Text Gestalt: Stroke-Aware Scene Text Image Super-Resolution

Jingye Chen[1], Haiyang Yu[1], Jianqi Ma[2], Bin Li[*1], Xiangyang Xue[*1]

[1]Shanghai Key Laboratory of Intelligent Information Processing
School of Computer Science, Fudan University
[2]The Hong Kong Polytechnic University
{jingyechen19, hyyu20, libin, xyxue}@fudan.edu.cn, jianqi.ma@connect.polyu.hk

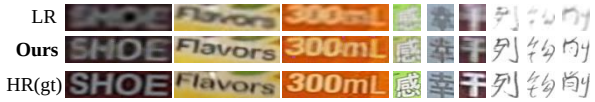
Abstract

In the last decade, the blossom of deep learning has witnessed the rapid development of scene text recognition. However, the recognition of low-resolution scene text images remains a challenge. Even though some super-resolution methods have been proposed to tackle this problem, they usually treat text images as general images while ignoring the fact that the visual quality of strokes (the atomic unit of text) plays an essential role for text recognition. According to Gestalt Psychology, humans are capable of composing parts of details into the most similar objects guided by prior knowledge. Likewise, when humans observe a low-resolution text image, they will inherently use partial stroke-level details to recover the appearance of holistic characters. Inspired by Gestalt Psychology, we put forward a Stroke-Aware Scene Text Image Super-Resolution method containing a Stroke-Focused Module (SFM) to concentrate on stroke-level internal structures of characters in text images. Specifically, we attempt to design rules for decomposing English characters and digits at stroke-level, then pre-train a text recognizer to provide stroke-level attention maps as positional clues with the purpose of controlling the consistency between the generated super-resolution image and high-resolution ground truth. The extensive experimental results validate that the proposed method can indeed generate more distinguishable images on TextZoom and manually constructed Chinese character dataset Degraded-IC13. Furthermore, since the proposed SFM is only used to provide stroke-level guidance when training, it will not bring any time overhead during the test phase. Code is available at https://github.com/FudanVI/FudanOCR/text-gestalt.

Introduction

In recent years, scene text recognition has achieved tremendous progress owing to the rapid development of deep learning. It has been widely used in many real-world applications such as auto-driving (Zhang et al. 2020a), ID card recognition (Satyawan et al. 2019), signature identification (Ren et al. 2020), etc. Although the recently proposed recognizers become stronger as reported, we observe that low-resolution (LR) text images still pose great challenges for them. In this context, a super-resolution module is required as a preprocessor to recover the missing details of LR images.

(a) According to Gestalt Psychology, humans can construct the complete appearance of the spotted dog in the minds with the guidance of local details.



(b) Strokes help humans recover the whole characters in their minds.

Figure 1: For incomplete or blurred images, detailed information (e.g., spots or strokes) play a significant role during recovery. Our method can generate recognizable English and Chinese text images with the guidance of stroke details.

The previous super-resolution methods usually try to learn degradation patterns through HR-LR pairs with global loss functions (e.g., L1 or L2 loss) to recover every pixel in text images (Xu et al. 2017; Pandey et al. 2018). These methods, however, usually view text images as general images regardless of text-specific properties. Recently, a few methods attempt to take several text-specific properties into account, which achieve better performance in terms of both image quality and recognition accuracy. For example, PlugNet (Yan and Huang 2020) employs a multi-task framework with the purpose of jointly optimizing super-resolution and text recognition tasks in one model. In (Wang et al. 2020), the authors introduced a Text Super-Resolution Network (TSRN) via appending two recurrent layers in the backbone to capture sequential information of text images. The recently proposed Scene Text Telescope (STT) (Chen, Li, and Xue 2021a) introduces text priors into the model by proposing a position-aware module and a content-aware module. The concurrent work TPGSR (Ma, Guo, and Zhang 2021) incorporates text-specific semantic features to each block in the backbone and exerts an iterative way to enhance text images. Through observations, the text priors used in these works usually regard ***character*** as the smallest unit of text lines, whereas ignoring the significance of more detailed internal structures. In this paper, we take a step further to answer the critical question: *Can text priors at a more fine-grained level (e.g., **stroke**) benefit the super-resolution procedure?*

*Corresponding author

According to **Gestalt Psychology** (Köhler 1967), humans can compose parts of details into the most similar objects guided by prior knowledge. As is shown in Figure 1(a), humans can inherently recover the whole appearance of the spotted dog with the guidance of local details such as spots. Likewise, for blurred text images, strokes that act as local details play an indispensable role in the recovery process. As is shown in Figure 1(b), even though the character "m" in "300ml" looks blurred, we can easily recover it when discovering the three parallel vertical strokes.

Inspired by Gestalt Psychology, we propose a stroke-aware Scene Text Image Super-Resolution method that utilizes a Stroke-Focused Module (SFM) to take advantage of fine-grained stroke-level attention maps generated by an auxiliary recognizer as guidance for recovery. Different from most existing recognizers (Shi, Bai, and Yao 2016; Shi et al. 2018; Luo, Jin, and Sun 2019) that predict at character-level, we design a recognizer working at the stroke level, thus is capable of generating more fine-grained attention maps. To validate the effectiveness of our method, we employ some recognizers and image quality metrics to evaluate the generated SR images. The experimental results show that our method can indeed achieve state-of-the-art performance on the TextZoom and designed Chinese character dataset Degraded-IC13 in terms of recognition accuracy. Moreover, since the proposed SFM is only used when training, it will not bring any time overhead during testing. Our contributions are listed as follows:

- We attempt to design rules for recognizing English letters and digits at the stroke level to provide more fine-grained attention-level guidance.
- Inspired by Gestalt Psychology, we propose a Stroke-Focused Module (SFM) to concentrate more on stroke regions with the guidance of stroke-level attention maps.
- Compared to the previous methods, our method can generate more distinguishable text images on the TextZoom and Degraded-IC13 in terms of recognition accuracy without bringing any time overhead during testing.

Related Work

Single Image Super-Resolution

Single image super-resolution aims to generate an SR image based on its LR counterpart while recovering several missing details. In the deep learning era, the first CNN-based method named SRCNN (Dong et al. 2014) establishes an end-to-end approach to learning the mapping from LR to HR images using a shallow network, while achieving better performance compared with previous traditional methods. EDSR (Lim et al. 2017) proposes a deep model by using multiple residual blocks for better representation and removing several unnecessary batch normalization layers in the residual blocks. MSRN (Li et al. 2018) introduces filters of different sizes in two branches while extracting multi-scale features.

Text Image Super-Resolution

Traditional methods usually utilize classical machine learning algorithms to upsample LR images. In (Capel and Zisserman 2000), a Maximum *a posteriori* approach combined with a Huber prior was applied to TISR. In (Dalley, Freeman, and Marks 2004), a Bayesian framework was proposed to upsample binary text images. However, the design of traditional features was time-consuming and the low-capacity features were subpar to tackle such task (Chen et al. 2021). Recently, PlugNet (Yan and Huang 2020) designs a multi-task framework by optimizing recognition and super-resolution branches in one model. To capture sequential information of text images, in (Wang et al. 2020), the authors proposed a TSRN containing two BLSTMs. STT (Chen, Li, and Xue 2021a) contains two text-focused modules including a position-aware module and a content-aware module providing text priors. TPGSR (Ma, Guo, and Zhang 2021) combines text priors in the encoder and employs an iterative manner to enhance low-resolution images. However, these methods usually view characters as the smallest units without considering the more fine-grained details like strokes.

Scene Text Recognition

Traditional methods usually adopt a bottom-up approach to recognize text images (Wang and Belongie 2010; Wang, Babenko, and Belongie 2011; Neumann and Matas 2012). Specifically, they first detect and classify separated characters and then compose them into text lines with the guidance of language models or lexicons. In the deep learning era, CRNN (Shi, Bai, and Yao 2016) combines CNN and RNN as the encoder and employs a CTC-based decoder (Graves et al. 2006) to maximize the probability of paths that can reach the ground truth. ASTER (Shi et al. 2018) introduces a Spatial Transformer Network (STN) (Jaderberg et al. 2015) to rectify irregular text images in an unsupervised manner for better recognition. SEED (Qiao et al. 2020) tries to capture global semantic features of text images with the guidance of a pre-trained fastText model. Although the semantics-based methods are capable of tackling those images with local missing details such as occlusion, they still have difficulty in recognizing low-resolution images with global missing details. Therefore, a preprocessor is required for recovering the details of low-resolution images.

Methodology

In this section, we introduce two modules and the way to decompose characters. At last, we introduce the overall loss function. The overall architecture is shown in Figure 2.

Pixel-wise Supervision Module

The existing super-resolution backbones usually follow this design: **(1)** Employ a series of stacked CNN layers to build up a backbone for extracting features, whose height and width are the same as the original images while containing more channels; **(2)** Utilize a pixel shuffle module containing multiple CNN layers to reshape the generated maps. Consequently, a super-resolution image is generated with a larger size. The widely used backbones contain SRCNN (Dong et al. 2014), SRResNet (Ledig et al. 2017), TSRN (Wang et al. 2020), TBSRN (Chen, Li, and Xue 2021a), etc.

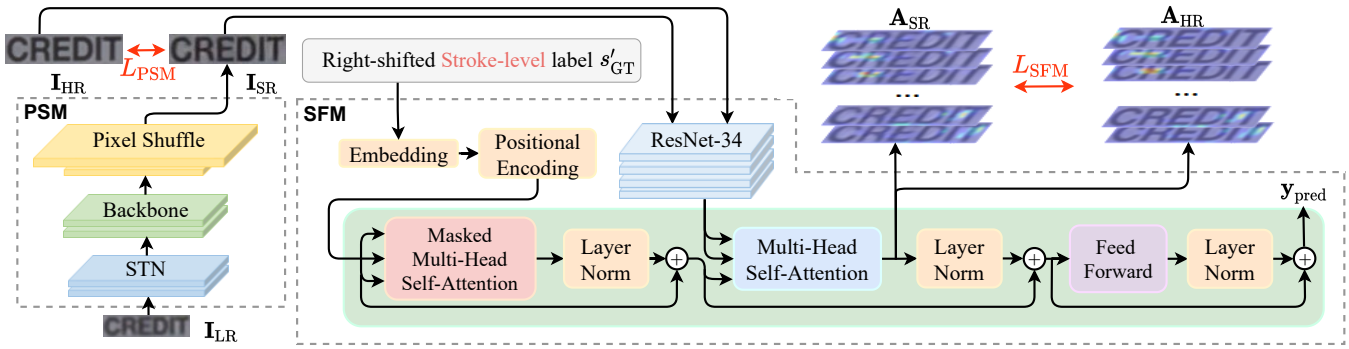


Figure 2: The overall architecture of our method. It contains two modules, including a Pixel-wise Supervision Module (PSM) to recover the color and contour of text images and a Stroke-Focused Module (SFM) to highlight the details of stroke regions.

Please note that there may exist a misalignment problem between LR-HR pairs (Wang et al. 2020). For example, in the TextZoom dataset, since the pairs are manually cropped and matched by humans, there are several pixel-level offsets that pose difficulties for the super-resolution methods. Hence, we follow (Wang et al. 2020) to append a STN (Jaderberg et al. 2015) before the backbone to alleviate this problem. Finally, the PSM module is supervised by an L2 loss. We denote the HR images as $\mathbf{I}_{\text{HR}}$ and the generated SR images as $\mathbf{I}_{\text{SR}}$. The loss is calculated as follows:

$$L_{\text{PSM}} = ||\mathbf{I}_{\text{SR}} - \mathbf{I}_{\text{HR}}||_2^2 \tag{1}$$

Stroke-Level Decomposition

Strokes are the atomic units of characters whatever the language it is. In this section, we try to decompose each character into a stroke sequence and construct stroke-level text labels for English characters, digits, and Chinese characters.

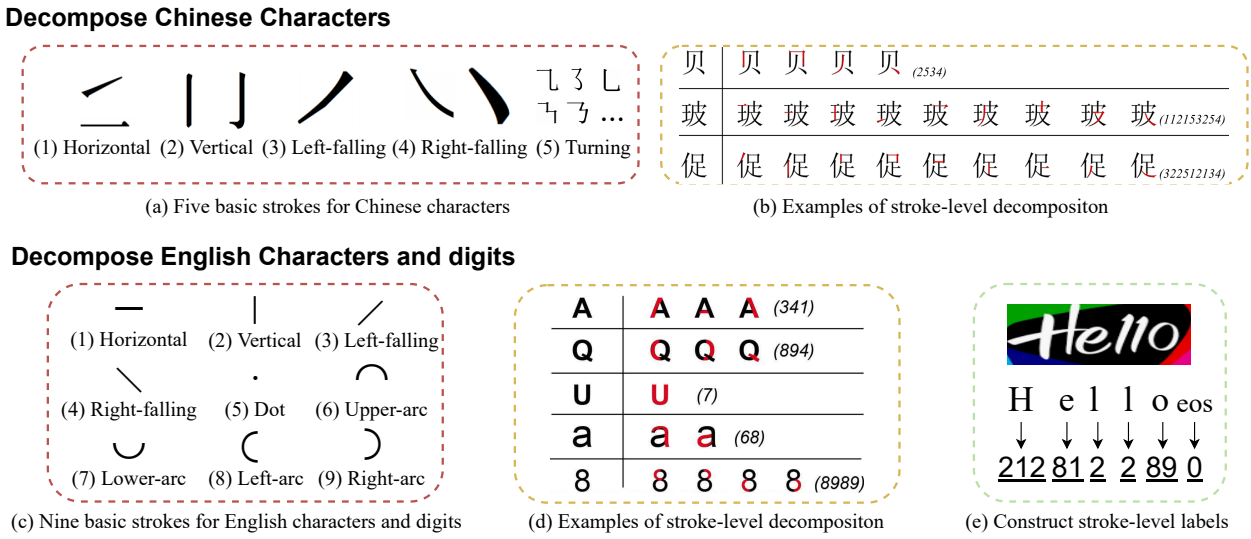
Decompose Chinese characters. According to Unicode Han Database, there are five basic strokes of Chinese characters including *horizontal*, *vertical*, *left-falling*, *right-falling*, and *turning*. Each character has a unique stroke sequence and some examples are shown in Figure 3(b).

Decompose English letters and digits. Derived from the approaches to decomposing Chinese characters, we attempt to create stroke encoding for English characters and digits: **(1)** Break down the characters and digits in more simplified structures, i.e., nine basic strokes (see Figure 3(c)). We reduce the total category number for the recognition models to generate better-learned and fine-grained supervision. **(2)** Represent each character as a sequence of these basic strokes (see Figure 3(d)) **(3)** Concatenate the stroke sequences of each character and pad a stop symbol "eos" in the end (see Figure 3(e)). Please note that we use the category '0' to represent the stop symbol. In this way, we can better make some similar characters distinguishable, e.g., '1' and '7' may look similar in some written cases. However, with our stroke encoding, we can denote the character '1' with stroke encoding "Vertical" and '7' with stroke encoding "Horizontal + Vertical", which can tell the SR model a more fine-grained knowledge for reconstruction.

Stroke-Focused Module

Strokes perform a significant role in the recognition process. When we see a low-resolution text image, we usually try to capture stroke-level details to infer the appearance of the whole characters in our brain according to Gestalt Psychology (Köhler 1967). Inspired by this, we try to design a module that can provide stroke-level guidance for the super-resolution model. We observe that the existing recognizers (Shi, Bai, and Yao 2016; Cheng et al. 2017; Shi et al. 2018; Qiao et al. 2020) usually regard characters as the smallest units, i.e., each character corresponds to a unique class in the alphabet. In this context, recognizers can only attend to coarse-grained character regions at each time step. To exploit more fine-grained attention maps, we pre-train a Transformer-based recognizer on two synthetic datasets, including Synth90k (Jaderberg et al. 2016) and SynthText (Gupta, Vedaldi, and Zisserman 2016) with stroke-level labels following (Chen, Li, and Xue 2021b). More specifically, given the character-level labels $c_{\text{GT}} = \{c_1, c_2, ..., c_t\}$, we decompose each character and concatenate them to construct the stroke-level labels $s_{\text{GT}} = \{s_1, s_2, ..., s_{t'}\}$, where t and t' denote the maximum length of labels at two different levels ($t \leq t'$). During pre-training, following (Vaswani et al. 2017; Shi et al. 2018), we use the force teaching strategy to accelerate the training procedure by employing right-shifted stroke-level label $s_{\text{GT}'} = \{s_{<start>}, s_1, s_2, ..., s_{t'-1}\}$ as input, where $s_{<start>}$ denotes the start symbol. We follow the basic design of (Chen, Li, and Xue 2021a) and more details of the encoder and decoder are shown in Supplementary Material. When reaching convergence, we discard the sequence prediction y_{pred} supervised with cross-entropy loss during training, and only leverage the sequence of stroke-level attention maps generated from the Multi-Head Self-Attention Module as stroke-level positional clues. Please note that the parameters in this model are **frozen** after pre-training. Specifically, we denote the attention maps of HR images as $\mathbf{A}_{\text{HR}} = \{\mathbf{A}^1_{\text{HR}}, \mathbf{A}^2_{\text{HR}}, ..., \mathbf{A}^{t'}_{\text{HR}}\}$ and SR images as $\mathbf{A}_{\text{SR}} = \{\mathbf{A}^1_{\text{SR}}, \mathbf{A}^2_{\text{SR}}, ..., \mathbf{A}^{t'}_{\text{SR}}\}$, then employ an L1 loss to constrain these two maps as follows:

$$L_{\text{SFM}} = ||\mathbf{A}_{\text{SR}} - \mathbf{A}_{\text{HR}}||_1 \tag{2}$$



(a) Five basic strokes for Chinese characters (b) Examples of stroke-level decompositon

(c) Nine basic strokes for English characters and digits (d) Examples of stroke-level decompositon (e) Construct stroke-level labels

Figure 3: Decomposition of Chinese characters, English characters, and digits. See more examples in Supplementary Material.

Overall Loss Function

Finally, we construct the overall loss function as follows:

$$L = L_{\text{PSM}} + \lambda_{\text{SFM}} L_{\text{SFM}} \tag{3}$$

where λ_{SFM} balances the weight of these two loss functions.

Experiments

In this section, we first introduce the datasets, some evaluation metrics, and implementation details. Then we discuss the choices of parameters. At last, we demonstrate the experimental results.

Datasets. The datasets used in this paper are as follows:

TextZoom (Wang et al. 2020) The images in TextZoom originate from RealSR (Cai et al. 2019) and SR-RAW (Zhang et al. 2019). These datasets involve LR-HR pairs which are taken by digital cameras in real scenes. Specifically, TextZoom contains $17,367$ LR-HR pairs for training and $4,373$ pairs for testing. In terms of different focal lengths of digital cameras, the test set is divided into three subsets, including $1,619$ LR-HR pairs for the easy subset, $1,411$ LR-HR pairs for the medium subset, and $1,343$ LR-HR pairs for the hard subset. LR images are resized to 16×64 and HR images are sized to 32×128, respectively. Different from handcraft degradation, the LR images in TextZoom suffer from more complicated real-scene degradation, which is more challenging for a certain model to perform the SR text image recovery.

IC15 (Karatzas et al. 2015) contains $1,811$ images originated from natural scenes. It is a challenging benchmark with 352 images with resolution lower than 16×64.

Degraded-IC13 is constructed based on IC13-HCCR (Yin et al. 2013), which contains 224,419 offline handwritten images covering 3,755 commonly-used Level-1 Chinese characters. Details of the construction are shown in the subsection of Experimental on Degraded-IC13.

Evaluation metrics. We remove all the punctuations and convert uppercase letters to lowercase letters for calculating recognition accuracy, which follows the setting of (Wang et al. 2020) for a fair comparison. In addition, we use Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) to evaluate the quality of SR images.

Implementation details. Our model is implemented in PyTorch. All experiments are conducted on one NVIDIA GTX 1080Ti GPU with 11GB memory. The model is trained using Adam (Kingma and Ba 2014) optimizer with learning rate set to 10^{-4}. The batch size is set to 16. After pretrained on Synth90k (Jaderberg et al. 2016) and SynthText (Gupta, Vedaldi, and Zisserman 2016), the parameters of the Transformer-based recognizers are **frozen**. SFM is a pluggable module that is only used when training, i.e., only the PSM is used to upsample LR images in the test phase.

Choices of Parameters

The experiments in this section are all conducted on the TextZoom dataset and we employ CRNN for validation. Specifically, we utilize TSRN as the backbone.

Choices of λ_{SFM}. λ_{SFM} performs an important role to balance the weight of two loss terms. The higher its value, the more our model focuses on the stroke-level details. We explore the value of λ_{SFM} ranging from $\{0, 0.1, 1, 10, 50, 100\}$ and the experimental results are shown in Table 1. When λ_{SFM} is set to 50, the recognition accuracy reaches the best and it boosts the average accuracy by 7.5% compared with the baseline (λ_{SFM}=0). When it values at a lower level such as 0.1, SFM does not bring much guidance for the module. So we set λ_{SFM} to 50 in the following experiments.

Choices of L1 loss and L2 loss. Empirically, L1 loss and L2 loss are interchangeable in super-resolution tasks. To further validate the impact of these two losses on the generated images, we conduct experiments on four combinations of

Table 1: Experiments on the choices of λ_{SFM}.

λ_{SFM}	Easy	Medium	Hard	Average
0	52.5%	38.2%	31.4%	41.4%
0.1	56.5%	40.9%	32.9%	44.2%
1	58.9%	43.7%	34.9%	46.6%
10	58.9%	46.1%	34.4%	47.2%
50	**61.2%**	**47.6%**	**35.5%**	**48.9%**
100	60.6%	47.7%	34.3%	48.4%

Table 2: Experiments on four combinations of two losses.

L_{PSM}	L_{SFM}	Easy	Medium	Hard	Average
L1	L1	59.5%	46.9%	33.9%	47.6%
L1	L2	56.0%	42.8%	33.3%	44.8%
L2	L1	**61.2%**	**47.6%**	**35.5%**	**48.9%**
L2	L2	58.9%	45.6%	34.2%	47.0%

them (see Table 2). The experimental results show that the performance reaches the best when L_{PSM} uses L2 loss and L_{SFM} uses L1 loss. We notice that L_{SFM} is usually a relatively small value with the order of magnitudes near 10^{-3}. Hence, it will produce a much smaller gradient when using L2 loss, which is inefficient to supervise the SR learning.

Experimental Results

In this section, we first conduct the experiments on TextZoom, IC15, and Degraded-IC13. When conducting the experiments on TextZoom and IC15, we test with six recognizers of different categories, including CTC-based CRNN (Shi, Bai, and Yao 2016), rectification-based MORAN (Luo, Jin, and Sun 2019) and ASTER (Shi et al. 2018), Transformer-based NRTR (Sheng, Chen, and Xu 2019), semantics-based SEED (Qiao et al. 2020), as well as NAS-based AutoSTR (Zhang et al. 2020b), all of which are available on GitHub in terms of source code and pre-trained weights. When experimenting on Degraded-IC13, we manually train a recognizer on HWDB1.0-1.1 (Liu et al. 2013).

Experiments on TextZoom. The experimental results are shown in Table 9. We notice that the proposed SFM can indeed provide positive guidance to boost recognition accuracy. When using TBSRN as the backbone and CRNN for evaluation, the model armed with SFM is capable of boosting the accuracy by 5.5% and 0.6% compared with non-focused and character-focused settings. Since the proposed SFM enhances the HR recovery mainly by concentrating on stroke regions, which may result in less fidelity to the background of the original HR image. Thus, the evaluation metrics like SSIM and PSNR are not stably improved in our cases (see Supplementary Material). To determine the experimental evidence in this situation, we further analyze the trend of L_{PSM}. L_{PSM} can drop fastly and converge at a relatively lower degree in the absence of SFM. Based on these observations, we come to the following conclusion: **(1)** PSM usually pays attention to all pixels in given images. Moreover, PSM can perform even better to decrease the super-resolution loss without SFM, thus achieving better scores in terms of two image quality metrics; **(2)** SFM mainly focuses on stroke-level details, which are separate from background pixels. Intuitively, when we set SFM to large weight ($\lambda_{\text{SFM}} = 50$), the model concentrates more on stroke regions while caring less about background pixels, resulting in lower scores on PSNR and SSIM. However, our aim is to recover recognition-friendly and visual-pleasing text images. The visualization and loss analysis also demonstrate that good PSNR and SSIM scores are not equivalent to well-recovered text images. Moreover, one can clearly see in Figure 4 that, the SR model with SFM supervision demonstrates superior SR text image recovery compared with those without SFM. We also explore the ability of SFM when combined with other character-focused methods, e.g., STT (Chen, Li, and Xue 2021a) and TPGSR (Ma, Guo, and Zhang 2021) (See Table 4). We observe that the guidance at character and stroke levels are complementary and the performance can be boosted further when combining text priors at two levels.

Experiments on IC15. IC15 (Karatzas et al. 2015) is one of the widely used English scene text recognition benchmarks. Compared with other datasets such as IC03 (Lucas et al. 2005) and CUTE80 (Risnumawan et al. 2014), this dataset contains more incidentally captured images with low resolution, which is a great challenge for the existing recognizers. We manage to validate the ability of the proposed method as a pre-processor. We extract 352 low-resolution images (i.e., resolution lower than 16×64) from IC15 as a subset named IC15-352 and test on six recognizers. Please note that we do not use the full dataset since the high-resolution image themselves can be well recognized without super-resolution. We follow three settings, including training TSRN without focus, with character-level focus, and with stroke-level focus. The experimental results are shown in Supplementary Material. The model with stroke-level guidance boosts the accuracy of 3.1% compared with the model focusing on the character level when evaluated on CRNN. Moreover, when using the guidance of SFM, the accuracy reaches the best in most cases.

Experiments on Degraded-IC13. Compare to English characters, hieroglyph character like Chinese is structured in more complex shape. However, with stroke prior in SFM, we can also equip the SR model capability to recover such complicated characters. To validate the performance of our method on Chinese characters, we construct the Degraded-IC13 dataset in the following ways: **(1)** We randomly divide IC13-HCCR (Liu et al. 2013) into two subsets. Specifically, 179,535 images (80%) are chosen for training and 44,884 images (20%) for testing. We first resize them to 64×64; **(2)** For each image, we randomly select n from 1,2,3,4,5 as the number of blurred operations; **(3)** We blur the original images for n times. For each time, the blurred type is randomly chosen from four choices, which are demonstrated in Figure 5; **(4)** We resize the blurred image to 32×32 as LR images using bicubic interpolation. Several examples of the generated HR-LR pairs are demonstrated in Figure 5(b). Following the setting of experiments on English datasets, we pre-train a Chinese recognizer for evaluation and a Transformer-based recognizer that provides stroke-level guidance on the HWDB1.0-1.1 dataset (Liu et al. 2013). The experimental

Table 3: The experimental results on TextZoom (The results of NRTR, SEED, and AutoSTR are in Supplementary Material). The module can generate more recognizable text images with the guidance of SFM. The *underlined* numbers indicate the best average accuracy using the specific backbone and recognizer for evaluation. The **bold** numbers denote the best accuracy.

Backbone	Focus	CRNN (Shi, Bai, and Yao 2016)				MORAN (Luo, Jin, and Sun 2019)				ASTER (Shi et al. 2018)			
		Easy	Medium	Hard	Average	Easy	Medium	Hard	Average	Easy	Medium	Hard	Average
LR	-	36.4%	21.1%	21.1%	26.8%	60.6%	37.9%	30.8%	44.1%	67.4%	42.4%	31.2%	48.2%
HR	-	76.4%	75.1%	64.6%	72.4%	91.2%	85.3%	74.2%	84.1%	94.2%	87.7%	76.2%	86.6%
SRCNN	None	41.1%	22.3%	22.0%	29.2%	63.9%	40.0%	29.4%	45.6%	70.6%	44.0%	31.5%	50.0%
	Char	41.7%	25.4%	23.1%	30.7%	66.2%	44.4%	31.3%	48.4%	70.2%	49.4%	32.5%	51.9%
	Stroke	46.5%	30.8%	25.2%	34.9%	65.2%	46.4%	32.2%	49.0%	68.8%	47.7%	33.1%	51.0%
SRResNet	None	45.2%	32.6%	25.5%	35.1%	66.0%	47.1%	33.4%	49.9%	69.4%	50.5%	35.7%	53.0%
	Char	50.0%	36.2%	28.4%	38.9%	70.4%	53.9%	37.9%	55.1%	72.6%	57.1%	38.7%	57.2%
	Stroke	55.5%	42.5%	31.2%	43.8%	72.9%	54.1%	36.8%	55.7%	74.7%	56.2%	38.3%	57.6%
TBSRN	None	54.2%	40.6%	32.7%	43.2%	71.1%	55.2%	39.5%	56.3%	75.2%	56.8%	40.2%	58.5%
	Char	59.6%	47.1%	35.3%	48.1%	74.1%	57.0%	40.8%	58.4%	75.7%	59.9%	41.6%	60.1%
	Stroke	**61.3%**	47.2%	35.0%	48.7%	73.6%	57.7%	40.3%	58.2%	77.4%	59.0%	41.3%	60.4%
TSRN	None	52.5%	38.2%	31.4%	41.4%	70.1%	55.3%	37.9%	55.4%	75.1%	56.3%	40.1%	58.3%
	Char	54.3%	40.4%	31.7%	42.9%	72.3%	55.6%	39.8%	56.9%	74.3%	59.7%	39.6%	58.9%
	Stroke	61.2%	**47.6%**	**35.5%**	**48.9%**	**75.8%**	**57.8%**	**41.4%**	**59.4%**	**77.9%**	**60.2%**	**42.4%**	**61.3%**

Table 4: Results of combining TPGSR and STT with SFM. We use TSRN as the backbone and CRNN for evaluation.

Model	SFM	Easy	Medium	Hard	Average
TPGSR	-	63.1%	52.0%	38.6%	51.8%
	✓	**64.2%**	**53.2%**	**38.9%**	**52.9%**
STT	-	61.2%	47.6%	**35.5%**	48.9%
	✓	**62.3%**	**48.1%**	35.2%	**49.4%**

Table 5: Experimental results on the necessity of the preprocessor. "TZ" denotes the training set of TextZoom.

Setting	Easy	Medium	Hard	Average
(1) Baseline	45.3%	25.4%	18.5%	30.6%
(2) Blur-Aug	53.3%	32.7%	22.7%	37.3%
(3) Train w/ TZ	52.1%	33.1%	22.5%	36.9%
(4) Fine-tune w/ TZ	**55.4%**	35.9%	23.6%	39.3%
(5) Preprocessor	54.8%	**40.1%**	**28.1%**	**41.9%**

results for the *none-focused*, *character-focused*, and *stroke-focused* are 83.4%, 84.6%, 86.1%, respectively. We notice that the model with SFM can boost the accuracy by almost 3% when focusing on stroke regions compared with the non-focused setting. Several examples are shown in Figure 6. Through the visualizations, we observe that the images generated with the guidance of SFM have relatively clearer strokes, thus achieving better accuracy on the recognizer.

Discussions

Deep insight in pre-trained stroke-level recognizer. To provide stroke-level attention maps, the pre-trained recognizer should employ stroke-level text labels unfolded by character-level labels. Before unfolding, the average length of character-level text labels in the training set of TextZoom is 5.0 and the average length of stroke-level text labels reaches 10.9. In fact, attention-based recognizers are easier to suffer from the attention drift problem (Cheng et al. 2017) when predicting longer sequences. In addition, we notice that after pre-trained on two Synthetic datasets, the recognizer can only achieve 78.0% recognition accuracy on the training set of TextZoom. Specifically, wrong predictions are usually accompanied by drifted attention maps, which may provide noise for the super-resolution model. To deeply analyze the effect of noise on the SR model, we experiment in three settings: **(1)** Use attention maps only with **Correct** predictions. **(2)** Use **All** attention maps. **(3)** Use attention maps only with **Wrong** predictions. We employ TSRN as the backbone and CRNN for validation. The experimental results are shown in Table 6. Interestingly, we observe that the average accuracy of settings (1) and (2) do not show many differences (48.5% *v.s.* 48.9%). Based on the result of setting (3), we notice that the performance drops drastically with the wrong guidance. Hence, we come to the conclusion that the stroke-level guidance indeed boosts the performance and the model is robust to resist some disturbances.

Table 6: Experiments on the effect of noise.

Setting	CRNN			
	Easy	Medium	Hard	Average
(1) Correct	**61.4%**	**47.2%**	34.4%	48.5%
(2) All	61.2%	**47.2%**	**35.5%**	**48.9%**
(3) Wrong	52.5%	35.9%	28.6%	39.8%

Can pre-processor be replaced by training strategies? As is mentioned before, the proposed method can indeed boost the recognition performance of existing recognizers on either TextZoom or IC15 datasets. However, here comes a question: *What if the recognizers for evaluation are better trained to adapt to low-quality TextZoom test sets?* To answer this question, we retrain CRNN with Synth90k (Jaderberg et al. 2016) and SynthText (Gupta, Vedaldi, and Zisserman 2016) as the baseline *(Setting 1)*, and utilize some training strategies, including randomly blur synthesize images for data augmentation *(Setting 2)*, combine the HR-LR pairs in the training set of TextZoom with two synthesize

Figure 4: Examples of the generated images. "None" means no text priors are taken into account, while "Char" and "Stroke" denote the model is trained with character-level guidance and stroke-level guidance. We choose TSRN as the backbone.

Figure 5: Manually design LR-HR pairs to construct Degraded-IC13. The upper row are four types of blur and the lower row are some examples of LR-HR pairs.

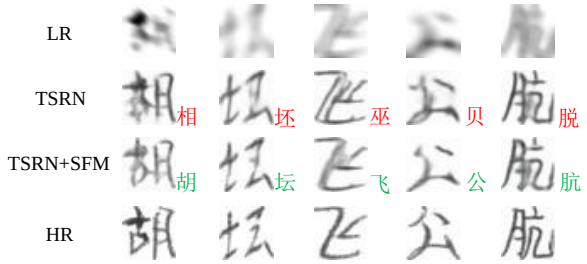


Figure 6: Examples of the generated Chinese characters.

datasets for training *(Setting 3)*, fine-tune CRNN with HR-LR pairs in the training set of TextZoom *(Setting 4)*. As is shown in Table 5, we observe that the performance reaches the best when the super-resolution model is used as the pre-processor *(Setting 5)*. The reasons may be two folds: **(1)** The downsampling strategy by algorithms can not simulate the situation in the real scene. **(2)** By training an additional pre-processor, the model can perceive the degradation process of text images, so it can better generalize to the test dataset.

Can the SR model be extended to other languages? We have also conducted experiments on the Korean character dataset PE92 (KIM et al. 1996) following the same settings for tackling Chinese characters. The stroke-level decomposition of Korean characters is available in the publicly available code. The accuracy of the Korean recognizer is 90.74% (none-focused), 90.32% (character-focused), 92.37% (stroke-focused), respectively. It further validates the superiority of our method in other languages.

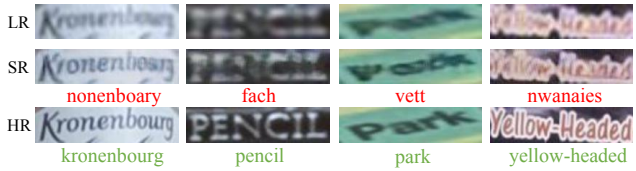


Figure 7: Visualization of some failure cases.

Table 7: Parameters and FLOPs for two backbones.

Backbone	SFM	Parameters	FLOPs
SRResNet	-	2.5M	0.7G
	✓	2.5M + 62.0M	0.7G + 13.6G
TSRN	-	2.8M	0.9G
	✓	2.8M + 62.0M	0.9G + 13.6G

Computational cost. In the test phase, we evaluate the time efficiency of our super-resolution method using TSRN as the backbone. We conduct the experment using one NVIDIA GTX 1080TI GPU. To run a batch, the model takes 0.16 seconds without SFM and 0.57 seconds with SFM. The details about parameters and FLOPs are shown in Figure 7. In particular, SFM does not bring any time overhead during testing since it is only used in the training phase to provide stroke-level positional clues.

Failure cases. Some failure cases are demonstrated in Figure 7. We observe that our super-resolution method are weak to tackle images with long text since the stroke details are not clear i.e., mix with adjacent strokes. Additionally, the oblique text images and images with uneven illumination also bring difficulties to our methods. We will try to mitigate these problems in our future work.

Conclusion

In this paper, we propose a Stroke-Aware Scene Text Image Super-Resolution method inspired by Gestalt Psychology, highlighting the details on stroke regions. The proposed method can indeed generate more distinguishable super-resolution text images. As is demonstrated in the experimental results, the proposed SFM is capable of achieving state-of-the-art performance on TextZoom and Chinese handwritten datasets without introducing additional time overhead.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (No.62176060), STCSM project (No.20511100400), Shanghai Municipal Science and Technology Major Project (No.2018SHZDZX01) and ZJLab, Shanghai Research and Innovation Functional Program (No.17DZ2260900), and the Program for Professor of Special Appointment (Eastern Scholar) at Shanghai Institutions of Higher Learning.

References

Cai, J.; Zeng, H.; Yong, H.; Cao, Z.; and Zhang, L. 2019. Toward real-world single image super-resolution: A new benchmark and a new model. In *ICCV*, 3086–3095.

Capel, D.; and Zisserman, A. 2000. Super-resolution enhancement of text image sequences. In *ICPR*, volume 1, 600–605.

Chen, J.; Li, B.; and Xue, X. 2021a. Scene Text Telescope: Text-Focused Scene Image Super-Resolution. In *CVPR*, 12026–12035.

Chen, J.; Li, B.; and Xue, X. 2021b. Zero-Shot Chinese Character Recognition with Stroke-Level Decomposition. In *IJCAI*.

Chen, X.; Jin, L.; Zhu, Y.; Luo, C.; and Wang, T. 2021. Text recognition in the wild: A survey. *ACM Computing Surveys (CSUR)*, 54(2): 1–35.

Cheng, Z.; Bai, F.; Xu, Y.; Zheng, G.; Pu, S.; and Zhou, S. 2017. Focusing attention: Towards accurate text recognition in natural images. In *ICCV*, 5076–5084.

Dalley, G.; Freeman, B.; and Marks, J. 2004. Single-frame text super-resolution: A bayesian approach. In *ICIP*, volume 5, 3295–3298.

Dong, C.; Loy, C. C.; He, K.; and Tang, X. 2014. Learning a deep convolutional network for image super-resolution. In *ECCV*, 184–199.

Graves, A.; Fernández, S.; Gomez, F.; and Schmidhuber, J. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *ICML*, 369–376.

Gupta, A.; Vedaldi, A.; and Zisserman, A. 2016. Synthetic data for text localisation in natural images. In *CVPR*, 2315–2324.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*, 770–778.

Jaderberg, M.; Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2016. Reading text in the wild with convolutional neural networks. *IJCV*, 116(1): 1–20.

Jaderberg, M.; Simonyan, K.; Zisserman, A.; et al. 2015. Spatial transformer networks. In *NeurIPS*, 2017–2025.

Karatzas, D.; Gomez-Bigorda, L.; Nicolaou, A.; Ghosh, S.; Bagdanov, A.; Iwamura, M.; Matas, J.; Neumann, L.; Chandrasekhar, V. R.; Lu, S.; et al. 2015. ICDAR 2015 competition on robust reading. In *ICDAR*, 1156–1160.

KIM, D.-H.; Hwang, Y.-S.; Park, S.-T.; Kim, E.-J.; Paek, S.-H.; and BANG, S.-Y. 1996. Handwritten Korean character image database PE92. *IEICE transactions on information and systems*, 79(7): 943–950.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Köhler, W. 1967. Gestalt psychology. *Psychologische Forschung*, 31(1): XVIII–XXX.

Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. 2017. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, 4681–4690.

Li, J.; Fang, F.; Mei, K.; and Zhang, G. 2018. Multi-scale residual network for image super-resolution. In *ECCV*, 517–532.

Lim, B.; Son, S.; Kim, H.; Nah, S.; and Mu Lee, K. 2017. Enhanced deep residual networks for single image super-resolution. In *CVPR*, 136–144.

Liu, C.-L.; Yin, F.; Wang, D.-H.; and Wang, Q.-F. 2013. Online and offline handwritten Chinese character recognition: benchmarking on new databases. *PR*, 46(1): 155–162.

Lucas, S. M.; Panaretos, A.; Sosa, L.; Tang, A.; Wong, S.; Young, R.; Ashida, K.; Nagai, H.; Okamoto, M.; Yamamoto, H.; et al. 2005. ICDAR 2003 robust reading competitions: entries, results, and future directions. *IJDAR*, 7(2-3): 105–122.

Luo, C.; Jin, L.; and Sun, Z. 2019. Moran: A multi-object rectified attention network for scene text recognition. *PR*, 90: 109–118.

Ma, J.; Guo, S.; and Zhang, L. 2021. Text Prior Guided Scene Text Image Super-resolution. *arXiv preprint arXiv:2106.15368*.

Neumann, L.; and Matas, J. 2012. Real-time scene text localization and recognition. In *CVPR*, 3538–3545.

Pandey, R. K.; Vignesh, K.; Ramakrishnan, A.; et al. 2018. Binary document image super resolution for improved readability and OCR performance. *arXiv preprint arXiv:1812.02475*.

Phan, T. Q.; Shivakumara, P.; Tian, S.; and Tan, C. L. 2013. Recognizing text with perspective distortion in natural scenes. In *Proceedings of the IEEE International Conference on Computer Vision*, 569–576.

Qiao, Z.; Zhou, Y.; Yang, D.; Zhou, Y.; and Wang, W. 2020. Seed: Semantics enhanced encoder-decoder framework for scene text recognition. In *CVPR*, 13528–13537.

Ren, H.; Pan, M.; Li, Y.; Zhou, X.; and Luo, J. 2020. ST-SiameseNet: Spatio-Temporal Siamese Networks for Human Mobility Signature Identification. In *KDD*, 1306–1315.

Risnumawan, A.; Shivakumara, P.; Chan, C. S.; and Tan, C. L. 2014. A robust arbitrary text detection system for natural scene images. *ESA*, 41(18): 8027–8048.

Satyawan, W.; Pratama, M. O.; Jannati, R.; Muhammad, G.; Fajar, B.; Hamzah, H.; Fikri, R.; and Kristian, K. 2019. Citizen Id Card Detection using Image Processing and Optical Character Recognition. In *JPCS*, volume 1235, 012049.

Sheng, F.; Chen, Z.; and Xu, B. 2019. NRTR: A no-recurrence sequence-to-sequence model for scene text recognition. In *ICDAR*, 781–786.

Shi, B.; Bai, X.; and Yao, C. 2016. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *TPAMI*, 39(11): 2298–2304.

Shi, B.; Yang, M.; Wang, X.; Lyu, P.; Yao, C.; and Bai, X. 2018. Aster: An attentional scene text recognizer with flexible rectification. *TPAMI*, 41(9): 2035–2048.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *NeurIPS*, 5998–6008.

Wang, K.; Babenko, B.; and Belongie, S. 2011. End-to-end scene text recognition. In *ICCV*, 1457–1464.

Wang, K.; and Belongie, S. 2010. Word spotting in the wild. In *ECCV*, 591–604.

Wang, W.; Xie, E.; Liu, X.; Wang, W.; Liang, D.; Shen, C.; and Bai, X. 2020. Scene text image super-resolution in the wild. In *ECCV*, 650–666.

Xu, X.; Sun, D.; Pan, J.; Zhang, Y.; Pfister, H.; and Yang, M.-H. 2017. Learning to super-resolve blurry face and text images. In *ICCV*, 251–260.

Yan, R.; and Huang, Y. 2020. PlugNet: Degradation Aware Scene Text Recognition Supervised by a Pluggable Super-Resolution Unit. In *ECCV*.

Yang, L.; Wang, P.; Li, H.; Li, Z.; and Zhang, Y. 2020. A holistic representation guided attention network for scene text recognition. *Neurocomputing*, 414: 67–75.

Yin, F.; Wang, Q.-F.; Zhang, X.-Y.; and Liu, C.-L. 2013. ICDAR 2013 Chinese handwriting recognition competition. In *ICDAR*, 1464–1470.

Zeiler, M. D. 2012. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.

Zhang, C.; Ding, W.; Peng, G.; Fu, F.; and Wang, W. 2020a. Street View Text Recognition With Deep Learning for Urban Scene Understanding in Intelligent Transportation Systems. *TITS*.

Zhang, H.; Yao, Q.; Yang, M.; Xu, Y.; and Bai, X. 2020b. Efficient backbone search for scene text recognition. In *ECCV*.

Zhang, X.; Chen, Q.; Ng, R.; and Koltun, V. 2019. Zoom to learn, learn to zoom. In *CVPR*, 3762–3770.