, we set a probability parameter α to ensure randomness.

$$I_{blur} = \begin{cases} f_{d-u}(f_{gau}(I)), & \text{if } p_1 >= \alpha; p_2 >= \alpha \\ f_{gau}(I), & \text{if } p_1 >= \alpha; p_2 < \alpha \\ f_{d-u}(I), & \text{if } p_1 < \alpha; p_2 >= \alpha \\ I, & \text{if } p_1 < \alpha; p_2 < \alpha \end{cases} \tag{1}$$

Herein, the f_{gau} refers to the Gaussian Blur and f_{d-u} refers to the down-up sampling. The random numbers $p_1, p_2 \in [0, 1]$ and we set $\alpha = 0.5$.

Nevertheless, another challenge exists in training the super-resolution branch. For the Rectification Network, it will change distribution of each pixel which makes a huge difference between the output image and input image. Following the original super-resolution methods and taking the Rectification Network into consideration, the loss L_{sr} can be described as:

$$L_{sr} = f_{loss}(f_{rn}(I), f_{blur}(f_{rn}(I))) \tag{2}$$

where the f_{loss} means loss function of super-resolution, the f_{rn} means the Rectification Network, and I refers to the input image and f_{blur} refers to the blur function as stated before. But, following this equation will cause a tricky problem. The $f_{blur}(f_{rn}(I))$ means the data generation strategies should take effect after the Rectification Network which means the input images are high-resolution. Therefore, we use the $f_{rn}(f_{blur}(I))$ to approximate $f_{blur}(f_{rn}(I))$ as:

$$L_{sr} = f_{loss}(f_{rn}(I), f_{rn}(f_{blur}(I))) \tag{3}$$

In this way, the Rectification Network can not only learn about solving low-quality images but also simplify the whole networks thus making it easy to achieve.

Loss Functions Following the success of multi-task learning, recognition loss and super-resolution loss are combined to train our model end-to-end as:

$$L = L_{rec} + \lambda L_{sr} \tag{4}$$

where L_{rec} denotes recognition loss and L_{sr} denotes super-resolution loss. In order to balance the weight in two different tasks and keep the recognition performance, we add a parameter λ. In our method, we set the $\lambda = 0.01$.

In most of the time, the recognition problem could formulated as a classification problem[33,41], so we use a cross-entropy loss to describe L_{rec}:

$$L_{rec} = -\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N} y_{i,j}\log(s_{i,j}) \tag{5}$$

where i is the index of the sample in a batch and j is the index of the number in the label. In addition that, p is the ground truth label, s is the recognition result.

For the super-resolution branch, as has mentioned in [20] that L1 loss provides better convergence than L2 loss, we select L1 loss to train our network. So, for each pixel (i, j) in output O, the L_{sr} is employed as:

$$L_{sr} = \frac{1}{W \times H}\sum_{i=1}^{W}\sum_{j=1}^{H} \left\| O^{i,j} - I^{i,j} \right\| \tag{6}$$

Herein, the I means the ground truth of input image, W and H refers to the width and height of the input image.

4 Experiment

4.1 Datasets

We evaluate PlugNet over 7 widely used benchmarks as IIIT5K, ICDAR2003, ICDAR2013, ICDAR2015, SVTP, and CUTE80 to demonstrate its ability. Among these 7 datasets, SVT and SVTP are highly blurred and low-resolution which seems more typical. Herein, we evaluate PlugNet without any lexicon to show its robust performance.

IIIT5K[27] includes 3000 test images that are cropped from the website. Each image has a 50-word lexicon and a 1000-word lexicon in this dataset.

Street View Text (SVT)[37] contains 647 images, which are collected from the Google Street View. Many images in this dataset are suffering from noise, blur or having very low resolutions. Each image has a 50-word lexicon attached.

ICDAR 2003 (IC03)[24] contains 860 images after selection. Following [37], we discarded images that contain nonalphanumeric characters or have less than three characters.

ICDAR 2013 (IC13)[16] contains 1015 cropped text images. Most of text images inherit from IC03 and provide no lexicon.

ICDAR 2015 (IC15)[15] contains 2077 cropped text images collected by Google Glasses. IC15 is one of the most challenge datasets in text recognition in recent years. Same as IC13, no lexicon is attached to this dataset.

SVT-Perspective (SVT-P)[29] contains 645 cropped images from side-view angle snapshots in Google Street View. This dataset is not only suffering from noise, blur or having very low resolutions as SVT but also suffering from perspective distorted.

CUTE80[30] contains 288 images cropped from the 80 high-resolution scene text images. This dataset focuses on the curved text and provides no lexicon.

4.2 Implementation Details

Our method implemented in Pytorch and trained end-to-end on the Synth90k and SynthText. The training images are all from these two datasets without any data augmentation or selection. The model is trained by batches of 128 examples. Each batch includes 64 samples from Synth90k and 64 samples from SynthText. During the training, the learning rate is initiated from the 1 and is decayed to 0.1 and 0.01 respectively after 0.6M and 0.8M iterations. We adopted the ADADELTA as an optimizer to minimize the objective function. In addition, all the experiments and training are accomplished on 2 NVIDIA GeForce GTX 2080Ti 11GB GPU.

Compared to the baseline method, both FSM and FEM needs very few computations when compared to the whole network. Therefore, in the inference stage, PSU is removing, the speed of PlugNet is 22ms per image when the test batch size is 1 which is a little higher than the Aster(baseline method) as 20ms. In the training stage, PSU is adding. The speed of the Plugnet is 0.97s per batch(128), and the Aster is 0.63s per batch(128). The training process could be accelerated by a larger batch size.

Table 2. Scene text recognition performance among various widely used benchmarks under different feature resolutions.

Resolution	Data	SVT	SVTP	IIIT5K	IC03	IC13	IC15	CUTE80
1×25	*90K*	85.2	76.1	80.7	91.8	89.3	69.3	66.3
2×25	*90K*	$87.0_{\uparrow 1.8}$	$78.1_{\uparrow 2.0}$	$82.7_{\uparrow 2.0}$	$92.3_{\uparrow 1.5}$	$89.4_{\uparrow 0.1}$	$69.3_{\uparrow 0}$	$68.4_{\uparrow 2.1}$
4×25	*90K*	$87.9_{\uparrow 0.9}$	$79.5_{\uparrow 1.4}$	$82.2_{\downarrow 0.5}$	$92.7_{\uparrow 0.4}$	$89.8_{\uparrow 0.4}$	$71.4_{\uparrow 2.1}$	$69.1_{\uparrow 0.7}$
8×25	*90K*	$\mathbf{89.0}_{\uparrow 2.1}$	$\mathbf{82.0}_{\uparrow 2.5}$	$\mathbf{85.3}_{\uparrow 3.1}$	$\mathbf{94.3}_{\uparrow 1.6}$	$\mathbf{91.0}_{\uparrow 1.2}$	$\mathbf{73.6}_{\uparrow 2.2}$	$\mathbf{69.1}_{\uparrow 0}$

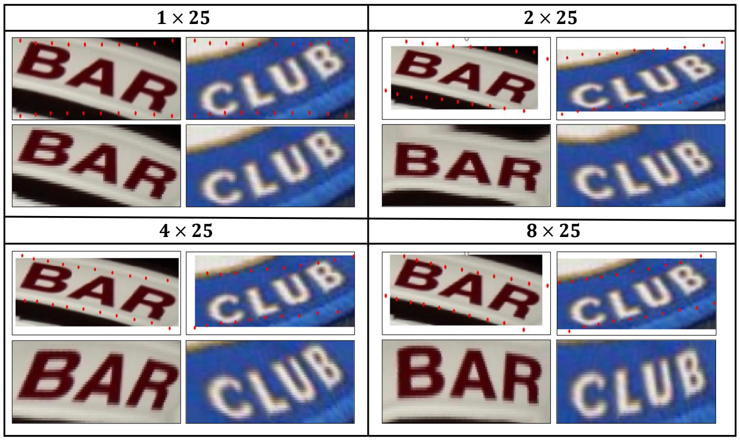


Fig. 3. Visualization of rectification results under different feature resolutions. Please zoom in to see the distribution of control points..

4.3 Ablation Study

Effectiveness of Feture Squeeze Module: The FSM is designed to offer better one-dimension vectors to connect the CNN part and the LSTM part. To further analyze the influence of the CNN feature resolution in the text recognition issue, we trained four networks with different CNN feature resolution under the 90K dataset. For better comparison, we change the number of the channel of the 1×1 convolutional layer to keep the output of FSM has the same dimension as 25×1024.

Table. 2 shows the result of four different networks in seven widely used text recognition datasets. With the broadening of the feature-resolution, the recognition accuracy gets increased in all of the datasets. It has already illustrated the importance of spatial information in text recognition tasks.

Additionally, we visualized the rectified results of the above four networks in Fig. 3. Clearly, decrease the loss spatial acuity helps the Rectification Network shows a much better location result of control points which helps the recognizer could overcome tougher irregular cases.

Effectiveness of Feature Enhance Module: As stated before, to obtain much robust feature maps for the recognition network, we designed a Feature

Table 3. Ablation study of PlugNet on several typical datasets. Herein, the SR-Plugnet is training following the SR-Recognition framework without PSU.

Methods	FSM	FEM	Data_Aug	ESRGAN	PSU	**SVT**	**SVTP**	**IC15**	**CUTE80**
Baseline(R) [33]	✗	✗	✗	✗	✗	89.5	78.5	76.1	79.5
PlugNet(R)	✓	✗	✗	✗	✗	$90.0_{\uparrow 0.5}$	$80.8_{\uparrow 2.3}$	$78.2_{\uparrow 2.1}$	$82.6_{\uparrow 3.1}$
PlugNet(R)	✓	✓	✗	✗	✗	$90.6_{\uparrow 0.6}$	$81.6_{\uparrow 0.8}$	$80.2_{\uparrow 2.0}$	$83.7_{\uparrow 1.1}$
PlugNet	✓	✓	✓	✗	✗	$89.8_{\downarrow 0.8}$	$82.2_{\uparrow 0.6}$	$79.8_{\downarrow 0.4}$	$81.6_{\downarrow 2.1}$
SR-PlugNet	✓	✓	✓	✓	✗	$90.6_{\uparrow 0.8}$	$80.8_{\downarrow 1.4}$	$79.4_{\downarrow 0.4}$	$82.6_{\uparrow 1.0}$
PlugNet	✓	✓	✓	✗	✓	$\mathbf{92.3}_{\uparrow 1.7}$	$\mathbf{84.3}_{\uparrow 3.5}$	$\mathbf{82.2}_{\uparrow 2.8}$	$\mathbf{85.0}_{\uparrow 2.4}$

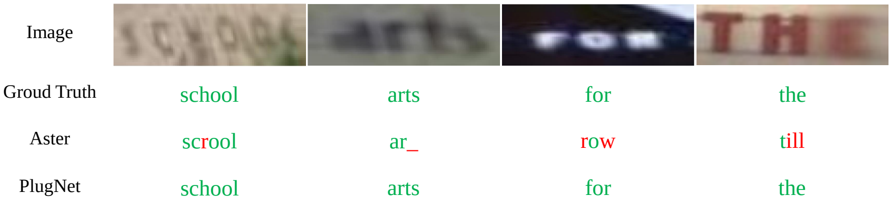


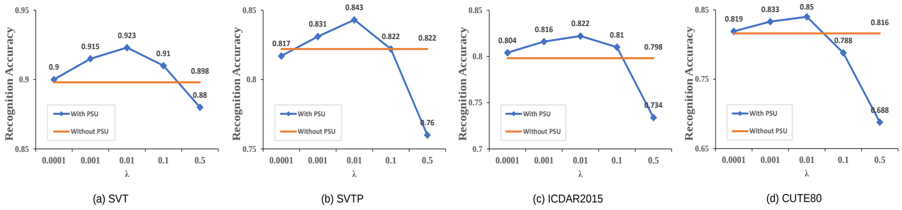
Fig. 4. Several recognition results produced by our PlugNet and Baseline method Aster [33] in low-quality text images of the SVT dataset.

Enhance Module (FEM) that aims to combine the feature maps from low to high to provide diversity semantics.

To analyze the influence of FEM, we training the FSM enhanced model with and without FEM. Herein, we chose four typical datasets for evaluation: SVT that including many highly-blurred images, SVTP that suffering from blur and perspective, CUTE80 that contains many irregular cases, and the most widely used challenging dataset-IC15. The experimental result in Table 3 has illustrated the efficiency of FEM. We observe that adding the FEM, all of the results in these four datasets get improved as 0.6%, 0.8%, 2%, 1.1% in final recognition accuracy when compared to the model without FEM. It indicates that the feature enhancement module has improved feature quality by low-level semantics which in turn improved the recognition performance.

Effectiveness of Pluggable Super-Resolution Unit: So far, recognizing those scene texts with highly-blurred and low-resolution remains a challenging task, thus we employed the super-resolution method to better solve this problem.

As shown in Table 3, we conduct a set of ablation experiments by adding the PSU or not. Owing to PSU, we use the generated data rather than the raw data for training. Hence, to better compare the influence of PSU, we train the network without PSU both in raw data and generated data. The results show that the recognizer with PSU produced a much better performance in solving low-quality scene text images. Coupled with PSU, the recognition accuracy in SVT, SVTP, IC15 and CUTE80 has improved from 89.8%, 82.2%, 79.8%, 81.6% to 92.3%, 84.3%, 82.6%, 85.0%. In the visual level, we chose the recognition results of four



(a) SVT (b) SVTP (c) ICDAR2015 (d) CUTE80

Fig. 5. Results of PlugNet recognition accuracy under different parameter λ value on various datasets.

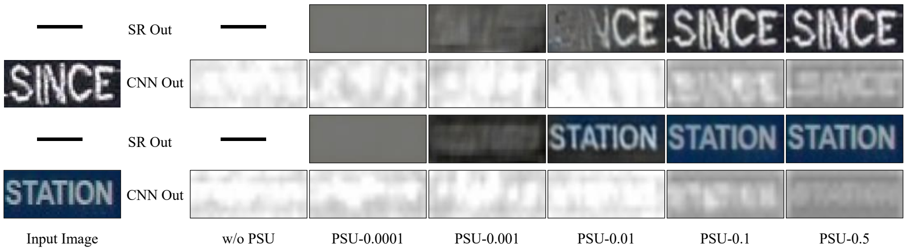


Fig. 6. Visualizing results of PlugNet under different parameter λ value.

low-quality images in the SVT dataset in Fig.4 to show the improvement of our method when compared to the baseline method.

Herein, to make a comparison between the PSU method and general super-resolution combine with the recognizer method, we trained an end-to-end SR-Plugnet (no PSU) model under the 90K and ST datasets. The SR-Plugnet(no PSU) using the ESRGAN for the super-resolution part and Plugnet (no PSU) for the recognition part. Obviously, as shown in Table 3, the SR-Plugnet shows limited improvement in recognition results, which is a similar case to the TextSR[39]. Of course, we think SR-Plugnet may reach a higher performance when using a better training strategy, more proper parameters or adding more effective data. Like the TextSR using extra selected data for training to reach an even better performance. But consider the structure and efficiency, PSU is obviously a better choice to solve low-quality images.Finally, about the effect on the sharing feature maps will be discussed in the Sec. 4.4, in which we set different weights of PSU to show the changes of feature gradually.

4.4 Experiments on the Parameter λ

Proper parameters are necessary for training a multi-task network. In this work, we use the parameter λ in the formula 4 to balance the recognition branch and the super-resolution branch. To analyze the influence of the parameter λ, we train the model with the λ from 0.0001 to 0.5. We extensively evaluate these models on the challenging SVT, SVTP, ICDAR2015, and CUTE80 to demonstrate the influence of λ.

Table 4. Scene text recognition performance of PlugNet among various widely used benchmarks. The methods marked with * indicate they use the character-level annotations that will highly improve the performance in irregular text recognition datasets like CUTE80.

Methods	Data	IIIT5K	SVT	IC03	IC13	IC15	SVTP	CUTE80
Jaderberg *et al.*[12]	*90K*	-	71.7	89.6	81.8	-	-	-
Jaderberg *et al.*[13]	*90K*	-	80.7	93.1	90.8	-	-	-
Shi *et al.*[32]	*90K*	81.9	81.9	90.1	88.6	-	71.8	59.2
Lee *et al.*[17]	*90K*	78.4	80.7	88.7	90.0	-	-	-
Wang *et al.*[36]	*90K*	80.8	81.5	91.2	-	-	-	-
Cheng *et al.*[6]	*90K, ST*	87.4	85.9	94.2	93.3	70.6	-	-
Cheng *et al.*[7]	*90K, ST*	87.0	82.8	91.5	-	68.2	73.0	76.8
Liu *et al.*[22]	*90K, ST*	92.0	85.5	92.0	91.1	74.2	78.9	-
Bai *et al.*[2]	*90K, ST*	88.3	87.5	94.6	94.4	73.9	-	-
Liu *et al.*[23]	*90K, ST*	89.4	87.1	94.7	94.0	-	73.9	62.5
Luo *et al.*[25]	*90K, ST*	91.2	88.3	95.0	92.4	68.8	76.1	77.4
Liao *et al.*[19]	*90K, ST*	91.9	86.4	-	91.5	-	-	79.9
Zhan *et al.*[45]	*90K, ST*	93.3	90.2	-	91.3	76.9	79.6	83.3
Yang *et al.*[42]	*90K, ST**	**94.4**	88.9	95.0	93.9	78.7	80.8	87.5
Wan *et al.*[35]	*90K, ST**	93.9	90.1	-	92.9	79.4	84.3	85.2
Liao *et al.* -Seg[18]	*90K*	94.0	87.2	93.1	92.3	73.8	76.3	82.6
Liao *et al.* -SAM[18]	*90K, ST**	93.9	90.6	95.2	**95.3**	77.3	82.2	**87.8**
Aster(Baseline)[33]	*90K, ST*	93.4	89.5	94.5	91.8	76.1	78.5	79.5
TextSR(SR-Aster)[39]	*90K, ST*	92.5	87.2	93.2	91.3	75.6	77.4	78.9
Ours	*90K, ST*	**94.4**	**92.3**	**95.7**	95.0	**82.2**	**84.3**	85.0

In Fig. 5, we set a baseline as training the whole network without the PSU. Obviously, adding the PSU improves the recognition performance in all of the datasets which also demonstrates the efficiency of PSU. From Fig. 5 (a)-(d), we can observe that the recognition accuracy improves monotonically when the λ is smaller than 0.01. After that, with the increase of the , the recognition accuracy decreased and will even have a negative impact. Obviously, 0.01 seems like a best choice of λ in most of the situation.

Based on this observation, we visualized the output feature maps of the sharing CNN backbone and the output of PSU in Fig. 6 to analyze the influence caused by the PSU. To visualize the feature maps, we calculate the average among the channels to generate one-channel images to represent the feature result. Since the Rectification Network will change the pixel distribution of each image, we removed this part when visualizing for better comparison. It is clear that by the increase of λ, the super-resolution result shows a growing visual quality. However, in the feature maps, there exist two types of effects: due to the increase of λ, the feature suffers much less noising, blur which helps the recognition part. Meanwhile, the increase of λ, let the sharing CNN backbone focus more on low-level images to rebuild the super-resolution images which makes a negative impact on the text recognition. These two types of effects make the PlugNet be sensitive to the λ in this work.

4.5 Comparison with State of the Art

We also compare our PlugNet with previous state-of-the-art text recognition methods on various widely used benchmarks to indicate the superiority of our method. Table 4 summarizes the recognition result among 7 widely used datasets including IIIT5K, SVT, IC03, IC13, IC15, SVTP, and CUTE80. Herein, we evaluated all the datasets without lexicon.

As Table 4 shows, our PlugNet outperforms all the previous state-of-the-art performance in 6 datasets and achieves competitive accuracy to the state-of-the-art techniques in the remain CUTE80 datasets. Especially in two low-quality text datasets as SVT and SVTP, our method shows a much robust performance. The CUTE80 dataset focuses on the high-resolution curved text images, so those methods that using the character level annotations to training the rectification part will perform much better.

Our method shows a significant improvement in most of the cases by using the combination of FSM, FEM and PSU. The PSU, FSM, and FEM are designed to obtaining more robust feature maps with high efficiency and performance. So, this constructure may also be useful for solving low-quality images in other computer vision tasks.

5 Conclusion

In this paper, we proposed an end-to-end trainable degradation aware scene text recognizer called PlugNet in short. The proposed method combined the pluggable super-resolution unit (PSU) to solve the low-quality text recognition from the feature-level. It only takes acceptable extra computation in the training stage and no additional computation in the inference stage. With PSU our method shows a significant improvement in a low-quality image feature representation that in turn improves the recognition accuracy. Moreover, in this paper, we further analyzed the important role of feature resolution in the text recognition issue and proposed the FSM for a better connection between CNN and LSTM for top-down recognition framework. Also, the FEM is attached to enhanced the backbone features by introducing those low-level semantics. Experiments show that FSM and FEM also improved performance markedly. Finally, our PlugNet achieves state-of-the-art performance on various widely used text recognition benchmark datasets, especially on SVT and SVTP which include many low-quality text images.

Acknowledgement

This work was supported by the Project of the National Natural Science Foundation of China Grant No. 61977027 and No. 61702208, the Hubei Province Technological Innovation Major Project Grant No. 2019AAA044 and the Colleges Basic Research and Operation of MOE Grant No. CCNU19Z02002, CCNU18KFY02.

References

1. Baek, J., Kim, G., Lee, J., Park, S., Han, D., Yun, S., Oh, S.J., Lee, H.: What is wrong with scene text recognition model comparisons? dataset and model analysis. In: Proceedings of the ICCV (2019)
2. Bai, F., Cheng, Z., Niu, Y., Pu, S., Zhou, S.: Edit probability for scene text recognition. In: Proceedings of the CVPR. pp. 1508–1516 (2018)
3. Bai, Y., Zhang, Y., Ding, M., Ghanem, B.: Finding tiny faces in the wild with generative adversarial network. In: Proceedings of the CVPR. pp. 21–30 (2018)
4. Bissacco, A., Cummins, M., Netzer, Y., Neven, H.: Photoocr: Reading text in uncontrolled conditions. In: Proceedings of the ICCV. pp. 785–792 (2013)
5. Busta, M., Neumann, L., Matas, J.: Deep textspotter: An end-to-end trainable scene text localization and recognition framework. In: Proceedings of the ICCV. pp. 2204–2212 (2017)
6. Cheng, Z., Bai, F., Xu, Y., Zheng, G., Pu, S., Zhou, S.: Focusing attention: Towards accurate text recognition in natural images. In: Proceedings of the ICCV. pp. 5076–5084 (2017)
7. Cheng, Z., Xu, Y., Bai, F., Niu, Y., Pu, S., Zhou, S.: Aon: Towards arbitrarily-oriented text recognition. In: Proceedings of the CVPR. pp. 5571–5579 (2018)
8. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder-decoder for statistical machine translation. In: Proceedings of the EMNLP. p. 1724C1734 (2014)
9. Gupta, A., Vedaldi, A., Zisserman, A.: Synthetic data for text localisation in natural images. In: Proceedings of the CVPR. pp. 2315–2324 (2016)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the CVPR. pp. 770–778 (2016)
11. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Synthetic data and artificial neural networks for natural scene text recognition. In: Proceedings of the NIPS (2014)
12. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Deep structured output learning for unconstrained text recognition. In: Proceedings of the ICLR (2015)
13. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Reading text in the wild with convolutional neural networks. IJCV **116**(1), 1–20 (2016)
14. Jaderberg, M., Simonyan, K., Zisserman, A., et al.: Spatial transformer networks. In: Proceedings of the NeurIPS. pp. 2017–2025 (2015)
15. Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., Matas, J., Neumann, L., Chandrasekhar, V.R., Lu, S., et al.: Icdar 2015 competition on robust reading. In: Proceedings of the ICDAR. pp. 1156–1160. IEEE (2015)
16. Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., i Bigorda, L.G., Mestre, S.R., Mas, J., Mota, D.F., Almazan, J.A., De Las Heras, L.P.: Icdar 2013 robust reading competition. In: Proceedings of the ICDAR. pp. 1484–1493. IEEE (2013)
17. Lee, C.Y., Osindero, S.: Recursive recurrent nets with attention modeling for ocr in the wild. In: Proceedings of the CVPR. pp. 2231–2239 (2016)
18. Liao, M., Lyu, P., He, M., Yao, C., Wu, W., Bai, X.: Mask textspotter: An end-to-end trainable neural network for spotting text with arbitrary shapes. IEEE T-PAMI (2019)
19. Liao, M., Zhang, J., Wan, Z., Xie, F., Liang, J., Lyu, P., Yao, C., Bai, X.: Scene text recognition from two-dimensional perspective. In: Proceedings of the AAAI. vol. 33, pp. 8714–8721 (2019)

20. Lim, B., Son, S., Kim, H., Nah, S., Mu Lee, K.: Enhanced deep residual networks for single image super-resolution. In: Proceedings of the CVPRW. pp. 136–144 (2017)
21. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the CVPR. pp. 2117–2125 (2017)
22. Liu, W., Chen, C., Wong, K.Y.K.: Char-net: A character-aware neural network for distorted scene text recognition. In: Proceedings of the AAAI (2018)
23. Liu, Y., Wang, Z., Jin, H., Wassell, I.: Synthetically supervised feature learning for scene text recognition. In: Proceedings of the ECCV. pp. 435–451 (2018)
24. Lucas, S.M., Panaretos, A., Sosa, L., Tang, A., Wong, S., Young, R., Ashida, K., Nagai, H., Okamoto, M., Yamamoto, H., et al.: Icdar 2003 robust reading competitions: entries, results, and future directions. IJDAR **7**(2-3), 105–122 (2005)
25. Luo, C., Jin, L., Sun, Z.: Moran: A multi-object rectified attention network for scene text recognition. Pattern Recognition **90**, 109–118 (2019)
26. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: Proceedings of the ICLR (2013)
27. Mishra, A., Alahari, K., Jawahar, C.: Top-down and bottom-up cues for scene text recognition. In: Proceedings of the CVPR. pp. 2687–2694. IEEE (2012)
28. Neumann, L., Matas, J.: Real-time scene text localization and recognition. In: Proceedings of the CVPR. pp. 3538–3545. IEEE (2012)
29. Quy Phan, T., Shivakumara, P., Tian, S., Lim Tan, C.: Recognizing text with perspective distortion in natural scenes. In: Proceedings of the ICCV. pp. 569–576 (2013)
30. Risnumawan, A., Shivakumara, P., Chan, C.S., Tan, C.L.: A robust arbitrary text detection system for natural scene images. Expert Systems with Applications **41**(18), 8027–8048 (2014)
31. Shi, B., Bai, X., Yao, C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE T-PAMI **39**(11), 2298–2304 (2016)
32. Shi, B., Wang, X., Lyu, P., Yao, C., Bai, X.: Robust scene text recognition with automatic rectification. In: Proceedings of the CVPR. pp. 4168–4176 (2016)
33. Shi, B., Yang, M., Wang, X., Lyu, P., Yao, C., Bai, X.: Aster: An attentional scene text recognizer with flexible rectification. IEEE T-PAMI (2018)
34. Su, B., Lu, S.: Accurate scene text recognition based on recurrent neural network. In: Proceedings of the ACCV. pp. 35–48. Springer (2014)
35. Wan, Z., He, M., Chen, H., Bai, X., Yao, C.: Textscanner: Reading characters in order for robust scene text recognition. In: Proceedings of the AAAI (2020)
36. Wang, J., Hu, X.: Gated recurrent convolution neural network for ocr. In: Proceedings of the NIPS. pp. 335–344 (2017)
37. Wang, K., Babenko, B., Belongie, S.: End-to-end scene text recognition. In: Proceedings of the ICCV. pp. 1457–1464. IEEE (2011)
38. Wang, K., Belongie, S.: Word spotting in the wild. In: Proceedings of the ECCV. pp. 591–604. Springer (2010)
39. Wang, W., Xie, E., Sun, P., Wang, W., Tian, L., Shen, C., Luo, P.: Textsr: Content-aware text super-resolution guided by recognition. arXiv:1909.07113 (2019)
40. Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y., Change Loy, C.: Esrgan: Enhanced super-resolution generative adversarial networks. In: Proceedings of the ECCV (2018)

41. Wei, K., Yang, M., Wang, H., Deng, C., Liu, X.: Adversarial fine-grained composition learning for unseen attribute-object recognition. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3741–3749 (2019)
42. Yang, M., Guan, Y., Liao, M., He, X., Bian, K., Bai, S., Yao, C., Bai, X.: Symmetry-constrained rectification network for scene text recognition. In: Proceedings of the ICCV (2019)
43. Yao, C., Bai, X., Shi, B., Liu, W.: Strokelets: A learned multi-scale representation for scene text recognition. In: Proceedings of the CVPR. pp. 4042–4049 (2014)
44. Yin, X.C., Yin, X., Huang, K., Hao, H.W.: Robust text detection in natural scene images. IEEE T-PAMI **36**(5), 970–983 (2013)
45. Zhan, F., Lu, S.: Esir: End-to-end scene text recognition via iterative image rectification. In: Proceedings of the CVPR. pp. 2059–2068 (2019)
46. Zhang, R.: Making convolutional networks shift-invariant again. In: Proceedings of the ICML (2019)
47. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: Proceedings of the ECCV. pp. 286–301 (2018)