MSA^2: Multi-task Framework with Structure-aware and Style-adaptive Character Representation for Open-set Chinese Text Recognition

Yangfu Li[♣] Hongjian Zhan[♣†*] Qi Liu[♣] Li Sun[♣] Yu-Jie Xiong[‡] Yue Lu[♣]

[♣]East China Normal University, [†]Chongqing Institute of East China Normal University, [‡]Shanghai University of Engineering Science

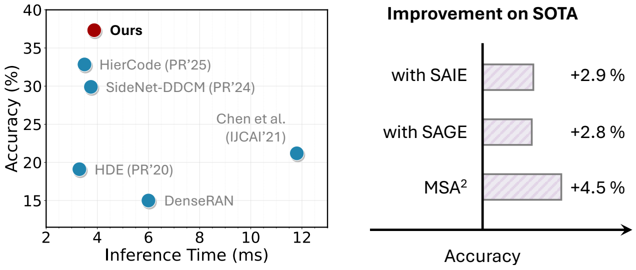
{yfli_cee, qiliu}@stu.ecnu.edu.cn, hjzhan@cee.ecnu.edu.cn, ylu@cs.ecnu.edu.cn

Abstract

Most existing methods regard open-set Chinese text recognition (CTR) as a single-task problem, primarily focusing on prototype learning of linguistic components or glyphs to identify unseen characters. In contrast, humans identify characters by integrating multiple perspectives, including linguistic and visual cues. Inspired by this, we propose a multi-task framework termed MSA^2, which considers multi-view character representations for open-set CTR. Within MSA^2, we introduce two novel strategies for character representation: structure-aware component encoding (SACE) and style-adaptive glyph embedding (SAGE). SACE utilizes a binary tree with dynamic representation space to emphasize the primary linguistic components, thereby generating structure-aware and discriminative linguistic representations for each character. Meanwhile, SAGE employs glyph-centric contrastive learning to aggregate features from diverse forms, yielding robust glyph representations for the CTR model to adapt to the style variations among various fonts. Extensive experiments demonstrate that our proposed MSA^2 outperforms state-of-the-art CTR methods, achieving average improvements of 1.3% and 6.0% in accuracy under closed-set and open-set settings on the BCTR dataset, respectively. The code is available at https://github.com/LPAIS/MSA-2.

1. Introduction

Chinese Text Recognition (CTR) is a fundamental task in computer vision that has been extensively studied for decades [4, 18, 23, 30, 33–35, 37, 41]. Unlike Latin, Chinese vocabulary is vast and continuously expanding, which naturally leads to open-set recognition challenges, *i.e.*, requiring recognizers to identify out-of-vocabulary characters, in real-world applications. Conventional CTR methods must be fine-tuned with updated vocabularies whenever new Chinese characters emerge, which is very inefficient and resource-consuming.



(a) (b)

Figure 1. Performance comparisons of MSA^2 and previous methods on ICDAR2013 in open-set scenarios. (a) accuracy *vs*. inference time; (b) improvements from three key strategies.

To address the open-set recognition problem, existing solutions can be broadly categorized into linguistic and glyph-based methods. Linguistic methods generate a unified representation for both seen and unseen characters by decomposing them into more basic linguistic components, such as stroke order [3, 16, 29], radical distribution [11, 31], structured radicals [21, 38, 42, 45, 46], and hierarchical information [1, 48]. Open-set recognition is achieved by matching the predicted sequence with a representation lexicon. In contrast, glyph-based methods directly assess the similarity of features between the input and glyphs rendered in a standard form (*e.g*., printed), including glyph-based prototype learning [13, 15, 36, 47] and deep matching networks [12, 14, 44]. Despite these advancements, existing approaches generally treat open-set CTR as a single-task problem, where the potential to integrate linguistic knowledge and glyphs has not been fully exploited.

When encountering unseen text images, native Chinese speakers typically utilize both linguistic knowledge and glyphs to infer their categories. Moreover, humans can easily recognize characters containing error secondary structures but struggle with those exhibiting error primary structures, as illustrated in Figure 2. This argues that humans rely more on primary structures than secondary structures

[*]Corresponding Author

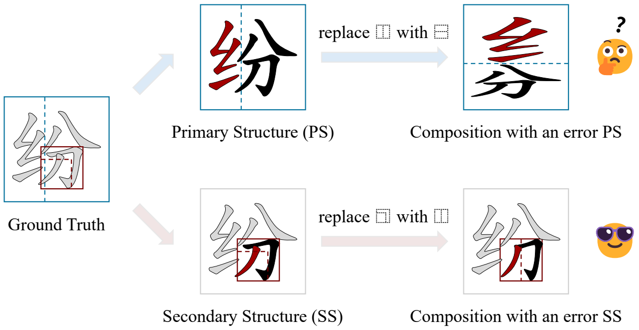


Figure 2. Impact of different structures on identifying the character, showing the primary structure is more crucial on recognition.

to identify characters. In addition, children learn to recognize characters by reading them in various forms, such as printed, handwritten, and artistic fonts, allowing them to develop robust character representations and naturally adapt to style variations across different fonts.

Inspired by human recognition of unseen Chinese texts, we propose a **M**ulti-task framework termed MSA^2 for open-set CTR, which incorporates two novel techniques: **S**tructure-**A**ware Component Encoding (SACE) and **S**tyle-**A**daptive Glyph Embedding (SAGE). To simultaneously leverage linguistic knowledge and glyphs during recognition, MSA^2 first applies SACE and SAGE to generate representation lexicons from different perspectives. Multi-task decoding is then applied to predict both linguistic components and glyph representations, along with similarity searching between predictions and representation lexicons to identify input text in open-set settings without fine-tuning. In SACE, linguistic components are organized using a binary tree and represented in dynamic space based on their contribution to recognition, encouraging the recognizer to prioritize primary structures. For SAGE, a specially designed glyph-centric contrastive learning pipeline derives robust glyph representations from a set of glyphs with diverse forms, significantly enhancing the style adaptation of the recognizer. As illustrated in Figure 1, MSA^2 is effective and efficient in recognizing unseen characters. Furthermore, comprehensive experiments on CTR benchmarks demonstrate that the proposed method achieves state-of-the-art results in both open-set and closed-set CTR.

In summary, our contributions are as follows:

- Inspired by human Chinese text recognition, we propose a multi-task framework, MSA^2, for open-set CTR, which unifies the representation of linguistic components and glyphs and employs multi-task decoding to predict them.
- We propose SACE to generate structure-aware representations for each character based on linguistic components, encouraging the model to prioritize essential structures.
- We present SAGE to create robust glyph representations via contrastive learning for the recognizer, enhancing the style adaptation for various fonts during recognition.
- Extensive experiments validate that MSA^2 outperforms previous CTR methods by a clear margin in both closed-set and open-set settings. Besides, SACE creates more effective linguistic representations, while SAGE enhances recognition, particularly for text with non-standard forms.

2. Related Work

2.1. Linguistic Methods

Linguistic methods involve supervised learning of the fundamental linguistic components of Chinese characters. Some researchers consider the open-set CTR problem from a stroke perspective. Liu *et al.* [16] and Su *et al.* [29] focus on extracting reliable stroke data for recognition, while Chen *et al.* [3] treat Chinese characters as sequences of strokes and employ a matching-based strategy for identification. Regarding radicals, several studies design radical count decoders to categorize inputs into different radical groups and predict their corresponding counts [11, 31]. To mitigate the problem of many-to-one mapping, Zhang *et al.* [46] and Yang *et al.* [38] utilize the Ideographic Description Sequence (IDS) to represent characters and predict IDS with RNN- and Transformer-based decoders. Moreover, Cao *et al.* [1] propose a hierarchical decomposition embedding (HDE) to represent character structures and align the embedding space with the visual feature space using cosine similarity. Recently, HierCode [48] introduced a lightweight framework for efficient open-set text recognition using the hierarchical linguistic information of the characters. However, these methods treat all linguistic components as equally significant for recognition, neglecting the distinction between primary and secondary structures, which leads to suboptimal performance.

2.2. Glyph-based Methods

Glyph-based methods regard character instances as indivisible units and employ deep matching or prototype learning to solve the problems in open-set text recognition. Xiao *et al.* [36] introduce an instance loss to constrain character glyphs and enhance recognition. Li *et al.* [12] and Zhang *et al.* [44] view the open-set CTR as a visual matching problem, achieving character recognition through deep matching networks and glyph sample localization, respectively. OpenCCD [14] uses a residual network to extract domain-specific visual features and predicts characters with a cosine similarity-based classifier. In [15], a label-to-prototype learning framework is proposed to emphasize intrinsic component information for open-set CTR. SideNet [13] specifically designs a counting-based spatial conversion module for glyph representation and develops a transformer-based classifier for recognition. Although these methods achieve satisfactory performance on various CTR benchmarks, they

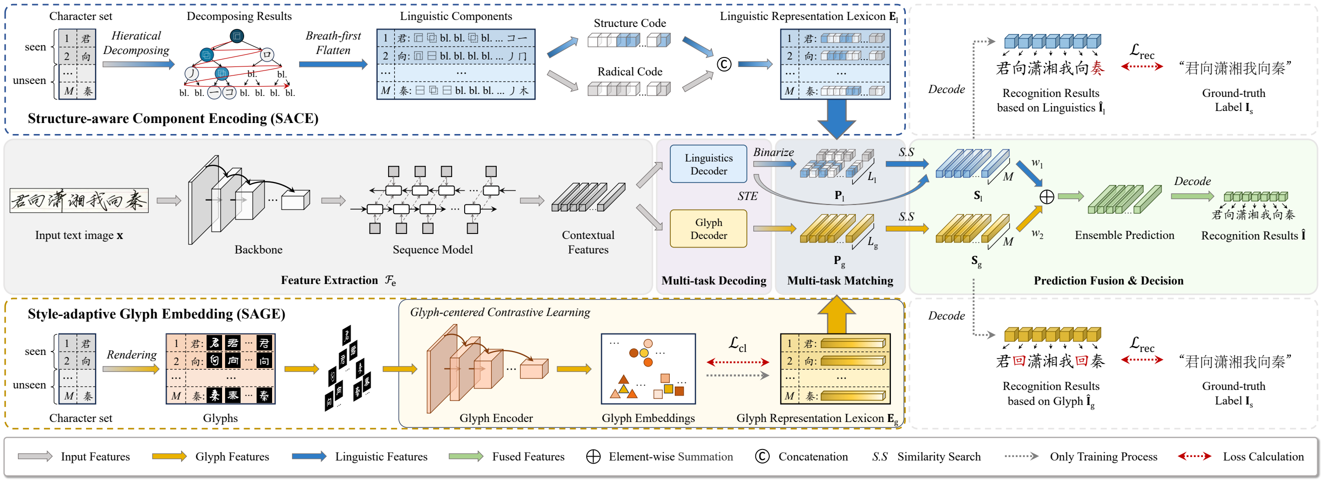

Character set
Decomposing Results
Linguistic Components
Structure Code
Radical Code
Linguistic Representation Lexicon $\mathbf{E}_l$
seen
unseen
1 君
2 向
...
M 秦
Hieratical Decomposing
Breath-first Flatten
bl.
1 君: bl. bl. bl. ... コ一
2 向: bl. bl. bl. bl. ... 丿冂
M 秦: bl. bl. bl. ... 丿木
©
Structure-aware Component Encoding (SACE)
Decode
君向潇湘我向秦
$\mathcal{L}_{\mathrm{rec}}$
"君向潇湘我向秦"
Recognition Results based on Linguistics $\hat{\mathbf{I}}_l$
Ground-truth Label $\mathbf{I}_s$
Input text image $\mathbf{x}$
Backbone
Sequence Model
Contextual Features
Feature Extraction $\mathcal{F}_e$
Linguistics Decoder
Glyph Decoder
Binarize
STE
Multi-task Decoding
L_l
$\mathbf{P}_l$
L_g
$\mathbf{P}_g$
Multi-task Matching
S.S
S.S
M
$\mathbf{S}_l$
$\mathbf{S}_g$
w_1
w_2
Ensemble Prediction
Decode
君向潇湘我向秦
Recognition Results $\hat{\mathbf{I}}$
Prediction Fusion & Decision
Style-adaptive Glyph Embedding (SAGE)
Rendering
Glyphs
Glyph-centered Contrastive Learning
Glyph Encoder
Glyph Embeddings
$\mathcal{L}_{\mathrm{cl}}$
Glyph Representation Lexicon $\mathbf{E}_g$
Decode
君回潇湘我回秦
$\mathcal{L}_{\mathrm{rec}}$
"君向潇湘我向秦"
Recognition Results based on Glyph $\hat{\mathbf{I}}_g$
Ground-truth Label $\mathbf{I}_s$
Input Features
Glyph Features
Linguistic Features
Fused Features
⊕ Element-wise Summation
© Concatenation
S.S Similarity Search
Only Training Process
Loss Calculation

Figure 3. Overall architecture of the proposed MSA^2, consisting of a multi-task recognition paradigm, structure-aware component encoding (SACE), and style-adaptive glyph embedding (SAGE). SACE and SAGE produce canonical representations of Chinese characters based on linguistic knowledge and glyphs for the recognizer, respectively. 'STE' represents the straight-through estimator.

are sensitive to style variations in text and struggle with recognizing samples with non-standard forms, *e.g.*, handwritten or artistic text, which limits their potential applications in real-world scenarios.

3. Methodology

3.1. Overview

To effectively leverage linguistic knowledge and glyphs for recognition, we propose a novel multi-task framework consisting of three key components: SACE, SAGE, and a multi-task recognition paradigm. As shown in Figure 3, SACE and SAGE are separately designed to generate the representation lexicon based on linguistics and glyphs: $\mathbf{E}_\mathrm{l} \in \mathbb{R}^{M\times L_\mathrm{l}}$ and $\mathbf{E}_\mathrm{g} \in \mathbb{R}^{M\times L_\mathrm{g}}$, where M is the vocabulary size, L_l and L_g denote the dimensions of the linguistic and glyph representations, respectively. The recognition paradigm then utilizes these representation lexicons through multi-task decoding and similarity searching to identify the input.

Recognition Paradigm The combination of a backbone and a sequence model is responsible for extracting contextual features of the input $\mathbf{x}$. Subsequently, two lightweight decoders, *i.e.*, linguistics decoder $\mathcal{D}_\mathrm{l}$ and glyph decoder $\mathcal{D}_\mathrm{g}$, are employed separately to predict the corresponding representations, *i.e.*, $\mathbf{P}_\mathrm{l}$ and $\mathbf{P}_\mathrm{g}$, from the contextual features:

$$\begin{aligned}\mathbf{P}_\mathrm{l} &= \mathrm{Binarize}(\mathcal{D}_\mathrm{l}(\mathcal{F}_\mathrm{e}(\mathbf{x}))) \in \mathbb{R}^{N\times L_\mathrm{l}},\\ \mathbf{P}_\mathrm{g} &= \mathcal{D}_\mathrm{g}(\mathcal{F}_\mathrm{e}(\mathbf{x})) \in \mathbb{R}^{N\times L_\mathrm{g}},\end{aligned} \tag{1}$$

where $\mathcal{F}_\mathrm{e}$ denotes the function for feature extraction, and N denotes the sequence length. Notably, we binarize the output of the linguistics decoder to more precisely describe the discrete linguistic representation. The similarity searching is then conducted between the predicted representations and the representation lexicons to determine the characters:

$$\mathbf{S}_\mathrm{l} = \mathbf{P}_\mathrm{l} \cdot (\mathbf{E}_\mathrm{l})^\top, \quad \mathbf{S}_\mathrm{g} = \mathbf{P}_\mathrm{g} \cdot (\mathbf{E}_\mathrm{g})^\top \in \mathbb{R}^{N\times M}, \tag{2}$$

where $\mathbf{S}_\mathrm{l}$, $\mathbf{S}_\mathrm{g}$ denotes the prediction based on linguistics and glyph. Finally, an element-wise summation is performed to fuse the predictions, yielding the recognition result $\hat{\mathbf{I}}$:

$$\hat{\mathbf{I}} = \mathcal{D}_{\mathrm{seq}}(\omega_1\mathbf{S}_\mathrm{l} + \omega_2\mathbf{S}_\mathrm{g}), \tag{3}$$

where ω_1 and ω_2 are both set to 0.5 for normalization. The term $\mathcal{D}_{\mathrm{seq}}$ refers to the sequence decoder, *i.e.*, CTC or attention decoder, which is utilized to convert the ensemble predictions into recognition results.

Loss function We employ the similarity-based recognition loss proposed in [48] as the loss function for the recognizer, which is defined as the negative log likelihood between the recognition results $\hat{\mathbf{I}}$ and the label $\mathbf{I}$:

$$\mathcal{L}_{\mathrm{rec}}(\mathbf{I}, \hat{\mathbf{I}},) = -\sum \log p\,(\mathbf{I}\,|\,\hat{\mathbf{I}}). \tag{4}$$

Subsequently, the total loss $\mathcal{L}_{\mathrm{total}}$ is defined as the sum of the recognition losses $\mathcal{L}_{\mathrm{rec}}$ on both the linguistic and glyph branches, which can be expressed as follows:

$$\mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\mathrm{rec}}(\mathbf{I}_\mathrm{s}, \hat{\mathbf{I}}_\mathrm{l}) + \mathcal{L}_{\mathrm{rec}}(\mathbf{I}_\mathrm{s}, \hat{\mathbf{I}}_\mathrm{g}), \tag{5}$$

where $\hat{\mathbf{I}}_\mathrm{l}$ and $\hat{\mathbf{I}}_\mathrm{g}$ denote the recognition result based on the linguistics and glyphs, respectively. $\mathbf{I}_\mathrm{s}$ represents the ground truth label composed of seen characters.

3.2. Structure-aware Component Encoding

From a linguistic perspective, each Chinese character can be uniquely represented by a set of components, comprising

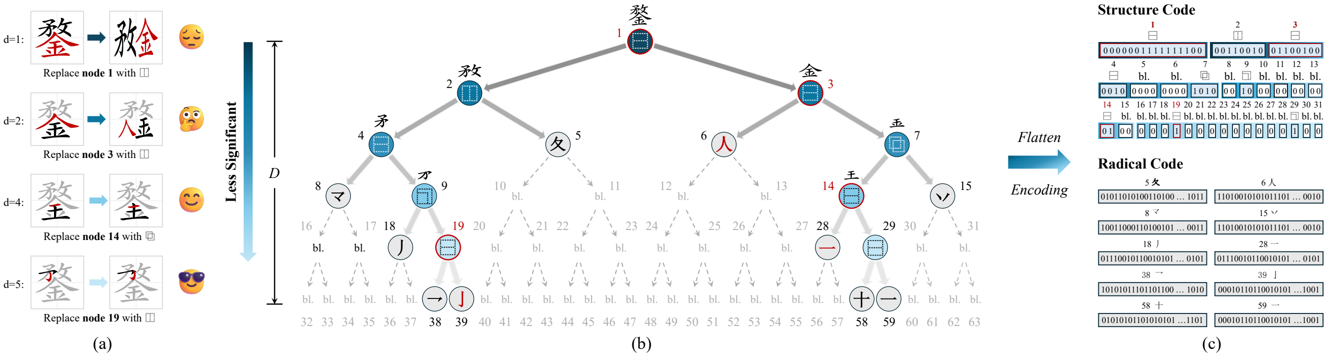


Figure 4. Illustration of (a) the impact of structures at different depths on character appearance, demonstrating that deeper structures have less impact on recognition; (b) the decomposition of the Chinese character ‘鍪’ organized with a full binary tree, where structures reside at parent nodes, radicals reside at leaf nodes, and ‘bl.’ denotes the null node for padding; (c) the generated linguistic representation.

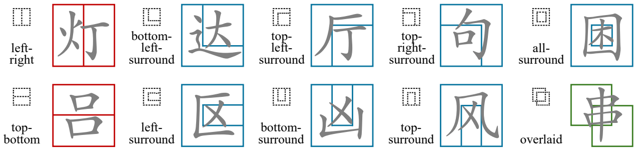


Figure 5. Ten basic structures of Chinese characters, where the ‘red,’ ‘blue,’ and ‘green’ denote the different class of structures.

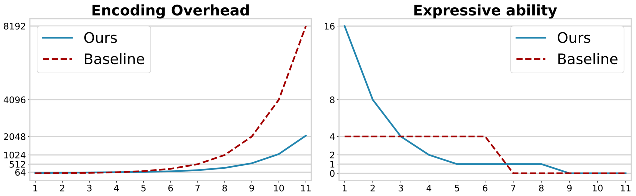


Figure 6. Comparisons of encoding overhead and expressive ability under varying decomposition iterations.

structures and radicals. As shown in Figure 4, SACE hierarchically organizes these components using a binary tree to differentiate between their significance for recognition, where the less significant components are placed in deeper layers. To align the representations of different characters, SACE constructs the tree as a full tree with a maximum depth of D. The final linguistic representation is generated by concatenating the structures and radical codes.

Structures Encoding As illustrated in Figure 5, Chinese characters exhibit ten distinct structures, which can be categorized into three classes. SACE utilizes varying-length codes to represent structures across different layers. In the shallower layers, structures are encoded with longer codes, playing a more crucial role in similarity searching, and thus influencing recognition results. Specifically, the structure at the root node is encoded with a 16-bit multi-hot code, and the code length is halved as the depth increases until it reaches 1 bit. Since the structures residing deeper than the 3rd layer have minimal impact on recognition, SACE uses the same 2-bit code to represent the same class of structures in the 4th layer. For deeper layers, SACE simply flags their structure nodes with a single bit. Notably, although secondary structures may share the same code, SACE can still generate unique and discriminative representations for each character by combining them with radicals.

To analyze the effectiveness of SACE, we compare it with the baseline using static space in terms of encoding overhead and expressive ability. The expressive ability is reflected by the size of the representation space, while the encoding overhead L_s is calculated as follows:

$$L_s = \sum_{d=1}^{D-1} L_{S_d} 2^{d-1}, \tag{6}$$

where L_{S_d} represents the size of the representation space at the d^{th} layer. As shown in Figure 6, SACE generates more informative and expressive representations than the baseline in most cases, especially for the complex characters requiring large decomposition iterations.

Radicals Encoding Dynamic space encoding increases the complexity of representations. In contrast to structures, there are far more than ten types (*i.e.*, approximately 500) of radicals needed for representing all Chinese characters. Encoding radicals with the dynamic space would yield a highly complex representation lexicon, leading to a negative impact on recognition. Therefore, the radical nodes are encoded using fixed-length multi-hot codes with 60 bits.

3.3. Style-adaptive Glyph Embedding

SAGE aims to generate the robust glyph representation lexicon, *i.e.*, $\mathbf{E}_{\text{g}} = [\boldsymbol{e}_{\text{g}}^{1}, \boldsymbol{e}_{\text{g}}^{2}, \ldots, \boldsymbol{e}_{\text{g}}^{M}]$, which involves a specially designed glyph-centric contrastive learning (GCCL) framework. As presented in Figure 7, the GCCL consists of two stages: the initialization and update of the representation lexicon and the optimization of the glyph encoder.

In the first stage, we construct a glyph set with T different styles for each character, where the details are provided

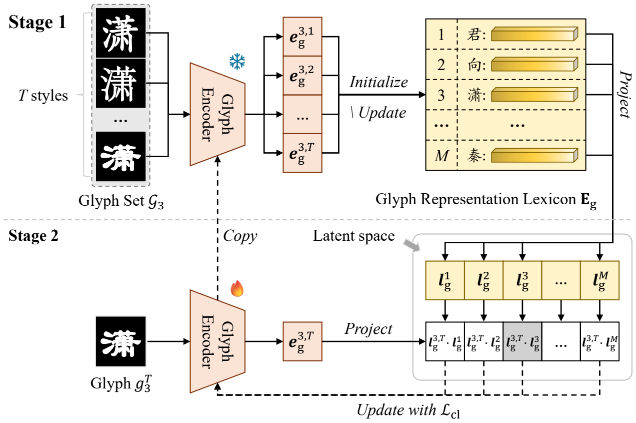


Figure 7. Pipeline of the glyph-centric contrastive learning.

in the Supplementary Material. Mathematically, the glyph set of character m is defined as $\mathcal{G}_m = \{g_m^1, g_m^2, \ldots, g_m^T\}$. We then adopt a frozen glyph encoder $\mathcal{F}_g$, derived from CLIP [26], to extract the glyph embeddings:

$$\boldsymbol{e}_g^{m,t} = \mathcal{F}_g(g_m^t), t \in [1, 2, \ldots, T], \tag{7}$$

where $\boldsymbol{e}_g^{m,t}$ is the embedding corresponding to the style t. Subsequently, the pseudo-representation $\hat{\boldsymbol{e}}_g^m$ of the character m is defined as the center of these embeddings:

$$\hat{\boldsymbol{e}}_g^m = \tfrac{1}{T} \textstyle\sum_{t=1}^{T} \boldsymbol{e}_g^{m,t}. \tag{8}$$

Finally, we adopt the pseudo-representation to initialize the glyph representation lexicon: $\hat{\boldsymbol{e}}_g^m \mapsto \boldsymbol{e}_g^m \in \mathbf{E}_g$.

In the second stage, we fine-tune the glyph encoder using the initialized representation lexicon with contrastive learning [24]. Specifically, we employ two MLP layers to project each glyph embedding and the pseudo-representations into latent space, and then calculate the contrastive loss $\mathcal{L}_{cl}$ for updating the glyph encoder and the MLP layers:

$$\mathcal{L}_{cl} = -\frac{1}{MT} \sum_{i=1}^{M} \sum_{t=1}^{T} \log \frac{\exp(\boldsymbol{l}_g^{i,t} \cdot \boldsymbol{l}_g^i)}{\sum_{j=1, j \neq i}^{M} \exp(\boldsymbol{l}_g^{i,t} \cdot \boldsymbol{l}_g^j)}, \tag{9}$$

where the $\boldsymbol{l}_g^{i,t}$ and $\boldsymbol{l}_g^i$ are the projections of the glyph embedding $\boldsymbol{e}_g^{i,t}$ and the pseudo-representation $\boldsymbol{e}_g^i$. In this way, the glyph encoder is encouraged to minimize intra-class distances and maximize inter-class distances of glyph embeddings, thereby generating discriminative representations.

By iteratively applying these two stages to each character in the vocabulary, GCCL progressively bridges the gap between the pseudo-representation and the robust representation, ultimately yielding the glyph representation lexicon.

4. Experiments

4.1. Experiment Setting

Benchmark Extensive benchmarks on both character and text recognition across various scenarios are conducted to validate the effectiveness of the MSA^2. **ICDAR2013** [39] is a handwritten Chinese competition database that includes subsets for text line data (denoted as ICDAR-line) and isolated character data (denoted as ICDAR-char), and we utilize both of them as the evaluation set. **CASIA-HWDB** [17] is a large-scale Chinese handwritten database, and we use the text line portion (*i.e.*, HWDB 2.0-2.2) and the isolated character portion (*i.e.*, HWDB 1.0-1.2) as the training set for ICDAR2013. **BCTR** [41] is a comprehensive benchmark for Chinese text images, consisting of four subsets: Scene, Web, Document (denoted as Doc), and Handwriting (denoted as Handw). **CTW** [43] contains $812,872$ Chinese character instances collected from street views across $3,650$ classes, where $760,107$ character images are used for training and $52,765$ images are reserved for testing.

Evaluation protocol Following the previous work [22, 48], we adopt line-level accuracy for each subset of BCTR to assess the performance of text recognition. To further investigate cross-lingual generalization capabilities, we analyze the averaged recall rates of different type characters on ICDAR-Line and BCTR benchmarks. For character-level evaluation, we leverage character-level accuracy on handwritten (*i.e.*, ICDAR-char) and scene characters (*i.e.*, CTW) for quantitative comparison of recognition methods.

Implementation Details The maximum depth of the binary tree, the number of encoded radicals, and the radical code length are set to 7, 16, and 60, respectively. The GCCL iteration is set to 7. We use the Adam optimizer with a learning rate of 1e-6 to fine-tune the glyph encoder. In text recognition, text images are resized to a height of 128 while maintaining their original aspect ratio. For character recognition, input images are resized to 96 × 96. Non-Chinese characters are treated as basic Chinese characters represented by a special radical (*i.e.*, themselves), allowing them to be processed consistently as Chinese characters. All experiments were conducted using PyTorch on an NVIDIA RTX 4090 GPU with 24 GB memory. For training of the recognizer, we employed the Adadelta optimizer with an initial learning rate of 0.1 and a batch size of 128. More details are provided in the Supplementary Material.

4.2. Evaluation of Text Recognition

Closed-set Recognition We evaluate the effectiveness of the proposed MSA^2 for closed-set text recognition across a broad spectrum of scenarios, which include four distinct text types: scene, web, document, and handwritten. The results are presented in Table 1. Compared with existing closed-set CTR methods, the proposed method establishes new records on all subsets of BCTR. Specifically, it surpasses SOTA methods by 1.5%, 2.4%, 0.9%, and 1.2% in accuracy on Scene, Web, Doc, and Handw. Besides, when compared with the open-set CTR baseline HierCode under identical model configurations, MSA^2 achieves a 1.2% av-

Methods	Venue	Scene	Web	Doc	Handw	Avg
CRNN [27]	PAMI'16	53.4	54.5	97.5	46.4	67.0
ASTER [28]	PAMI'18	54.6	52.3	93.1	38.9	64.7
MORAN [20]	PR'19	51.7	49.5	95.4	39.6	64.3
SAR [10]	AAAI'19	62.5	54.1	94.2	33.7	67.3
SRN [40]	CVPR'20	60.1	52.3	96.7	18.0	65.0
SEED [25]	CVPR'20	49.6	46.3	93.7	32.1	61.2
MASTER [19]	PR'21	62.8	52.1	84.4	26.9	56.6
TransOCR [2]	CVPR'21	63.3	62.3	96.9	53.4	72.8
ABINet [7]	CVPR'21	64.4	67.4	97.2	54.8	74.1
SVTR-B [6]	IJCAI'22	71.7	73.8	98.2	52.2	75.2
SVTR-L [6]		<u>72.2</u>	<u>74.1</u>	98.1	53.6	75.5
CCR-CLIP [42]	ICCV'23	71.3	69.2	<u>98.3</u>	<u>60.3</u>	<u>75.8</u>
MSA^{2,†}	-	**73.7**	**76.5**	**99.2**	**61.5**	**77.1**
Δ	-	+1.5	+2.4	+0.9	+1.2	+1.3
One-hot	-	60.3	60.2	92.8	54.1	70.0
HierCode [48]	PR'25	63.7	66.2	98.2	56.3	74.2
MSA^2	-	**65.9**	**69.4**	**98.7**	**59.2**	**75.4**
Δ_1 Δ_2	-	+2.2 +5.6	+3.2 +9.2	+0.5 +5.9	+2.9 +5.1	+1.2 +5.4

† applies the same configuration of backbone as [42] for fair comparisons.

Table 1. Comparison of recognition accuracy in sentence level (%) with previous methods on the BCTR dataset, where Δ_1 and Δ_2 separately indicate the increment of our method when compared with the Hiercode [48] and one-hot baseline over each subset and average. The first ten results are derived from [42] and [48].

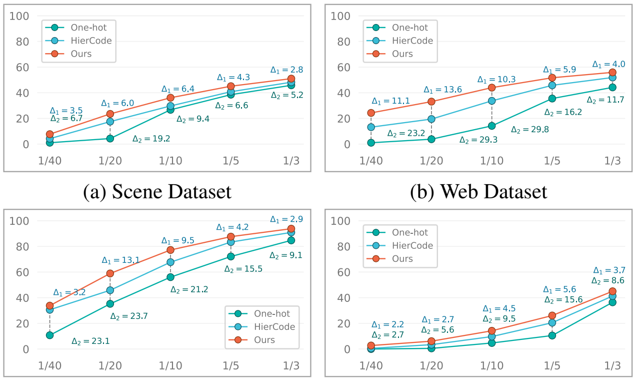
Char	Methods	ICDAR	BCTR			
		Line	Scene	Web	Doc	Handw
Ch.	One-hot	93.35	82.09	79.57	98.64	91.65
	HierCode [48]	94.53	83.41	83.39	99.71	92.35
	MSA^2	**95.54**	**85.85**	**85.64**	**99.73**	**94.38**
	Δ_1	+1.01	+2.44	+2.25	+0.02	+2.03
	Δ_2	+2.19	+3.76	+6.07	+1.09	+2.73
NCh.	One-hot	85.56	90.24	84.67	99.37	86.59
	HierCode [48]	85.65	90.27	85.19	99.54	86.61
	MSA^2	**87.73**	**92.78**	**88.02**	**99.79**	**89.22**
	Δ_1	+2.08	+2.51	+2.83	+0.25	+2.61
	Δ_2	+2.17	+2.54	+3.35	+0.42	+2.63

Table 2. Comparison of recall rate (%) of Chinese character (Ch.) and Non-Chinese characters (NCh.) on ICDAR-line and BCTR Datasets. Δ_1 and Δ_2 separately marks the improvement provided by our method for the HierCode [48] and one-hot baseline.

erage accuracy improvement across all subsets.

Furthermore, we evaluate the text recognition performance across multi-language scenarios using the ICDAR-line and BCTR datasets. As shown in Table 2, MSA^2 demonstrates significant improvements over HierCode not only for Chinese characters but also for Latin characters, numbers, and symbols across each subset. Notably, our method yields more substantial improvements for recognizing non-Chinese characters. We attribute this improvement to the incorporation of glyphs, which effectively represent these basic characters that are difficult to describe using Chinese linguistic knowledge.

Open-Set Recognition To evaluate the performance of text recognition in open-set scenarios, we train the open-set CTR models using limited data resources. Specifically, we randomly select distinct proportions p of subsets from the training data of BCTR, where $p \in \{\frac{1}{40}, \frac{1}{20}, \frac{1}{10}, \frac{1}{5}, \frac{1}{3}\}$, resulting in various test sets containing different numbers of unseen characters. As shown in Figure 8, under the same training strategy, our method consistently outperforms both the vanilla one-hot baseline and HierCode by a significant margin across all open-set conditions on four subsets. Quantitatively, compared with Hiercode, MSA^2 achieved an average accuracy improvement of 4.6%, 8.9%, 6.6%, and 3.7% on Scene, Web, Doc, and Handw. This demonstrates the effectiveness of the proposed MSA^2 for sentence-level recognition under open-set settings.



(a) Scene Dataset (b) Web Dataset (c) Document Dataset (d) Handwriting Dataset

Figure 8. Performance comparison in each subset of BCTR under data-scarce scenarios, where the x and y are separately present the proportion of training data and the line accuracy. Δ_1, and Δ_2 indicates the increment of MSA^2 when compared with the HierCode (reproduced by us) and one-hot baseline, respectively.

4.3. Evaluation of Character Recognition

We conduct character-level evaluations using handwritten and scene characters under both open-set and closed-set settings. For open-set recognition, we follow the configurations applied in the previous works [1, 3, 21, 32, 48]. Specifically, for handwritten characters, experiments are conducted on the HWDB1.0-1.1 and ICDAR-Char datasets, which comprise a total of $3,755$ characters. The first m classes from HWDB are used for training, where m ranges from $\{500, 1000, 1500, 2000, 2755\}$, while the $1,000$ classes of ICDAR-Char serve as the evaluation set. For scene characters, we utilize the CTW dataset and select samples from the first m classes as the training set, where m ranges from $\{500, 1000, 1500, 2000, 3150\}$. The last $1,000$ classes are designated as the test set. Notably, during the training phase, we perform similarity searches only for the representations of characters present in the training set. In the inference phase, the final decision is made by matching the model predictions against the complete lexicon of representations derived from both training and test samples.

As shown in Table 3, although the MSA^2 is designed

Methods	Venue	Representation		Handwritten/% (m for classes)						Scene/% (m for classes)				
		Linguistics	Glyph	500	1000	1500	2000	2755	Full	500	1000	1500	2000	3150
DenseRAN [32]	ICFHR'18	✓		1.70	8.44	14.71	19.51	30.68	96.66	0.15	0.54	1.60	1.95	5.39
HDE [1]	PR'20	✓		4.90	12.77	19.25	25.13	33.49	97.14	0.82	2.11	3.11	6.96	7.75
Chen *et al.* [3]	IJCAI'21	✓	✓̸	5.60	13.85	22.88	25.73	37.91	96.73	1.54	2.54	4.32	6.82	8.61
CUE [21]	PR'23	✓		7.43	15.75	24.01	27.04	40.55	96.96	-	-	-	-	-
SideNet [13]	PR'24	✓		5.10	16.20	33.80	44.10	50.30	-	-	-	-	-	-
HierCode [48]	PR'25	✓		6.22	20.71	35.39	45.67	56.21	97.68	1.67	2.59	4.54	7.02	9.13
MSA^2	-	✓	✓	**8.24**	**26.13**	**40.67**	**51.44**	**60.17**	**98.85**	**2.05**	**3.11**	**4.98**	**7.65**	**9.68**
Δ	-	-		+2.02	+5.96	+5.28	+5.73	+3.96	+1.17	+0.38	+0.52	+0.44	+0.63	+0.55

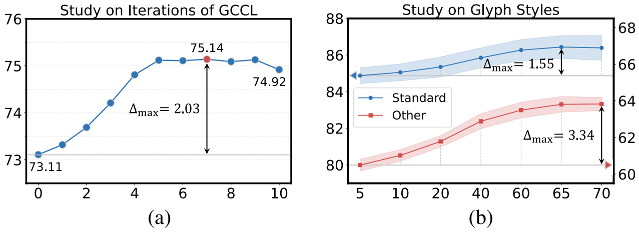
Table 3. Comparison of recognition accuracy in character level (%) under open-set setting on ICDAR-char and CTW with previous methods, where Δ denote the improvements over each setting. '✓̸' means the glyphs are only used in the testing phase.

Char	Linguistics	Glyph	Standard	Other
Compound Characters	✓		91.24	88.38
		✓	90.95	87.93
	✓	✓	**91.95**	**89.82**
Basic Characters	✓		92.11	88.57
		✓	93.25	89.26
	✓	✓	**93.43**	**90.41**

Table 4. Ablation study on the recognition task in recall rate (%) of characters with standard and other forms, where 'Standard' refers to Web and Doc, while 'Other' includes Scene and Handwriting.

L_{S_1}	L_{S_2}	L_{S_3}	L_{S_4}	L_{S_5}	L_S	AR	RR
Random Structure Code					124	94.03	92.42
4	-	-	-	-	124	96.47	94.23
8	-	-	-	-	128	96.60	94.41
16	4	-	-	-	136	96.78	94.64
16	8	4	4	-	144	97.03	94.86
16	16	-	-	-	160	96.94	94.76
16	8	8	-	-	160	96.96	94.79
16	8	2	-	-	136	96.82	94.45
16	8	-	2	4	128	97.12	94.89
16	8	-	1	-	120	97.08	94.84
16	8	-	2	2	96	97.24	95.05
16	8	-	2	1	80	**97.35**	**95.12**

Table 5. Ablation study on the representation space in terms of accurate rate (%) (AR) and recall rate (%) (RR) of compound Chinese characters. '-' denotes 4 regarded as the default set.



(a)

(b)

Figure 9. Ablation study on (a) iterations of GCCL; (b) numbers of glyph styles in GCCL; where $\Delta_{\max}$ indicates the maximum improvement. Results are measured in line accuracy (%) in BCTR.

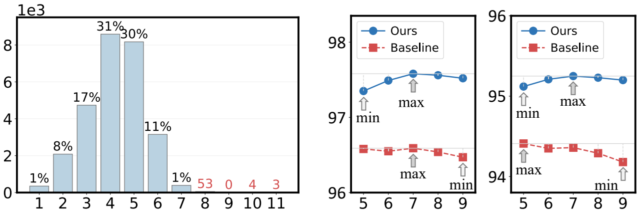
for open-set text recognition, it still achieves superior results compared to existing methods for open-set character recognition. In particular, on the handwritten dataset, MSA^2 demonstrates absolute accuracy improvements of 2.02%, 5.96%, 5.28%, 5.73%, and 3.96% at m values of $\{500, 1000, 1500, 2000, 2755\}$ when compared to the state-of-the-art method HierCode [48] with the same model configuration. On the scene text dataset, MSA^2 also achieves an average increase of over 0.5% in character-level accuracy across all open-set settings, demonstrating its effectiveness in open-set character recognition. As for closed-set character recognition, MSA^2 also improves performance by 1.17% compared to HierCode.

4.4. Ablation Study

Influence of Recognition Task As shown in Table 4, linguistic knowledge plays a more significant role than glyphs in recognizing compound characters. In contrast, for basic characters that cannot be further decomposed, glyphs provide greater benefits to recognition. This difference likely arises because compound characters contain richer linguistic components compared to basic characters. Consequently, combining these two tasks yields comprehensive improvements to recognition across various scenarios.

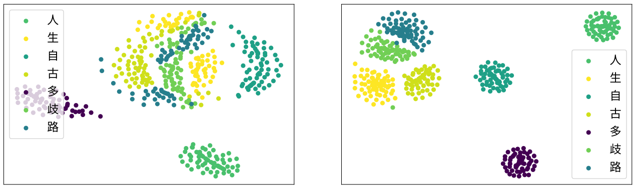
Influence of Structure Code Since basic Chinese characters lack internal structures, ablation is conducted with samples of compound Chinese characters on ICDAR-char. As reported in Table 5, the worst performance of the random code demonstrates that structures play an essential role in recognition. Furthermore, assigning a smaller representation space for the structures in deeper layers improves recognition performance, validating the key motivation of SACE. We also analyze the impact of radical code on recognition, which is presented in the Supplementary Material.

Influence of GCCL As shown in Figure 9(a), recognition performance benefits from iterations of GCCL, validating its effectiveness. However, excessive iteration does not yield continuous improvement in recognition, as it leads to overfitting and compromises the robust feature extraction ability learned by the glyph encoder through CLIP. As shown in Figure 9(b), increasing the diversity of glyph styles results in a notable improvement in recognizing text with non-standard forms. This improvement occurs because greater diversity enables SAGE to better account for style variations in non-standard forms.



(a) (b)

Figure 10. Investigation of (a) distribution of maximum iteration required for complete decomposition per character; (b) accurate rate (left), and recall rate (right) *vs*. distribution iteration.



(a) Without the GCCL (b) With the GCCL

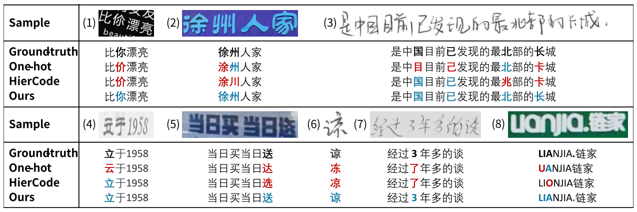
Figure 11. Glyph embedding distribution visualization of whether applying the GCCL in SAGE.

4.5. Discussion

How does SACE benefit CTR? We think the advantages of SACE can be summarized as follows:

- The representations created by SACE are consistent with human perception, where primary structures are encoded with longer bits to exert a greater influence on recognition results during the similarity searching process.
- The representations generated by SACE are more informative than those produced by baseline with static representation spaces. Specifically, we calculate the maximum iteration required for the complete decomposition of each Chinese character in the GB18030-2000 standard[1], as shown in Figure 10(a). To handle complex characters, a large maximum decomposition (*e.g.*, ≥ 5) is typically set for each character, resulting in numerous null nodes in the decomposition results of simpler characters. SACE optimizes component encoding by assigning smaller representation spaces in deeper layers, where nodes are more likely to be null, thus reducing the number of meaningless bits encoded by these null nodes in linguistic representations. This is further supported by Figure 10(b), which shows that our method benefits from more iterations compared to the baseline with static representation space.

Visualization To validate the effectiveness of GCCL, we sample 7 characters and visualize their glyph embeddings in a 2-D space using t-SNE, where each character is represented by a distinct color. As shown in Figure 11(a), the glyph embeddings generated by the glyph encoder without fine-tuning are not sufficiently discriminative, with some embeddings deviating significantly from their cluster centers in the feature space. When introducing GCC, the glyph embeddings become closer to their cluster centers (see Figure 11(b)), proving that GCCL enhances the recognizer by providing more robust glyph representations.



Sample	(1)	(2)	(3)
Groundtruth	比你漂亮	徐州人家	是中国目前已发现的最北部的长城
One-hot	比价漂亮	涂州人家	是中目目前己发现的最北部的卡城
HierCode	比价漂亮	涂川人家	是中国目前已发现的最兆部的卡城
Ours	比你漂亮	徐州人家	是中国目前已发现的最北部的长城

Sample	(4)	(5)	(6)	(7)	(8)
Groundtruth	立于1958	当日买当日送	谅	经过 3 年多的谈	LIANJIA.链家
One-hot	云于1958	当日买当日达	冻	经过了年多的谈	UANJIA链家
HierCode	立于1958	当日买当日选	凉	经过了年多的谈	LIONJIA链家
Ours	立于1958	当日买当日送	谅	经过 3 年多的谈	LIANJIA.链家

Figure 12. Visualizations analysis, where correctly and incorrectly recognized characters are marked in ‘blue’ and ‘red’, respectively.

We also present visualizations to analyze the strengths of the MSA^2. Benefiting from SACE, MSA^2 can identify characters with incomplete or ambiguous local details, as demonstrated in Figure 12 (1) and (5). Additionally, as shown in Figure 12 (7) and (8), the incorporation of visual cues from glyphs provides MSA^2 with a clear advantage in recognizing basic characters, such as numbers and Latin letters, compared to the one-hot baseline and HierCode.

Limitations MSA^2 relies on linguistic components and glyphs to represent characters. Unfortunately, some samples may lack linguistic information (*e.g.*, ancient texts) or are difficult to render with specific forms of glyphs (*e.g.*, alien characters), potentially leading to suboptimal performance. Additionally, MSA^2 has not considered the connections among same-type structures with varying importance during encoding, which will be addressed in future work.

5. Conclusion

In this paper, we introduce MSA^2, a multi-task framework for open-set Chinese text recognition (CTR), which incorporates two innovative character modeling strategies: SACE and SAGE. Inspired by human recognition of Chinese characters, MSA^2 leverages both linguistic and glyph representations to determine character categories. Within MSA^2, SACE generates more informative linguistic representations by allocating larger representation spaces to primary components. Meanwhile, SAGE enhances the robustness of glyph representations through glyph-centric contrastive learning. Our experiments show that emphasizing primary structures significantly improves recognition performance, and glyph-centric contrastive learning also benefits the recognizer through more discriminative glyph representations. Comprehensive evaluations demonstrate that MSA^2 outperforms previous CTR methods by a substantial margin in both closed-set and open-set scenarios.

[1] https://openstd.samr.gov.cn/bzgk/gb/

Acknowledgments

The research was performed at the Shanghai Key Laboratory of Multidimensional Information Processing and the Chongqing Key Laboratory of Precision Optics, with joint support from the National Natural Science Foundation of China (62176091), the Natural Science Foundation of Chongqing (CSTB2024NSCQ-MSX0877), and the AI project from the economic and information commission of Shanghai (2024-GZL-RGZN-01038).

References

[1] Zhong Cao, Jiang Lu, Sen Cui, and Changshui Zhang. Zero-shot handwritten chinese character recognition with hierarchical decomposition embedding. *Pattern Recognition*, 107: 107488, 2020. 1, 2, 6, 7

[2] 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. 6

[3] Jingye Chen, Bin Li, and Xiangyang Xue. Zero-shot chinese character recognition with stroke-level decomposition. *Proceedings of the International Joint Conference on Artificial Intelligence*, 2021. 1, 2, 6, 7

[4] Zhao Chen, Yaohua Yi, Chaohua Gan, Ziwei Tang, and Dezhu Kong. Scene chinese recognition with local and global attention. *Pattern Recognition*, page 111013, 2024. 1

[5] Kyunghyun Cho. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014. 1

[6] 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. *Proceedings of the International Joint Conference on Artificial Intelligence*, 2022. 6

[7] 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. 6

[8] 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. 1

[9] S Hochreiter. Long short-term memory. *Neural Computation MIT-Press*, 1997. 1

[10] 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*, pages 8610–8617, 2019. 6

[11] Yunqing Li, Yixing Zhu, Jun Du, Changjie Wu, and Jianshu Zhang. Radical counter network for robust chinese character recognition. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 4191–4197. IEEE, 2021. 1, 2

[12] Zhiyuan Li, Qi Wu, Yi Xiao, Min Jin, and Huaxiang Lu. Deep matching network for handwritten chinese character recognition. *Pattern Recognition*, 107:107471, 2020. 1, 2

[13] Ziyan Li, Yuhao Huang, Dezhi Peng, Mengchao He, and Lianwen Jin. Sidenet: Learning representations from interactive side information for zero-shot chinese character recognition. *Pattern Recognition*, 148:110208, 2024. 1, 2, 7

[14] Chang Liu, Chun Yang, and Xu-Cheng Yin. Open-set text recognition via character-context decoupling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4523–4532, 2022. 1, 2

[15] Chang Liu, Chun Yang, Hai-Bo Qin, Xiaobin Zhu, Cheng-Lin Liu, and Xu-Cheng Yin. Towards open-set text recognition via label-to-prototype learning. *Pattern Recognition*, 134:109109, 2023. 1, 2

[16] Cheng-Lin Liu, In-Jung Kim, and Jin H Kim. Model-based stroke extraction and matching for handwritten chinese character recognition. *Pattern Recognition*, 34(12):2339–2352, 2001. 1, 2

[17] Cheng-Lin Liu, Fei Yin, Da-Han Wang, and Qiu-Feng Wang. Casia online and offline chinese handwriting databases. In *2011 international conference on document analysis and recognition*, pages 37–41. IEEE, 2011. 5

[18] Yangyang Liu, Yi Chen, Fei Yin, and Cheng-Lin Liu. Context-aware confidence estimation for rejection in handwritten chinese text recognition. In *International Conference on Document Analysis and Recognition*, pages 134–151. Springer, 2024. 1

[19] Ning Lu, Wenwen Yu, Xianbiao Qi, Yihao Chen, Ping Gong, Rong Xiao, and Xiang Bai. Master: Multi-aspect non-local network for scene text recognition. *Pattern Recognition*, 117: 107980, 2021. 6

[20] Canjie Luo, Lianwen Jin, and Zenghui Sun. Moran: A multi-object rectified attention network for scene text recognition. *Pattern Recognition*, 90:109–118, 2019. 6

[21] Guo-Feng Luo, Da-Han Wang, Xia Du, Hua-Yi Yin, Xu-Yao Zhang, and Shunzhi Zhu. Self-information of radicals: A new clue for zero-shot chinese character recognition. *Pattern Recognition*, 140:109598, 2023. 1, 6, 7

[22] Pengyuan Lyu, Chengquan Zhang, Shanshan Liu, Meina Qiao, Yangliu Xu, Liang Wu, Kun Yao, Junyu Han, Errui Ding, and Jingdong Wang. Maskocr: Text recognition with masked encoder-decoder pretraining. *arXiv preprint arXiv:2206.00311*, 2022. 5

[23] Ronaldo Messina and Jerome Louradour. Segmentation-free handwritten chinese text recognition with lstm-rnn. In *2015 13th International conference on document analysis and recognition (icdar)*, pages 171–175. IEEE, 2015. 1

[24] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 5

[25] 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. 6

[26] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry,

Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 5

[27] 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. 6

[28] 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. 6

[29] Yih-Ming Su and Jhing-Fa Wang. A novel stroke extraction method for chinese characters using gabor filters. *Pattern Recognition*, 36(3):635–647, 2003. 1, 2

[30] Qiu-Feng Wang, Fei Yin, and Cheng-Lin Liu. Handwritten chinese text recognition by integrating multiple contexts. *IEEE transactions on pattern analysis and machine intelligence*, 34(8):1469–1481, 2011. 1

[31] Tianwei Wang, Zecheng Xie, Zhe Li, Lianwen Jin, and Xiangle Chen. Radical aggregation network for few-shot offline handwritten chinese character recognition. *Pattern Recognition Letters*, 125:821–827, 2019. 1, 2

[32] Wenchao Wang, Jianshu Zhang, Jun Du, Zi-Rui Wang, and Yixing Zhu. Denseran for offline handwritten chinese character recognition. In *2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 104–109. IEEE, 2018. 6, 7

[33] Zi-Rui Wang and Jun Du. Joint architecture and knowledge distillation in cnn for chinese text recognition. *Pattern Recognition*, 111:107722, 2021. 1

[34] Yi-Chao Wu, Fei Yin, Zhuo Chen, and Cheng-Lin Liu. Handwritten chinese text recognition using separable multi-dimensional recurrent neural network. In *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*, pages 79–84. IEEE, 2017.

[35] Yi-Chao Wu, Fei Yin, and Cheng-Lin Liu. Improving handwritten chinese text recognition using neural network language models and convolutional neural network shape models. *Pattern Recognition*, 65:251–264, 2017. 1

[36] Yao Xiao, Dan Meng, Cewu Lu, and Chi-Keung Tang. Template-instance loss for offline handwritten chinese character recognition. In *2019 International conference on document analysis and recognition (ICDAR)*, pages 315–322. IEEE, 2019. 1, 2

[37] Yuhuan Xiu, Qingqing Wang, Hongjian Zhan, Man Lan, and Yue Lu. A handwritten chinese text recognizer applying multi-level multimodal fusion network. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 1464–1469. IEEE, 2019. 1

[38] Chen Yang, Qing Wang, Jun Du, Jianshu Zhang, Changjie Wu, and Jiaming Wang. A transformer-based radical analysis network for chinese character recognition. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 3714–3719. IEEE, 2021. 1, 2

[39] Fei Yin, Qiu-Feng Wang, Xu-Yao Zhang, and Cheng-Lin Liu. Icdar 2013 chinese handwriting recognition competition. In *2013 12th international conference on document analysis and recognition*, pages 1464–1470. IEEE, 2013. 5

[40] 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. 6

[41] Haiyang Yu, Jingye Chen, Bin Li, Jianqi Ma, Mengnan Guan, Xixi Xu, Xiaocong Wang, Shaobo Qu, and Xiangyang Xue. Benchmarking chinese text recognition: Datasets, baselines, and an empirical study. *arXiv preprint arXiv:2112.15093*, 2021. 1, 5

[42] Haiyang Yu, Xiaocong Wang, Bin Li, and Xiangyang Xue. Chinese text recognition with a pre-trained clip-like model through image-ids aligning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11943–11952, 2023. 1, 6

[43] 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*, 34:509–521, 2019. 5

[44] Chuhan Zhang, Ankush Gupta, and Andrew Zisserman. Adaptive text recognition through visual matching. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pages 51–67. Springer, 2020. 1, 2

[45] Jianshu Zhang, Yixing Zhu, Jun Du, and Lirong Dai. Radical analysis network for zero-shot learning in printed chinese character recognition. In *2018 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2018. 1

[46] Jianshu Zhang, Jun Du, and Lirong Dai. Radical analysis network for learning hierarchies of chinese characters. *Pattern Recognition*, 103:107305, 2020. 1, 2

[47] Xu-Yao Zhang, Fei Yin, Yan-Ming Zhang, Cheng-Lin Liu, and Yoshua Bengio. Drawing and recognizing chinese characters with recurrent neural network. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):849–862, 2017. 1

[48] Yuyi Zhang, Yuanzhi Zhu, Dezhi Peng, Peirong Zhang, Zhenhua Yang, Zhibo Yang, Cong Yao, and Lianwen Jin. Hiercode: A lightweight hierarchical codebook for zero-shot chinese text recognition. *Pattern Recognition*, 158:110963, 2025. 1, 2, 3, 5, 6, 7