Orientation-Independent Chinese Text Recognition in Scene Images

Haiyang Yu, Xiaocong Wang, Bin Li*, Xiangyang Xue*
Shanghai Key Laboratory of Intelligent Information Processing
School of Computer Science, Fudan University
{hyyu20, xcwang20, libin, xyxue}@fudan.edu.cn

Abstract

Scene text recognition (STR) has attracted much attention due to its broad applications. The previous works pay more attention to dealing with the recognition of Latin text images with complex backgrounds by introducing language models or other auxiliary networks. Different from Latin texts, many vertical Chinese texts exist in natural scenes, which brings difficulties to current state-of-the-art STR methods. In this paper, we take the first attempt to extract orientation-independent visual features by disentangling content and orientation information of text images, thus recognizing both horizontal and vertical texts robustly in natural scenes. Specifically, we introduce a Character Image Reconstruction Network (CIRN) to recover corresponding printed character images with disentangled content and orientation information. We conduct experiments on a scene dataset for benchmarking Chinese text recognition, and the results demonstrate that the proposed method can indeed improve performance through disentangling content and orientation information. To further validate the effectiveness of our method, we additionally collect a Vertical Chinese Text Recognition (VCTR) dataset. The experimental results show that the proposed method achieves 45.63% improvement on VCTR when introducing CIRN to the baseline model.

1 Introduction

Scene text recognition (STR) has received much attention in the field of computer vision due to its broad range of applications, such as traffic sign recognition [Sermanet and LeCun, 2011] and text image retrieval [Wang *et al.*, 2019]. It aims to transcribe texts from natural images into sequences of digital characters. Reading texts from natural images faces many difficulties, *e.g.*, text distortion, partial occlusion, and complex backgrounds. Different from Latin text recognition, Chinese text recognition poses additional challenges, such as commonly-seen vertical texts and complicated sequential patterns [Chen *et al.*, 2021c]. These unique features make Chinese text recognition a challenging task.

(a) English texts in scene images

(b) Chinese texts in scene images

Figure 1: In street-view images, most English texts are horizontal; in contrast, vertical Chinese texts are commonly-seen as well.

Compared with Latin texts, Chinese texts are more likely to appear in the vertical orientation due to the commonly-used traditional couplets or signboards in natural scenes (as shown in Figure 1(b)). On the contrary, there are few vertical Latin texts due to different inherent reading habits (as shown in Figure 1(a)). Most of the early methods [Yin *et al.*, 2017; Liu *et al.*, 2018; Shi *et al.*, 2016] are specially designed for Latin text recognition, and limited to horizontal texts. Thus, they can hardly handle text instances with various shapes such as curved and vertical texts, leading to a serious impact on recognizing Chinese texts in scene images. To tackle curved texts, some methods [Shi *et al.*, 2018; Li *et al.*, 2019] introduce a rectification network [Jaderberg *et al.*, 2015] to straighten irregular text instances or rely on the 2D attention mechanism to locate each character. In addition, researchers have tried to introduce linguistic knowledge and corpora to improve the performance on curved texts [Fang *et al.*, 2021; Yu *et al.*, 2020]. However, these methods are still inefficient for vertical text recognition since the layout of vertical texts is completely different from horizontal or curved texts. Some

*Corresponding author

Chinese character recognition methods [Wu *et al.*, 2019] have attempted to improve the robustness of models for rotated characters, but they cannot be directly applied to text line recognition. On the whole, existing scene text recognition methods still have difficulties in dealing with vertical Chinese texts. Thus, developing a network to learn visual features that are independent of text orientation is crucial for recognizing vertical Chinese texts.

We observe that visual features contain not only the content information that determines character predictions but also text orientation information. Therefore, in this paper, we try to disentangle the content and orientation information from the visual features to obtain orientation-independent features for accurate recognition of vertical Chinese texts. The proposed method consists of a customized ResNet [He *et al.*, 2016] encoder, a transformer-based decoder [Vaswani *et al.*, 2017], and a character image reconstruction network. By making modifications to ResNet, the encoder captures more details and preserves more visual features. The character image reconstruction network contains a content information extractor, an orientation information extractor, and a reconstruction module. The content information extractor is used to obtain content information from visual features, and the orientation information extractor disentangles orientation information. We decouple the content and orientation information of horizontal and rotated vertical characters and exchange their orientation information to reconstruct corresponding printed character images. Finally, we use a transformer-based decoder to capture the semantic dependencies between characters to generate final predictions.

To benchmark the performance of existing state-of-the-art methods in vertical Chinese text recognition, we collect a vertical Chinese text recognition (VCTR) dataset from Poster-Erase [Jiang *et al.*, 2022]. The experimental results show that our method outperforms existing STR models by a large margin on VCTR. In addition, we achieve better results on a generic Chinese text recognition dataset. The code of our method and VCTR dataset are available at GitHub[1]. The contributions of this paper can be summarized as follows:

- We collect a Vertical Chinese Text Recognition (VCTR) dataset to benchmark the performance of vertical Chinese text recognition since vertical texts are the key issue affecting Chinese scene text recognition.
- We take the first attempt to disentangle the content and orientation information from visual features with a character image reconstruction network, which can eliminate the disturbance of text orientation.
- The proposed method significantly outperforms the existing methods on vertical Chinese text recognition and also achieves new state-of-the-art results on a Chinese scene text recognition dataset.

2 Related Work

Scene text recognition (STR) has been a long-standing research topic in computer vision. Early works in this field focus on utilizing low-level features such as histograms of oriented gradient descriptors [Wang *et al.*, 2011], connected components [Neumann and Matas, 2012], and so on. With the rapid development of deep learning, STR research has made significant progress in the last few years. Based on their linguistic categories, we divide them into two categories: Latin text recognition and Chinese text recognition.

2.1 Latin Text Recognition

Latin scene text recognition can be divided into two categories: regular and irregular text recognition. The sequence-to-sequence models based on the CTC loss [Graves *et al.*, 2006; Shi *et al.*, 2016] and attention mechanism [Cheng *et al.*, 2017] have made great progress in regular text recognition. However, these methods struggle to handle curved or rotated texts. For irregular texts, previous methods [Zhan and Lu, 2019; Yang *et al.*, 2019; Shi *et al.*, 2018] tend to integrate a spatial transformer module into an attention-based framework to rectify the curved text images to the horizontal form, but the predefined transformation space limits their generalization capabilities. The segmentation-based methods [Liao *et al.*, 2019; Wan *et al.*, 2020] first detect characters and then integrate characters into text predictions. Some recently proposed approaches attempt to use linguistic rules to aid the recognition process, showing strong performance on irregular text recognition. For example, ABINet [Fang *et al.*, 2021] and VisionLAN [Wang *et al.*, 2021] develop a specific module to integrate language information into text recognition. The aforementioned approaches are all specially designed for Latin text recognition, and cannot work well when facing Chinese text recognition due to the large alphabet and commonly-seen vertical texts.

2.2 Chinese Text Recognition

Due to complex inner structures of Chinese characters, some methods [Yu *et al.*, 2022; Zu *et al.*, 2022] are proposed to recognize Chinese characters. DenseRAN [Wang *et al.*, 2018] treats a Chinese character as a composition of two-dimensional structures and radicals. Based on DenseRAN, STN-DenseRAN [Wu *et al.*, 2019] further employs a rectification block to handle distorted character images. HDE [Cao *et al.*, 2020] designs a unique embedding vector for each Chinese character according to its radical-level constitution. In [Chen *et al.*, 2021b], characters are decomposed into a combination of five strokes in Chinese, and the predicted stroke sequence is transformed to a specific character through a matching-based strategy. Recently, some works [Chen *et al.*, 2021c; Su *et al.*, 2023] focus on Chinese text recognition (CTR). For instance, the authors of [Chen *et al.*, 2021c] proposed a benchmark for CTR and introduced radical-level supervision to improve the performance of text recognition models on CTR. SVTR [Du *et al.*, 2022] proposes a transformer-based framework, utilizing global mixing and local mixing to perceive the inter-character and intra-character patterns, respectively. It performs well on the Chinese scene dataset. However, these approaches mainly focus on Chinese character or horizontal text recognition, while ignoring the commonly-seen vertical texts.

[1]https://github.com/FudanVI/FudanOCR/orientation-independent-CTR

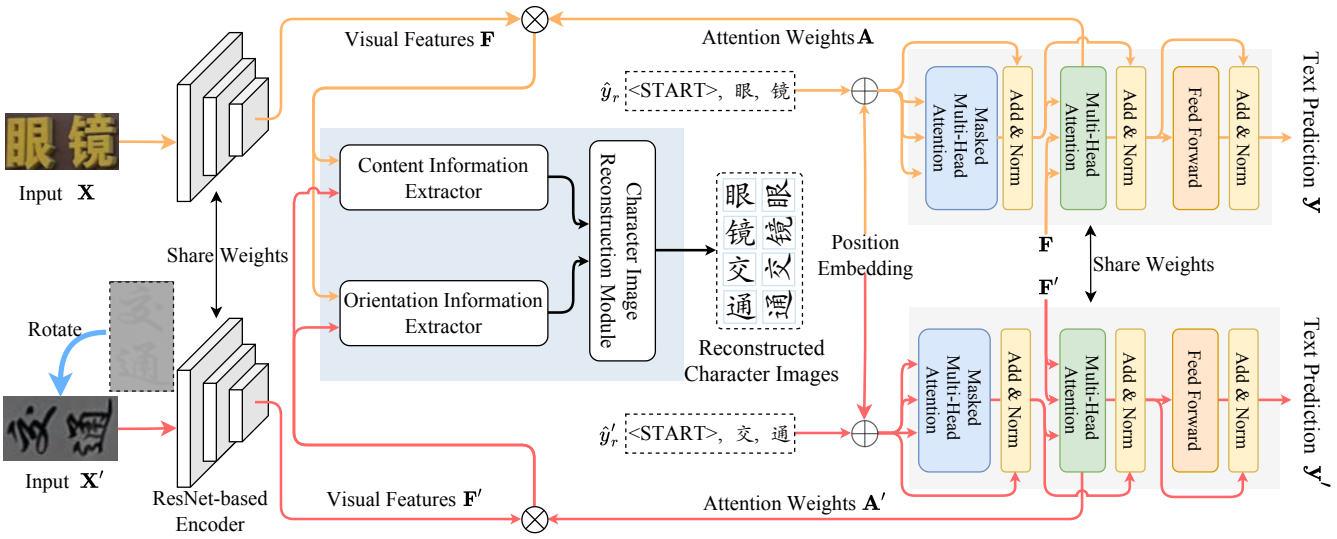

Figure 2: The overall architecture of the proposed method. It consists of a ResNet-based encoder, a transformer-based decoder, and a character image reconstruction network. The data flow of horizontal and vertical text images is in orange and red lines, respectively. The blue box represents the character image reconstruction network; the grey boxes represent the transformer-based decoder.

3 Methodology

In this section, we first review the commonly-used encoder-decoder framework in scene text recognition. Then, we analyze the information contained in extracted visual features by the encoder. Finally, we introduce the details of each module in the proposed architecture.

3.1 Generic Framework

In the past few years, researchers tend to adopt the encoder-decoder framework to solve the text recognition task. Generally, a ResNet-based backbone [He *et al.*, 2016] is employed as the encoder to extract visual features $\mathbf{F}$. Subsequently, the features $\mathbf{F}$ are fed into designed decoders, such as the attention-based decoder [Li *et al.*, 2019] and the transformer-based decoder [Vaswani *et al.*, 2017]. Both of these decoders are composed of two modules: the attention module and the prediction module. At the t-th time step, the attention module calculates the glimpse vector $\mathbf{g}_t$ as follows:

$$\mathbf{g}_t = \sum_{ij} \alpha_{ij}^t \mathbf{f}_{ij} \tag{1}$$

where $\mathbf{f}_{ij}$ represents the feature vector at the position (i, j) of $\mathbf{F}$ and α_{ij}^t is the attention weight of $\mathbf{f}_{ij}$ at the t-th time step. Finally, the glimpse vector $\mathbf{g}_t$ is taken as the input of the prediction module to predict the corresponding character or the end token $\langle$EOS$\rangle$:

$$\mathbf{y}_t = \text{Softmax}(\mathbf{W}\mathbf{g}_t + b) \tag{2}$$

where $\mathbf{W}$ and b represent the linear transformation and the bias of the prediction module, respectively.

3.2 Visual Features Dissection

The existing methods, which are mostly designed for tackling Latin text recognition, rarely consider the problem of vertical text recognition. In contrast, vertical Chinese texts are commonly-seen in natural scenes, as shown in Figure 1. To recognize vertical and horizontal texts simultaneously, some researchers [Li *et al.*, 2019] proposed to rotate those images with height larger than width by 90 degrees anticlockwise and then feed them into recognition models. The experimental results demonstrate that this strategy can indeed improve the performance on vertical text images. However, this strategy will make recognizers confused to a certain degree since recognizers are forced to classify completely different visual features (*i.e.*, the visual features of horizontal and rotated vertical characters) into the same character. Through experiments, we observe that when a small proportion of vertical text images are included in the training set, the performance of recognizers will descend when this strategy is used at the training stage.

Based on the above observation, we speculate that the extracted visual features contain not only content information but also orientation information of characters. To verify the conjecture about orientation information, we calculate the similarity S_o between the visual features of two different text images with the same orientation and the similarity S_c between two text images with different orientations but with the same characters. We observe that S_o tends to be larger than S_c, implying that the visual features indeed contain the orientation information. Therefore, in this paper, we try to develop a character image reconstruction network to disentangle the orientation-independent content information from the visual features, which will make the recognizer more robust to vertical text images.

3.3 Proposed Method

Overview. As depicted above, we observe that the extracted visual features contain not only the content information, which determines the character prediction, but also the orientation information, which is useless for final predictions.

Thus, we attempt to disentangle the content and orientation information from the extracted visual features. As shown in Figure 2, we take TransOCR [Chen *et al.*, 2021c] as the baseline model, which consists of a ResNet-based encoder and a transformer-based decoder. In this paper, we additionally develop a Character Image Reconstruction Network (CIRN). In the following, we introduce the details of our method.

ResNet-based Encoder. Given the input text image $\mathbf{X}$, the ResNet-based encoder is employed to extract its visual features $\mathbf{F} \in \mathbb{R}^{\frac{H}{8} \times \frac{W}{8} \times C}$ (H and W represent the height and width of the input image $\mathbf{X}$, respectively). We adopt ResNet-34 as the main body of the encoder and modify some layers in the original ResNet-34. First, we replace the 7×7 kernel of the first convolution layer with a 3×3 kernel since the smaller kernel can capture more local details for recognizing text images. In addition, we drop the last convolution block to reduce the number of parameters in the encoder and improve the efficiency of feature extraction. Finally, we remove the max pooling layer of the third convolution block in ResNet-34 to reserve more visual features for the subsequent decoder. Although the max pooling layer can make the model have rotation invariance to some extent [Laptev *et al.*, 2016], the loss of visual information outweighs the gains of rotation invariance.

Transformer-based Decoder. As shown in Figure 2, the transformer-based decoder consists of three modules: the masked multi-head attention module, the multi-head attention module and the feed forward module. The masked multi-head attention module, taking the right-shifted ground truth $\hat{y}_r$ as input, is to capture the semantic dependence between characters. Then, the multi-head attention module calculates the attention weights $\mathbf{A}$ between the extracted visual features $\mathbf{F}$ and $\hat{y}_r$. Finally, the weighted features are input to the feed forward module to extract deeper features, which is used to generate the final predictions $\mathbf{y}$ through a linear layer.

Character Image Reconstruction Network. To disentangle the content information from $\mathbf{F}$ and avoid the disturbance of orientation information, we propose a Character Image Reconstruction Network (CIRN). Specifically, we first obtain the visual features $\mathbf{F}_c$ of each character in the text image by a position-wise multiplication between $\mathbf{F}$ extracted by the encoder and the attention weights $\mathbf{A}$ from the transformer-based decoder. Then, $\mathbf{F}_c$ is fed into CIRN to disentangle its content and orientation information.

The proposed CIRN contains a content information extractor, an orientation information extractor and a reconstruction module. The content information extractor simply adopts a 1×1 convolution layer rather than other complex structures (more analysis is shown in Section 5). The orientation information extractor additionally employs a global average pooling layer after a 1×1 convolution layer to extract the orientation information since the orientation information should be obtained from a global perspective. Among the sampled data in a training batch, there are horizontal and vertical text images. Assuming that the orientation and content information of a horizontal character a and a rotated vertical character b are denoted as O_a, O_b, C_a, and C_b, we exchange their orientation information to reconstruct corresponding printed character images. Formally, four character representations can be obtained as follows:

$$H_a = \text{Fuse}(C_a, O_a), V_a = \text{Fuse}(C_a, O_b) \tag{3}$$

$$H_b = \text{Fuse}(C_b, O_a), V_b = \text{Fuse}(C_b, O_b) \tag{4}$$

where "Fuse" indicates that we concatenate the vector of orientation information to each position of the content information. Then, four character representations are fed into the reconstruction module to generate corresponding printed horizontal character images H and rotated vertical character images V. Specifically, the reconstruction module simply consists of 5 layers of deconvolution with the kernel size of 5 and the stride of 2.

Loss Functions. Four loss functions are introduced to supervise the proposed method:

1) Text prediction loss $\mathcal{L}_t$, supervising the prediction of texts in input images, is calculated by:

$$\mathcal{L}_t = \text{CE}(\mathbf{y}, \hat{\mathbf{y}}) \tag{5}$$

where "CE" represents the cross entropy loss function; $\hat{\mathbf{y}}$ denotes the ground truth of predicted texts.

2) Orientation classification loss $\mathcal{L}_o$ is used to supervise the prediction of character orientation, which can be regarded as a binary classification problem and be computed as follows:

$$\mathcal{L}_o = \text{CE}(O, \hat{O}) \tag{6}$$

where O and $\hat{O}$ denote the orientation prediction of characters and corresponding ground truth respectively. $\hat{O}$ can be obtained by comparing input images' height and width.

3) Content classification loss $\mathcal{L}_c$, similar to $\mathcal{L}_o$, is calculated by:

$$\mathcal{L}_c = \text{CE}(C, \hat{C}) \tag{7}$$

where C represents the character predictions through the content information, and $\hat{C}$ denotes corresponding ground truth.

4) Character image reconstruction loss $\mathcal{L}_r$ is employed to constrain the reconstruction of horizontal and rotated vertical character images, which can be formulated as:

$$\mathcal{L}_r = \text{MSE}(H, \hat{H}) + \text{MSE}(V, \hat{V}) \tag{8}$$

where $\hat{H}$ and $\hat{V}$ represent the printed character images in font Simsun corresponding to H and V.

Therefore, the overall loss function $\mathcal{L}$ of the proposed method is the weighted sum of the above four loss functions:

$$\mathcal{L} = \mathcal{L}_t + \alpha\mathcal{L}_o + \beta\mathcal{L}_c + \gamma\mathcal{L}_r \tag{9}$$

where α, β, and γ are hyperparameters to balance these four loss functions.

4 Experiments

In this section, we first introduce the details of the adopted scene dataset in [Chen *et al.*, 2021c] and the collected Vertical Chinese Text Recognition (VCTR) dataset. Then, we introduce the utilized evaluation methods and the implementation details of our method. Finally, we present the results of ablation studies and experiments.

(a) Horizontal samples of the Scene dataset

(b) Vertical samples of the Scene dataset

Figure 3: Some examples of horizontal and vertical text images in the scene dataset.

(a) Some examples of VCTR

(b) Filtered unrecognizable examples

Figure 4: Some examples of VCTR are shown on the left. Only vertical texts are reserved for VCTR. Some unrecognizable examples (*e.g.*, multi-line and oblique texts) that are removed during collecting VCTR are shown on the right.

4.1 Datasets

Scene dataset. The scene dataset is collected by [Chen *et al.*, 2021c] and derives from six existing datasets, including RCTW [Shi *et al.*, 2017], ReCTS [Zhang *et al.*, 2019], LSVT [Sun *et al.*, 2019], ArT [Chng *et al.*, 2019], and CTW [Yuan *et al.*, 2019]. This dataset contains 509,164 samples for training, 63,645 for validation, and 63,646 for test. Some examples of this dataset are shown in Figure 3.

VCTR. To validate the effectiveness of our method in tackling vertical text images, we collect a Vertical Chinese Text Recognition (VCTR) dataset from PosterErase [Jiang *et al.*, 2022], which is originally proposed for the scene text erasing task. PosterErase includes 58,114 training samples, 148 validation samples and 146 test samples. We only collect vertical text images from the training set of PosterErase since its annotations contain the orientation information of cropped text areas. We obtain the VCTR dataset through the following steps: 1) Filter out the cropped text areas annotated as horizontal ones and reserve the remaining vertical text areas; 2) Remove those unrecognizable text areas (some examples are shown in Figure 4(b)). 3) Annotate the reserved text areas. Finally, we collect 5,456 samples for VCTR to verify the effectiveness of our method. Some examples of this dataset are shown in Figure 4(a).

4.2 Evaluation Methods

We follow [Chen *et al.*, 2021c] to utilize four rules to convert the predictions and labels: 1) Convert full-width characters to half-width characters; 2) Convert traditional Chinese characters to simplified characters; 3) Convert uppercase letters to lowercase letters; 4) Remove all spaces. After transforming the predictions and labels, two mainstream metrics are employed to evaluate our method: Accuracy (ACC) and Normalized Edit Distance (NED). The ACC is calculated as follows:

$$\text{ACC} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(y_i = \hat{y}_i) \tag{10}$$

where y_i and $\hat{y}_i$ denote the i-th transformed prediction and label, respectively; N is the number of text images; $\mathbb{I}(\cdot)$ denotes the indication function. The NED is computed by:

$$\text{NED} = 1 - \frac{1}{N}\sum_{i=1}^{N}\text{ED}(y_i, \hat{y}_i)/\text{maxlen}(y_i, \hat{y}_i) \tag{11}$$

where "ED" and "maxlen" represent the edit distance and the maximum sequence length, respectively.

Method	Rotation	$\mathcal{L}_c$	$\mathcal{L}_o$	$\mathcal{L}_r$	ACC / NED
TransOCR (base)					67.98 / 0.815
Ours	✓				69.54 / 0.836
Ours	✓	✓	✓		69.96 / 0.841
Ours	✓	✓		✓	71.53 / 0.850
Ours	✓		✓	✓	72.08 / 0.853
Ours (final)	✓	✓	✓	✓	**73.17 / 0.865**

Table 1: The ablation studies of our method on the validation set of the scene dataset. All the training settings of these methods (*e.g.*, the training data) are the same. The column "Rotation" denotes that whether the rotation strategy is utilized at the training stage. $\mathcal{L}_c$, $\mathcal{L}_o$, and $\mathcal{L}_r$ represent whether the content classification loss, the orientation classification loss, and the character image reconstruction loss are used for supervision, respectively.

4.3 Implementation Details

Our method is implemented with PyTorch, and all experiments are conducted on an NVIDIA RTX 2080Ti GPU with 11GB memory. The AdaDelta [Zeiler, 2012] optimizer is adopted to train our model with an initial learning rate 1.0, and the hyperparameters ρ and weight decay are set to 0.9 and 10^{-4}, respectively. The batch size is set to 64. For fair comparison with the previous method [Chen *et al.*, 2021c], the input text images are resized into 32×256. For vertical text images, we follow [Li *et al.*, 2019] to rotate them by 90 degrees anti-clockwise. However, different from the rule in [Li *et al.*, 2019] that regards the samples with height larger than width as vertical ones, we assume that the samples with height larger than $1.5\times$ width are vertical text images.

4.4 Ablation Study

To disentangle the content and orientation information from visual features, we introduce three additional loss functions to supervise the proposed CIRN. In this section, we conduct ablation studies on these key loss functions and the rotation strategy, and the experimental results are shown in Table 1. All methods are trained on the training set of the scene dataset. Through experimental results, we observe that introducing the rotation strategy and two classification losses (*i.e.*,

Figure 5: Comparison of recognition results on the scene dataset. The vertical text images are rotated by 90 degrees anticlockwise for convenience. The proposed method performs well on both the horizontal and vertical texts. The characters in red are wrongly predicted.

Figure 6: Comparison of recognition results on VCTR. The shown samples are rotated by 90 degrees anticlockwise.

$\mathcal{L}_c$ and $\mathcal{L}_o$) can achieve 1.56% and 0.42% improvement, respectively. When adding the character image reconstruction loss $\mathcal{L}_r$, the content and orientation information can be better disentangled and our method achieves the best performance.

4.5 Experimental Results

In the following experiments, our method is trained on the training set of the scene dataset in [Chen *et al.*, 2021c]. At the training stage, we evaluate the performance of our method on the validation set of the scene dataset, and reserve the optimal model to test on the test set. For all experiments, the number of heads in the transformer-based decoder is set to 4. Following [Chen *et al.*, 2021c], we select CRNN [Shi *et al.*, 2016], ASTER [Shi *et al.*, 2018], MORAN [Luo *et al.*, 2019], SAR [Li *et al.*, 2019], SEED [Qiao *et al.*, 2020], and TransOCR [Chen *et al.*, 2021a] for comparison.

Hyperparameter			ACC / NED
α	β	γ	
1	1	1	70.13 / 0.832
1	1	2	71.25 / 0.841
1	1	5	**73.17 / 0.865**
1	2	5	72.31 / 0.847
1	5	5	70.76 / 0.839
2	1	5	70.30 / 0.827
5	1	5	68.15 / 0.815

Table 2: The experimental results of choosing appropriate hyperparameters. All results are evaluated on the validation dataset of the scene dataset.

Choices of hyperparameters. In the overall loss function, three hyperparameters are adopted to balance four introduced loss functions. We conduct experiments to choose the appropriate hyperparameters. As shown in Table 2, when α and β are set to 1, our method achieves relatively better performance. A possible reason is that the content and orientation information classification is easy to optimize in our method. Differently, our method achieves the best performance when γ is set to 5, indicating that the character image reconstruction is crucial for disentangling the orientation and content information.

Experiments on the scene dataset. We only conduct experiments on the scene dataset in [Chen *et al.*, 2021c] since more vertical text images are contained in this scenario. The experimental results shown in Table 1 demonstrate that the rotation strategy can alleviate the recognition of vertical text images to some extent, and achieve 1.56% improvement in accuracy compared with the baseline model. Therefore, our method also adopts this strategy at the training and test stage. Through disentangling the content and orientation information, our method surpasses the SOTA model TransOCR by

Method	Dataset	
	Scene	VCTR
CRNN [Shi *et al.*, 2016]	54.94 / 0.742	8.99 / 0.173
ASTER [Shi *et al.*, 2018]	59.37 / 0.801	19.70 / 0.434
MORAN [Luo *et al.*, 2019]	54.68 / 0.710	17.43 / 0.328
SAR [Li *et al.*, 2019]	53.80 / 0.738	9.53 / 0.187
SEED [Qiao *et al.*, 2020]	45.37 / 0.708	8.32 / 0.193
TransOCR [Chen *et al.*, 2021a]	67.81 / 0.817	18.35 / 0.341
Ours	**73.29 / 0.866**	**63.98 / 0.863**

Table 3: The experimental results on the test sets of the scene dataset and VCTR. ACC/NED follows the percentage format and decimal format, respectively.

Structure	ACC	NED
Transformer Encoder	67.97	0.834
2× Transformer Encoder	61.29	0.785
1×1 Convolution (Ours)	**73.17**	**0.865**

Table 4: Comparison of different structures adopted in the content information extractor. "2×" indicates that two transformer layers are stacked as the structure of the content information extractor.

5.48% in accuracy, which indicates the effectiveness of our method in tackling Chinese text recognition. We visualize some recognition results of the scene dataset in Figure 5. Through the visualization, we observe that the proposed method performs better on both horizontal and vertical texts. Additionally, compared with previous methods, the proposed method is more robust to artistic texts in scene images, which benefits from that the proposed CIRN implicitly pulls features of characters close to that of corresponding printed characters images.

Experiments on VCTR. To further validate the effectiveness of our method in tackling vertical text images, we also conduct experiments on the proposed VCTR dataset. Since the VCTR dataset is only used for test, our model is also trained on the scene dataset. The experimental results are shown in Table 3. Compared with previous methods, the proposed method achieves the best performance on the VCTR dataset. Specifically, our method surpasses the SOTA method [Chen *et al.*, 2021a] by around 45%. Although our method is not fine-tuned on training sets similar to VCTR, it still achieves satisfying performance. Some recognition results of VCTR are shown in Figure 6.

5 Discussions

Adopting complex structures for the content information extraction. In the content information extractor, we adopt a simple 1×1 convolution layer rather than a more complex structure. In this module, we additionally try to utilize a multi-layer transformer encoder to disentangle the content information from visual features. Through the experimental results shown in Table 4, we observe that when a more complex structure is employed in the content information extractor, the performance of our method decreases clearly, which may result from two reasons: 1) The transformer encoder contains more parameters while vertical text images account for a small share. Thus, a more complex structure for the content information extractor is hard to converge with the training of limited vertical text samples. 2) The content information can be easily disentangled from the extracted visual features with the supervision of the content classification loss $\mathcal{L}_c$ and character image reconstruction loss $\mathcal{L}_r$.

Figure 7: Failure cases of our method. Some zero-shot or few-shot characters in text images still bring difficulties to our method.

Is it better to reconstruct the whole text image? In this paper, we propose to reconstruct character images with different orientations, thus forcing the content information extractor to produce orientation-independent visual features. Additionally, we have attempted to reconstruct the whole text images to complete disentangling. Through experiments, however, we observe that the reconstruction loss cannot descend smoothly and the printed text images are not reconstructed well. A possible reason is that the reconstruction network cannot be aware of the position of each character. Therefore, we choose to reconstruct the printed image of each character in the input text image.

Failure cases. Some failure cases are shown in Figure 7. The vertical text images with a large height-width ratio are still challenging for our method. In addition, our method also has difficulties in solving text images containing few-shot or zero-shot characters as well as previous methods. Finally, since we only consider the vertical and horizontal orientation, the performance of our method on oblique texts can be further improved.

6 Conclusion

In this paper, we propose to extract orientation-independent features for Chinese text recognition by disentangling the content and orientation information. Specifically, we develop a character image reconstruction network to generate corresponding printed character images with two types of disentangled information. The proposed method surpasses previous methods on a scene dataset of Chinese text recognition. To benchmark the performance of existing methods on vertical text images, we collect a vertical Chinese text recognition dataset. Compared with state-of-the-art methods, the proposed method achieves around 45% improvement.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (No.62176060), STCSM projects (No.20511100400, No.22511105000), Shanghai Municipal Science and Technology Major Project (No.2021SHZDZX0103), 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

[Cao *et al.*, 2020] Zhong Cao, Jiang Lu, Sen Cui, and Changshui Zhang. Zero-shot handwritten chinese character recognition with hierarchical decomposition embedding. *Pattern Recognition*, 107:107488, 2020.

[Chen *et al.*, 2021a] Jingye Chen, Bin Li, and Xiangyang Xue. Scene text telescope: Text-focused scene image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12026–12035, 2021.

[Chen *et al.*, 2021b] Jingye Chen, Bin Li, and Xiangyang Xue. Zero-shot chinese character recognition with stroke-level decomposition. *International Joint Conference on Artificial Intelligence*, 2021.

[Chen *et al.*, 2021c] Jingye Chen, Haiyang Yu, Jianqi Ma, Mengnan Guan, Xixi Xu, Xiaocong Wang, Shaobo Qu, Bin Li, and Xiangyang Xue. Benchmarking chinese text recognition: Datasets, baselines, and an empirical study. *arXiv preprint arXiv:2112.15093*, 2021.

[Cheng *et al.*, 2017] Zhanzhan Cheng, Fan Bai, Yunlu Xu, Gang Zheng, Shiliang Pu, and Shuigeng Zhou. Focusing attention: Towards accurate text recognition in natural images. In *Proceedings of the IEEE international conference on computer vision*, pages 5076–5084, 2017.

[Chng *et al.*, 2019] Chee Kheng Chng, Yuliang Liu, Yipeng Sun, Chun Chet Ng, Canjie Luo, Zihan Ni, ChuanMing Fang, Shuaitao Zhang, Junyu Han, Errui Ding, et al. Icdar2019 robust reading challenge on arbitrary-shaped text-rrc-art. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1571–1576. IEEE, 2019.

[Du *et al.*, 2022] Yongkun Du, Zhineng Chen, Caiyan Jia, Xiaoting Yin, Tianlun Zheng, Chenxia Li, Yuning Du, and Yu-Gang Jiang. Svtr: Scene text recognition with a single visual model. *arXiv preprint arXiv:2205.00159*, 2022.

[Fang *et al.*, 2021] Shancheng Fang, Hongtao Xie, Yuxin Wang, Zhendong Mao, and Yongdong Zhang. Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7098–7107, 2021.

[Graves *et al.*, 2006] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376, 2006.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[Jaderberg *et al.*, 2015] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. *Advances in neural information processing systems*, 28, 2015.

[Jiang *et al.*, 2022] Gangwei Jiang, Shiyao Wang, Tiezheng Ge, Yuning Jiang, Ying Wei, and Defu Lian. Self-supervised text erasing with controllable image synthesis. *arXiv preprint arXiv:2204.12743*, 2022.

[Laptev *et al.*, 2016] Dmitry Laptev, Nikolay Savinov, Joachim M Buhmann, and Marc Pollefeys. Ti-pooling: transformation-invariant pooling for feature learning in convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 289–297, 2016.

[Li *et al.*, 2019] Hui Li, Peng Wang, Chunhua Shen, and Guyu Zhang. Show, attend and read: A simple and strong baseline for irregular text recognition. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 8610–8617, 2019.

[Liao *et al.*, 2019] Minghui Liao, Jian Zhang, Zhaoyi Wan, Fengming Xie, Jiajun Liang, Pengyuan Lyu, Cong Yao, and Xiang Bai. Scene text recognition from two-dimensional perspective. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8714–8721, 2019.

[Liu *et al.*, 2018] Wei Liu, Chaofeng Chen, and Kwan-Yee Wong. Char-net: A character-aware neural network for distorted scene text recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[Luo *et al.*, 2019] Canjie Luo, Lianwen Jin, and Zenghui Sun. Moran: A multi-object rectified attention network for scene text recognition. *Pattern Recognition*, 90:109–118, 2019.

[Neumann and Matas, 2012] Lukas Neumann and Jiri Matas. Real-time scene text localization and recognition. *Computer Vision and Pattern Recognition*, 2012.

[Qiao *et al.*, 2020] Zhi Qiao, Yu Zhou, Dongbao Yang, Yucan Zhou, and Weiping Wang. Seed: Semantics enhanced encoder-decoder framework for scene text recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13528–13537, 2020.

[Sermanet and LeCun, 2011] Pierre Sermanet and Yann LeCun. Traffic sign recognition with multi-scale convolutional networks. In *The 2011 international joint conference on neural networks*, pages 2809–2813. IEEE, 2011.

[Shi *et al.*, 2016] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based

sequence recognition and its application to scene text recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(11):2298–2304, 2016.

[Shi *et al.*, 2017] Baoguang Shi, Cong Yao, Minghui Liao, Mingkun Yang, Pei Xu, Linyan Cui, Serge Belongie, Shijian Lu, and Xiang Bai. Icdar2017 competition on reading chinese text in the wild (rctw-17). In *2017 14th iapr international conference on document analysis and recognition (ICDAR)*, volume 1, pages 1429–1434. IEEE, 2017.

[Shi *et al.*, 2018] Baoguang Shi, Mingkun Yang, Xinggang Wang, Pengyuan Lyu, Cong Yao, and Xiang Bai. Aster: An attentional scene text recognizer with flexible rectification. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2035–2048, 2018.

[Su *et al.*, 2023] Shangchao Su, Haiyang Yu, Bin Li, and Xiangyang Xue. Privacy-preserving collaborative chinese text recognition with federated learning. *arXiv preprint arXiv:2305.05602*, 2023.

[Sun *et al.*, 2019] Yipeng Sun, Zihan Ni, Chee-Kheng Chng, Yuliang Liu, Canjie Luo, Chun Chet Ng, Junyu Han, Errui Ding, Jingtuo Liu, Dimosthenis Karatzas, et al. Icdar 2019 competition on large-scale street view text with partial labeling-rrc-lsvt. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1557–1562. IEEE, 2019.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[Wan *et al.*, 2020] Zhaoyi Wan, Minghang He, Haoran Chen, Xiang Bai, and Cong Yao. Textscanner: Reading characters in order for robust scene text recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[Wang *et al.*, 2011] Kai Wang, Boris Babenko, and Serge Belongie. End-to-end scene text recognition. *International Conference on Computer Vision*, 2011.

[Wang *et al.*, 2018] Wenchao Wang, Jianshu Zhang, Jun Du, Zi-Rui Wang, and Yixing Zhu. Denseran for offline handwritten chinese character recognition. *International Conference on Frontiers in Handwriting Recognition*, 2018.

[Wang *et al.*, 2019] Zihao Wang, Xihui Liu, Hongsheng Li, Lu Sheng, Junjie Yan, Xiaogang Wang, and Jing Shao. Camp: Cross-modal adaptive message passing for text-image retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5764–5773, 2019.

[Wang *et al.*, 2021] Yuxin Wang, Hongtao Xie, Shancheng Fang, Jing Wang, Shenggao Zhu, and Yongdong Zhang. From two to one: A new scene text recognizer with visual language modeling network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14194–14203, 2021.

[Wu *et al.*, 2019] Changjie Wu, Zi-Rui Wang, Jun Du, Jianshu Zhang, and Jiaming Wang. Joint spatial and radical analysis network for distorted chinese character recognition. In *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, volume 5, pages 122–127. IEEE, 2019.

[Yang *et al.*, 2019] Mingkun Yang, Yushuo Guan, Minghui Liao, Xin He, Kaigui Bian, Song Bai, Cong Yao, and Xiang Bai. Symmetry-constrained rectification network for scene text recognition. *International Conference on Computer Vision*, 2019.

[Yin *et al.*, 2017] Fei Yin, Yi-Chao Wu, Xu-Yao Zhang, and Cheng-Lin Liu. Scene text recognition with sliding convolutional character models. *arXiv preprint arXiv:1709.01727*, 2017.

[Yu *et al.*, 2020] Deli Yu, Xuan Li, Chengquan Zhang, Tao Liu, Junyu Han, Jingtuo Liu, and Errui Ding. Towards accurate scene text recognition with semantic reasoning networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12113–12122, 2020.

[Yu *et al.*, 2022] Haiyang Yu, Jingye Chen, Bin Li, and Xiangyang Xue. Chinese character recognition with radical-structured stroke trees. *arXiv preprint arXiv:2211.13518*, 2022.

[Yuan *et al.*, 2019] Tai-Ling Yuan, Zhe Zhu, Kun Xu, Cheng-Jun Li, Tai-Jiang Mu, and Shi-Min Hu. A large chinese text dataset in the wild. *Journal of Computer Science and Technology*, 34(3):509–521, 2019.

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

[Zhan and Lu, 2019] Fangneng Zhan and Shijian Lu. Esir: End-to-end scene text recognition via iterative image rectification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2059–2068, 2019.

[Zhang *et al.*, 2019] Rui Zhang, Yongsheng Zhou, Qianyi Jiang, Qi Song, Nan Li, Kai Zhou, Lei Wang, Dong Wang, Minghui Liao, Mingkun Yang, et al. Icdar 2019 robust reading challenge on reading chinese text on signboard. In *2019 international conference on document analysis and recognition (ICDAR)*, pages 1577–1581. IEEE, 2019.

[Zu *et al.*, 2022] Xinyan Zu, Haiyang Yu, Bin Li, and Xiangyang Xue. Chinese character recognition with augmented character profile matching. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 6094–6102, 2022.