Group, Contrast and Recognize: A Self-supervised Method for Chinese Character Recognition

Xinzhe Jiang[1], Jun Du[1](✉), Pengfei Hu[1], Mobai Xue[1], Jiefeng Ma[1], Jiajia Wu[2], and Jianshu Zhang[2]

[1] University of Science and Technology of China, Hefei, China
{xzjiang,hudeyouxiang,xmb15,jfma}@mail.ustc.edu.cn, jundu@ustc.edu.cn
[2] iFLYTEK Research, Hefei, China
{jjwu,jszhang6}@iflytek.com

Abstract. Chinese character recognition has been a challenging problem in the field of computer vision, attracting significant research attention due to its widespread applications and technical complexity. However, previous methods rely heavily on manual annotations to guide model learning, without considering self-supervised representation learning. Motivated by the educational approach of teaching pupils to recognize Chinese characters through grouping and differentiation, we introduce a novel self-supervised method that employs clustering and contrastive learning to group similar characters and separate them. Our proposed objective consists of two components: intra-group and inter-group contrastive objectives. The intra-group objective distinguishes the target character from similar characters within the group, while the inter-group objective encourages the model to encode the discriminative semantic structure of each group. The experimental results demonstrate the advantages of our self-supervised representation over previous methods, as well as its superior performance on benchmark comparisons.

Keywords: Chinese character recognition · Self-supervised learning · Contrastive learning · Clustering

1 Introduction

Chinese characters play an irreplaceable role in the transmission of Chinese culture and in the interaction of the Chinese people. Over the years, great efforts have been made to study the problem of Chinese character recognition, and the ability to recognize Chinese characters has become the cornerstone of many commercial applications [20,23].

In the era of deep learning, there are three main categories of Chinese Character Recognition (CCR) methods: character-based ones [27,29,35], radical-based

G. A. Fink et al. (Eds.): ICDAR 2023, LNCS 14190, pp. 411–427, 2023.
https://doi.org/10.1007/978-3-031-41685-9_26

ones [2,24,26], and stroke-based ones [4,37]. These methods vary in terms of their modeling granularity, with character-based methods being the coarsest and stroke-based methods being the finest. Character-based methods treat CCR as a typical classification task, with the goal of finding the category to which each Chinese character image belongs. Radical-based methods, on the other hand, analyze characters by their internal components and present them as a sequence of radicals. Stroke-based methods decompose characters at the stroke level and determine the recognition result through a combination of similarity and edit distance matching. These methods require character, radical, and stroke level annotations as supervision, which require a large amount of labeled data for training. However, labeling these types of data is expensive and time-consuming, making it more cost-effective to find alternatives through self-supervised representation learning without human annotation.

For self-supervised Chinese character recognition, SAE [7] decomposes a character into individual stroke images generated from a predetermined writing sequence. However, this approach only takes printed character images as input and mainly focuses on reconstructing the stroke sequence, ignoring the writing style variations in the real world. In this paper, we present Group, Contrast and Recognize (GCR), a novel self-supervised method for Chinese character recognition that incorporates semantic knowledge from real-world unlabeled character images. The method combines clustering and contrastive learning, drawing inspiration from educational practices where presenting characters in radical-based groups can help beginning learners distinguish and memorize Chinese characters [31].

Typical contrastive learning uses an instance discrimination pre-text task to obtain useful representations by maximizing the agreement between positive pairs and disagreement between negative pairs. However, this approach has two drawbacks: (1) training with easily-distinguishable negative samples can lead to a shortcut solution [30], and (2) the semantic structure of negative samples is neglected [14]. To address these challenges, this paper introduces the hard negative sampling strategy and semantic structure through dynamic clustering during contrastive learning.

First, the method employs the k-means clustering to divide all negative samples in the dictionary queue into different clusters. Second, the proposed model considers both the intra-group comparison and the inter-group comparison as the optimization objective. The intra-group objective requires the model to distinguish the input from hard similar samples within close neighbor clusters, using two augmented views of the input as the positive pair and the input and samples within close neighbor clusters as the negative pairs. The inter-group objective incorporates the semantic structure of negative samples into representation learning by using the input and its closest centroid as the positive pair and the input and other centroids as the negative pairs.

The main contributions of our work are summarized as:

- We introduce a self-supervised method named Group, Contrast and Recognize (GCR) for Chinese character recognition, which leverages both clustering and discrimination to derive meaningful representation from unlabeled data.

- We propose two contrastive objectives, including the intra-group and inter-group objectives, which enhance the model's ability to learn more discriminative representations and better semantic structure.
- Extensive experiments on public benchmarks validate the advantages of GCR, which result in substantial improvements in accuracy compared to supervised baselines.

2 Related Works

2.1 Chinese Character Recognition

The Chinese character recognition problem has been researched for decades. Before the popularity of deep learning, early approaches [3,12,21] used morphology-based observations to obtain hand-crafted features for the CCR task. After that, the deep learning based methods can be categorized into three types: character-based, radical-based, and stroke-based ones. Character-based ones recognize the input image via classification. ATR-CNN [27] proposes relaxation convolution and alternate training to solve the slow convergence and over-fitting problems. DirectMap [35] combines the traditional normalization-cooperated direction-decomposed feature map with the deep convolutional neural network. [29] proposes the template-instance loss functions to alleviate the imbalance problem between easy and difficult character instances. Radical-based ones describe a Chinese character by its internal radicals and structures under the artificial rules. DenseRAN [26] designs an attention-based encoder-decoder model to recognize the radicals and structures of character. FewShotRAN [24] proposes the radical aggregation module to learn robust radical feature and the character analysis decoder to avoid the inflexible match decoding. HDE [2] integrates the tree-based decomposition of Chinese characters into model and learns the compatibility between the input image and the knowledge-based representation. Stroke-based ones adopt the smaller modeling unit and regard the character as a stroke sequence following the writing order. [4] proposes a stroke-based method which decomposes a character into a sequence of five stroke categories, which solves the character zero-shot and radical zero-shot problems. Besides that, it uses a matching-based strategy to acquire the final result in the test stage to overcome the one-to-many problem.

2.2 Self-supervised Contrastive Learning

Recently, self-supervised contrastive learning has achieved success on various vision tasks such as image classification and object detection. It intends to learn an embedding space with alignment and uniformity [25], where two augmentation views of the same instance attract each other while the sample embeddings from different instances are repelled. Specifically, the positive and negative pairs are indispensable for building the contrastive InfoNCE objective [19]. There exists a lot of methods varied with the augmentation and negative sampling strategies. SimCLR [5] generates the instance features within the mini-batch samples, exempt from the requirements of specialized architectures and memory bank. MoCo [8] adopts a momentum-updated encoder as one branch and maintains a dictionary queue of the past instance features. With the projection head and strong augmentation of SimCLR integrated into the vanilla MoCo, MoCo v2 [6] leads to better performance. [28] proposes a ring discrimination method to construct a conditional distribution for hard negative examples, proving the tradeoff between bias and variance. PCL [14] introduces prototypes as latent variables into contrastive learning by the ProtoNCE loss, which can capture high-level semantics. Nevertheless, self-supervised contrastive learning for CCR has rarely been researched.

2.3 Self-supervised Learning for Text Recognition

In order to leverage the potential of unlabeled data, many researchers have turned to self-supervised learning techniques for text recognition. One such method is SeqCLR [1], which is the first self-supervised representation learning approach for text recognition. By dividing the feature map into different instances and conducting sequence-to-sequence contrastive learning, SeqCLR can learn effective self-supervised representations. Another promising approach is PerSec [16], which utilizes dual context perceivers to contrast and learn latent representations from both low-level stroke and high-level semantic contextual spaces simultaneously through hierarchical contrastive learning. Inspired by the reading and writing behaviors of humans, [32] proposes DiG to enhance the performance of text recognition and other text-related tasks. By integrating contrastive learning and masked image modeling, DiG can effectively learn discrimination and generation, ultimately leading to the acquisition of useful representations. These methods are primarily focused on text line recognition rather than isolated Chinese character recognition.

3 Methodology

Our approach adheres to the standard two-stage workflow for self-supervised representation learning, consisting of pre-training and fine-tuning. The fine-tuning stage starts with initializing the encoder with pre-trained backbone weights. In Sect. 3.1, we present our observations and motivation. In Sect. 3.2, we introduce the architecture of the proposed Group, Contrast and Recognize (GCR) method. The intra- and inter-group contrastive objectives are explained in Sect. 3.3 and Sect. 3.4 respectively. Finally, the algorithmic implementation is outlined in Sect. 3.5.

3.1 Observation and Motivation

In exploring the potential of pre-training for CCR, we first aim to examine the distribution of pre-trained features in the latent space. For this purpose, we input a set of labeled character images into the MoCo pre-trained DenseNet encoder, extract their features without fine-tuning, and use k-means clustering to assign the sample features into different clusters based on Euclidean distance. The labels are used for demonstration purposes only and not for training supervision. As shown in Fig. 1, we randomly select some clusters and display their corresponding labels. Our observations are as follows: (1) similar characters with identical components tend to be clustered together, and (2) the major shared component of each cluster is different.

The first observation reveals the fact that it is difficult to distinguish the characters with similar appearances, and the second observation shows the inherent semantics of Chinese characters. As explored in [31], presenting characters with shared radicals in groups can enhance a learner's semantic understanding of Chinese characters. With this in mind, we hypothesize that discrimination among similar characters within a group and among different semantic groups can help the model learn more discriminative features. To achieve this, we leverage the combination of clustering and contrastive learning to mimic the grouping and distinguishing processes.

Clusters	Sample Labels within One Cluster	Major Shared Component
cluster-1	荏 苗 莛 葆 芭 芴 芮 菏 艽 茓 茯 喏 送	艹
cluster-2	湿 淘 澄 浇 渴 溪 浚 淏 谟 璜 躐 缧 逵	氵
cluster-3	捩 拉 拓 拘 掊 捉 相 稍 秋 斌 炉	扌
cluster-4	锨 锡 镇 铌 镔 锭 锯 锐 辗 踉 玻	钅
cluster-5	桩 枷 枳 枸 权 积 忙 恢 敉 菥	木

Fig. 1. The sample labels of some clusters.

3.2 Architecture

The architecture of our proposed GCR is depicted in Fig. 2. The input image x is augmented to create two views, x_a and x_b, which are then processed by the query encoder and the momentum key encoder, with the query instance q and key instance k obtained. The query encoder f_q consists of the backbone $F(\cdot)$ and the projection head $P(\cdot)$, and the momentum key encoder f_k consists of the $F_m(\cdot)$ and $P_m(\cdot)$. θ_q and θ_k are the parameters of the query encoder f_q and momentum key encoder f_k, respectively. Additionally, a dictionary queue Q is maintained, where the encoded momentum representations of the current batch are stored, and the oldest are removed. Finally, the acquired clusters and instance features in the dictionary queue are utilized to achieve both intra-group and inter-group contrastive learning.

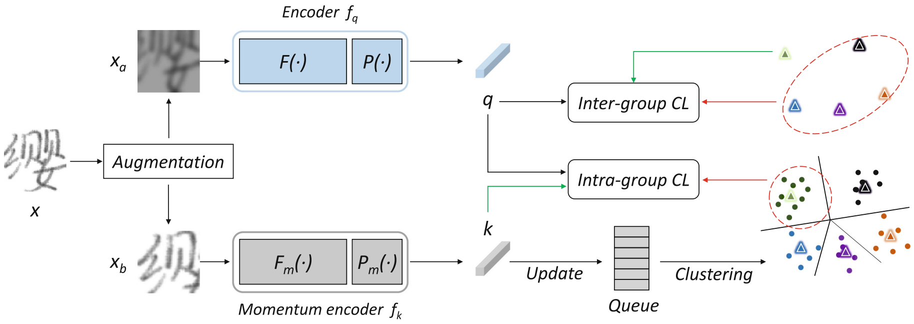


Fig. 2. Architecture of Group, Contrast, and Recognize (GCR). The green lines and red lines indicate the source of positive and negative samples in contrastive learning, respectively. CL is the abbreviation for contrastive learning. The triangles with various colors represent the cluster centroids. (Color figure online)

3.3 Intra-group Contrastive Learning

A long-standing issue in CCR is the tendency of similar characters to be easily confused. This is due to their encoded features often being close in the embedding space, making it difficult for predictors to correctly recognize them. This mirrors the common experience of beginning learners who frequently struggle to distinguish among similar characters. To address this, we sample hard negatives for contrastive learning in order to magnify the differences among similar characters. The key point is how to effectively sample the required hard negatives.

Based on the first observation in Sect. 3.1, we introduce the concept of intra-group contrastive learning, as depicted in Fig. 2. During model pre-training, we dynamically cluster the momentum representations of instances in the dictionary queue into M clusters. With these clusters, we divide the negatives into M subsets $\mathbf{C} = \{\mathbf{C}_1, \mathbf{C}_2, \ldots, \mathbf{C}_M\}$. The proposed intra-group contrastive loss is given by:

$$\mathcal{L}_{\text{intra}} = -\log \frac{\exp(q \cdot k/\tau)}{\exp(q \cdot k/\tau) + \sum_{\mathbf{C}_i \subset \mathbf{G}_q} \sum_{k_j \in \mathbf{C}_i} \exp(q \cdot k_j/\tau)} \tag{1}$$

where τ is the temperature, and $\mathbf{G}_q$ is the sampled group that includes the hard negatives. We choose the union of the top-D closest clusters as the hard negatives group $\mathbf{G}_q$. Notably, $\mathbf{G}_q$ excludes the closest cluster, since the closest cluster probably contains the identical instances of q, which are the false negatives and cause the model to discard semantic information [11].

In this section, we distinguish the query instance from the similar instances in the hard negatives group to ensure local uniformity in the embedding space. By sampling hard negatives, the model receives more discriminative information, leading to improved representation.

3.4 Inter-group Contrastive Learning

In this section, we aim to ensure the distinction among different semantic groups by enlarging the separation among all cluster centroids. To achieve this, we introduce an inter-group contrastive loss. This loss minimizes the distance between the query instance and its corresponding centroid in the embedding space, while pushing other centroids away.

In detail, we assign one centroid to each query instance and calculate the inter-group contrastive loss as follows:

$$\mathcal{L}_{\text{inter}} = -\log \frac{\exp\left(q \cdot c_q^{\text{s}} / \tau\right)}{\sum_{c_j^{\text{s}} \in \mathbf{C}^{\text{s}}} \exp\left(q \cdot c_j^{\text{s}} / \tau\right)} \tag{2}$$

where c_q^{s} is the closest centroid to the query q, c_j^{s} is the cluster centroid of $\mathbf{C}_j$ and $\mathbf{C}^{\text{s}}$ is the union set of c_j^{s}. Note that the centroid embedding is calculated as the average of the instance embeddings within the cluster. With this objective, we aim to encourage global uniformity in the embedding space by treating each cluster as a single group.

3.5 Network Training

The procedure for self-supervised pre-training of the GCR framework is outlined in Algorithm 1. Unlike DnC [22], our clustering process is integrated seamlessly into the contrastive learning process, instead of being separated into several steps. In addition to the intra- and inter-group objectives, we also incorporate the vanilla InfoNCE loss as formulated in Eq. 3, to ensure local smoothness and support the clustering bootstrapping, following the strategy of PCL [14].

$$\mathcal{L}_{\text{infonce}} = -\log \frac{\exp\left(q \cdot k / \tau\right)}{\exp\left(q \cdot k / \tau\right) + \sum_{k_j \in Q} \exp\left(q \cdot k_j / \tau\right)} \tag{3}$$

The final loss function is a combination of all these objectives, formulated as follows:

$$\mathcal{L}_{\text{total}} = \lambda_1 \mathcal{L}_{\text{infonce}} + \lambda_2 \mathcal{L}_{\text{intra}} + \lambda_3 \mathcal{L}_{\text{inter}} \tag{4}$$

where λ_1, λ_2, λ_3 are coefficients that control the contribution of each part to the total loss.

4 Experiments

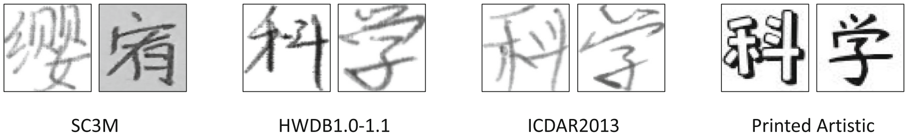
In this section, we outline the experimental setup, including details on the datasets, baseline models, and implementation specifications. Subsequently, we perform extensive experiments on benchmark datasets to evaluate the GCR from both qualitative and quantitative perspectives.

Algorithm 1 Main algorithm of GCR

```
Input unlabeled image $x$, temperature $\tau$, mini-batch size $N$, query encoder $f_q$, key encoder $f_k$, momentum coefficient $m$, number of desired clusters $M$, clustering interval $r$, total training steps $s$, loss coefficients $\lambda_1$, $\lambda_2$ and $\lambda_3$
Randomly initialize parameters $\theta_q$ and $\theta_k$, $\theta_q = \theta_k$
Randomly initialize the queue $Q$ of negative instances $k_j$
for $step \in s$ do
    if $step\%r == 0$ then
        $\mathbf{C}^{\mathrm{s}}$,   $\mathbf{C} \leftarrow$ K-means Clustering on $Q$ for $M$ clusters
    end if
    for $x \in$ mini-batch do
        $x_a = Aug_1(x)$
        $q = f_q(x_a)$
        $x_b = Aug_2(x)$
        $k = f_k(x_b)$
        $\mathcal{L}_{\text{intra}} = -\log \frac{\exp(q\cdot k/\tau)}{\exp(q\cdot k/\tau)+\sum_{\mathbf{C}_i \subset \mathbf{G}_q}\sum_{k_j \in \mathbf{C}_i}\exp\left(q\cdot k_j/\tau\right)}$
        $\mathcal{L}_{\text{inter}} = -\log \frac{\exp\left(q\cdot c^{\mathrm{s}}_q/\tau\right)}{\sum_{c^{\mathrm{s}}_j \in \mathbf{C}^{\mathrm{s}}}\exp\left(q\cdot c^{\mathrm{s}}_j/\tau\right)}$
        $\mathcal{L}_{\text{infonce}} = -\log \frac{\exp(q\cdot k/\tau)}{\exp(q\cdot k/\tau)+\sum_{k_j \in Q}\exp\left(q\cdot k_j/\tau\right)}$
        $\mathcal{L}_{\text{total}} = \lambda_1\mathcal{L}_{\text{infonce}} + \lambda_2\mathcal{L}_{\text{intra}} + \lambda_3\mathcal{L}_{\text{inter}}$
    end for
    update $f_q$ by back-propagation
    update $f_k$ with momentum from $f_q$:   $\theta_k \leftarrow m\theta_k + (1-m)\theta_q$
    enqueue the keys $k$ to $Q$
    dequeue the oldest keys
end for
```

4.1 Datasets

We utilize a collection of 3 million scanned and camera images of handwritten Chinese characters for pre-training, which we name the SC3M dataset. For fine-tuning, we use the HWDB1.0-1.1 dataset [15], consisting of 2.73 million offline handwritten Chinese character images from 720 writers. To evaluate the performance of the GCR framework, we conduct experiments on the ICDAR2013 benchmark [33], which includes 224,419 offline handwritten Chinese characters from 60 writers with 3755 classes. We also evaluate the model's ability to recognize printed artistic characters using the Printed Artistic dataset [4], which



SC3M

HWDB1.0-1.1

ICDAR2013

Printed Artistic

Fig. 3. Some examples in the datasets.

contains 3755 characters in 105 printed artistic fonts. An illustration of some examples from these datasets can be seen in Fig. 3.

4.2 Baselines

In this work, we have constructed three baseline models for CCR, including character-level, radical-level, and stroke-level models. Our first model, called DenseClassifier, is a character-level model that combines a CNN encoder with a linear classifier. The second model, RAN, is a radical-level method that utilizes an encoder-decoder architecture with a coverage attention mechanism. Finally, our stroke-level model, SLD, is comprised of an image-to-feature encoder and a feature-to-stroke decoder that employs a matching-based strategy. Both the DenseClassifier and RAN models use a modified DenseNet [10] as their backbone for feature extraction, while SLD uses a modified ResNet [9]. The character accuracy is employed as the evaluation metric for the downstream CCR task.

4.3 Implementation Details

In the pre-training stage, we follow the configuration of the vanilla MoCo v2 [6] and apply random crop, random color jittering, random grayscale conversion, and random Gaussian blur. The optimization algorithm used is SGD with a momentum of 0.9, a weight decay of 0.0001, and a batch size of 3200. The temperature τ is set to 0.2, the queue size K to 65536, the number of clusters M to 1500, the number of hard negative clusters D to 5, the momentum coefficient m to 0.999, and the clustering interval steps r to 30. The coefficients for each loss λ_1, λ_2, λ_3 are set to 1, 0.5, and 0.5, respectively. The dynamic k-means clustering is implemented using the efficient faiss tool [13]. The initial learning rate is set to 0.03 and adjusted using a cosine scheduler. The experiments are run on 16 NVIDIA Tesla V100 (24GB RAM) GPUs.

In the fine-tuning stage, we use the plateau scheduler and Adadelta optimizer with an initial learning rate of 0.0001, a weight decay of 0.0001, and a batch size of 96 for the DenseClassifier and RAN. For the SLD model, the Adadelta optimizer is used with an initial learning rate of 1.0 and a weight decay of 0.0001. The input images for DenseClassifier and RAN are resized to 64 × 64, while the input for SLD is resized to 32 × 32. The experiments are conducted on 4 NVIDIA Tesla V100 (12GB RAM) GPUs.

4.4 Representation Quality of Self-supervised Pre-training

In order to assess the impact of self-supervised learning on representation quality in CCR, we use DenseClassifier as our baseline model and carry out experiments with different pre-text tasks for pre-training. The representation quality is evaluated by freezing the weights of the pre-trained encoder and training a randomly initialized linear layer on the entire HWDB1.0-1.1 dataset, followed by testing on ICDAR2013. The results, as shown in Table 1, demonstrate that incorporating prior knowledge from the pre-text tasks can improve the representation quality and overall performance of the model. Among the various approaches, our proposed GCR, which combines contrastive learning and clustering, achieves the best result and outperforms the MoCo method by 5.19%.

Table 1. Performance comparison in the frozen setting of different pre-training methods. 'None' means the encoder is randomly initialized and frozen, with the single linear classifier trained.

Pre-train Method	None	Jigsaw [18]	MoCo [8]	GCR
Accuracy	0.05%	74.91%	78.90%	**84.09%**

To evaluate the performance of self-supervised pre-training in low-resource scenarios, we conduct N-shot experiments where the training set includes N images per character. The pre-trained encoder is utilized for initialization and then fine-tuned. As shown in Table 2, the results indicate that self-supervised methods are capable of improving model performance when training data is limited. Our GCR method consistently enhances the supervised baseline performance, outperforming the Jigsaw and MoCo methods when N is set to 1, 3, 5, and 10. As N increases to 10, the performance gain of pre-training methods reaches a limit.

Table 2. Performance comparison in N-shot setting of different pre-training methods. 'None' means the encoder is randomly initialized and trained, i.e., supervised baseline.

Pre-train Method	1-shot	3-shot	5-shot	10-shot
None	0.10%	17.32%	67.54%	91.38%
Jigsaw [18]	7.81%	69.84%	88.03%	93.10%
MoCo [8]	8.52%	73.49%	89.11%	93.16%
GCR	**17.18%**	**79.61%**	**90.10%**	**93.38%**

To further demonstrate the discrimination power of GCR, we conduct an experiment using two sets of similar characters with 60 images per character, which are selected from ICDAR2013. We utilize the pre-trained encoder to extract the self-supervised features and average-pool them into vectors. These vectors are then embedded into a 2-D space using t-SNE visualization [17]. As shown in Fig. 4, each color represents a different character, with the shared component of the top and bottom rows being 'kou' and 'zou' respectively. Our results indicate that compared to MoCo and Jigsaw, GCR provides more discriminative features for similar characters, resulting in better cluster separation.

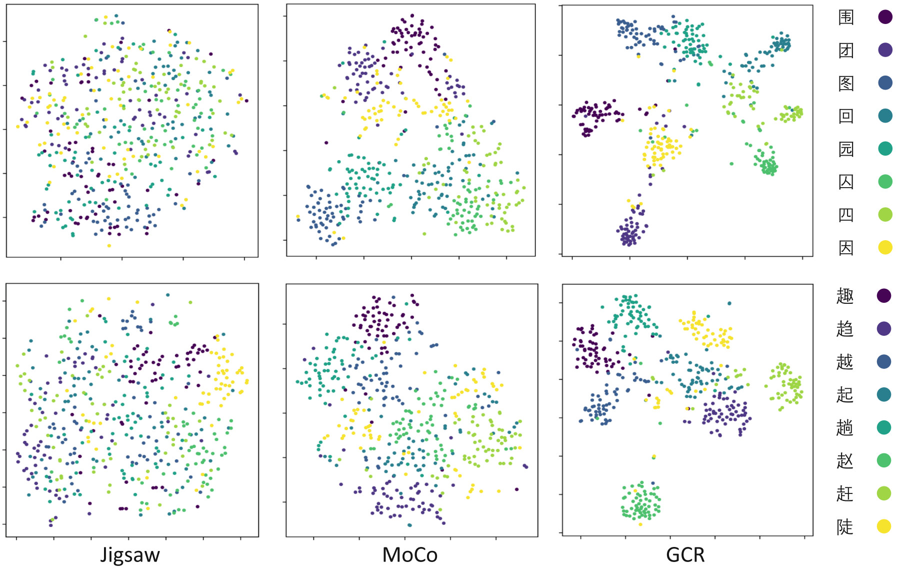


Fig. 4. T-SNE visualization of the self-supervised learned representation of two sets of similar characters. Left: Jigsaw; Middle: MoCo; Right: GCR (ours). Colors represent character classes.

4.5 Handwritten Benchmark Comparison in Zero-Shot Setting

Performance Comparison: We conduct experiments on handwritten characters in the zero-shot setting. We fine-tune the RAN and SLD baseline models. For the training set, we select the first m classes of 3755 characters from HWDB1.0-1.1, where m ranges in $\{500, 1000, 1500, 2000, 2755\}$. The test set consists of samples with labels from the last 1000 classes of the ICDAR2013 dataset. Note

Table 3. Performance comparison in the character zero-shot setting on the handwritten benchmark.

Handwritten	Pre-train	Character Zero-Shot Setting				
		500	1000	1500	2000	2755
DenseRAN [26]	None	1.70%	8.44%	14.71%	19.51%	30.68%
HDE [2]	None	4.90%	12.77%	19.25%	25.13%	33.49%
ACPM [37]	None	9.72%	18.50%	27.74%	34.00%	42.43%
RAN [34]	None	2.65%	10.10%	16.92%	21.56%	31.78%
	MoCo [8]	2.96%	10.14%	17.78%	21.86%	32.59%
	GCR	**3.99%**	**10.17%**	**18.67%**	**23.59%**	**33.29%**
SLD [4]	None	5.60%	13.85%	22.88%	25.73%	37.91%
	SAE [7]	5.91%	14.35%	24.32%	30.17%	40.22%
	MoCo [8]	5.70%	16.60%	24.62%	29.47%	38.90%
	GCR	**6.45%**	**21.03%**	**28.11%**	**33.00%**	**42.01%**

that our partition method is the same as that used in the SLD for a fair comparison.

The results of our experiments are summarized in Table 3. In the case of RAN, both MoCo and GCR are able to improve the baseline performance. For SLD, GCR makes a substantial improvement to the baseline accuracy across all partition settings, outperforming MoCo and SAE. The success of GCR can be attributed to its ability to capture more discriminative details and distinguish similar characters, which helps SLD perform better. Overall, GCR achieves the best results compared to both supervised baselines and other self-supervised methods, demonstrating its superiority.

Table 4. Ablation study on each part of pre-training objectives.

InfoNCE loss	Inter-group loss	Intra-group loss	Accuracy
			37.91%
✓			38.90%
✓	✓		40.60%
✓		✓	41.17%
✓	✓	✓	42.01%

Ablation Study: Since the total loss consists of three parts, it is necessary to investigate whether the proposed intra-group and inter-group contrastive loss can improve the capability of feature representation. To this end, we conduct experiments on the SLD baseline model under the zero-shot partition 2755.

The results, presented in Table 4, show that incorporating the vanilla InfoNCE loss into the SLD model leads to an improvement of 0.99%. By removing the inter-group loss, the fine-tuning accuracy decreases by 0.84% (from 42.01% to 41.17%). The removal of the intra-group loss results in a decrease of 1.41% in fine-tuning accuracy (from 42.01% to 40.60%). Our experiments reveal that the intra-group loss has a more significant impact than the inter-group loss, as the former is designed to distinguish similar characters, which is more crucial for unseen character recognition. Finally, when both the intra-group and inter-group losses are employed, the accuracy improves by 3.11% (from 38.90% to 42.01%), further confirming their advantages.

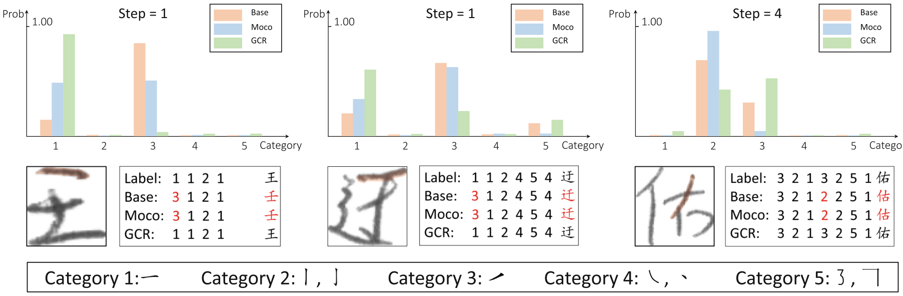


Fig. 5. Case Study. The bar charts depict the probabilities of predictions for each category of strokes, excluding the categories ‘sos’ and ‘eos’. A single category comprises multiple instances of strokes. The utilized SLD model decomposes a character into a series of stroke categories. The red bold number signifies an incorrect recognition result, which is represented by the red area in the image. (Color figure online)

Qualitative Analysis: As seen in Fig. 5, we can qualitatively observe how the proposed GCR captures the detail information and correctly recognize the unseen characters, compared with the baseline SLD model and MoCo. The cases are from the zero-shot partition 2755 experiment. Taking the left ‘wang’ as an example, the confusing region is wrongly recognized as category 3 by the baseline and MoCo at the first decoding step, and the final result is ‘ren’ which is similar to the character ‘wang’ and has appeared in the training set. However, our GCR can correctly recognize it with high confidence, which suggests the capability of GCR to distinguish similar characters.

4.6 Printed Artistic Benchmark Comparison in Zero-Shot Setting

Besides the handwritten characters, we also conduct experiments with printed artistic characters in the zero-shot setting. The dataset is Printed Artistic and

Table 5. Performance comparison in the character zero-shot setting on the Printed Artistic benchmark.

Printed Artistic	Pre-train	Character Zero-Shot Setting				
		500	1000	1500	2000	2755
DenseRAN [26]	None	0.20%	2.26%	7.89%	10.86%	24.80%
HDE [2]	None	7.48%	21.13%	31.75%	40.43%	51.41%
RAN [34]	None	0.83%	19.13%	28.49%	43.57%	56.85%
	MoCo [8]	4.55%	22.19%	30.20%	45.80%	57.10%
	GCR	**7.12%**	**24.11%**	**31.24%**	**48.25%**	**59.35%**
SLD [4]	None	7.03%	26.22%	48.42%	54.86%	65.44%
	SAE [7]	8.25%	32.24%	50.72%	57.13%	68.88%
	MoCo [8]	10.81%	36.50%	53.85%	60.56%	69.22%
	GCR	**11.85%**	**41.14%**	**55.46%**	**63.04%**	**70.69%**

the partition manner is the same as that of SLD. The SLD fine-tuned from GCR outperforms not only the supervised baselines and self-supervised methods, but also other previous methods, as shown in Table 5. Compared with handwritten characters, printed artistic characters have more clear strokes and fixed writing styles relatively, which are easier to be correctly recognized.

Table 6. The results in seen character setting on ICDAR2013.

Method	Decomposition	Accuracy
HCCR-GoogLeNet [36]	Character	96.35%
DirectMap+ConvNet+Adaptation [35]	Character	97.37%
DenseRAN [26]	Radical	96.66%
FewShotRAN [24]	Radical	96.97%
HDE [2]	Radical	97.14%
template+instance [29]	Character	97.45%
SLD [4]	Stroke	96.74%
ACPM [37]	All	97.80%
RAN [34]	Radical	96.61%
RAN+MoCo	Radical	96.67%
RAN+GCR	Radical	96.79%
DenseClassifier	Character	97.23%
DenseClassifier+MoCo	Character	97.34%
DenseClassifier+GCR	Character	**97.51%**

4.7 Handwritten Benchmark Comparison in Seen Setting

The results of our experiments under the seen character setting are presented in Table 6. In line with previous studies, we use the ICDAR2013 dataset as the test set, where all labels have appeared in the training set HWDB1.0-1.1, without any zero-shot challenge. Our results indicate that the RAN model benefits from both MoCo and GCR, with accuracy improvements of 0.06% and 0.18% respectively over the baseline. Similarly, the DenseClassifier model shows improvements with MoCo and GCR, yielding accuracy improvements of 0.11% and 0.28% respectively. Notably, the DenseClassifier fine-tuned from GCR is only second to the state-of-the-art model ACPM which incorporates multi-level decomposition information.

5 Conclusion and Future Work

In this paper, we propose GCR, a novel self-supervised method for CCR. By combining clustering and contrastive learning, and optimizing the proposed inter-group and intra-group contrastive objectives, GCR significantly enhances the representation ability compared to the baseline model and other self-supervised methods. Consequently, our GCR achieves obvious performance improvements on the benchmark datasets ICDAR2013 and Printed Artistic. The key takeaway is that the hard similar negatives and semantic structure of the unlabeled data can be utilized to obtain useful self-supervised representations for the downstream CCR task. In the future, we will further evaluate the generalization capability of GCR for other languages, such as Korean.

References

1. Aberdam, A., et al.: Sequence-to-sequence contrastive learning for text recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 15297–15307 (2021)
2. Cao, Z., Lu, J., Cui, S., Zhang, C.: Zero-shot handwritten Chinese character recognition with hierarchical decomposition embedding. Pattern Recogn. **107**, 107488 (2020)
3. Chang, F.: Techniques for solving the large-scale classification problem in chinese handwriting recognition. In: Proceedings of the 2006 Conference on Arabic and Chinese Handwriting Recognition, pp. 161–169 (2006)
4. Chen, J., Li, B., Xue, X.: Zero-shot Chinese character recognition with stroke-level decomposition. In: Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, pp. 615–621. International Joint Conferences on Artificial Intelligence Organization (2021)
5. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International Conference on Machine Learning, pp. 1597–1607. PMLR (2020)
6. Chen, X., Fan, H., Girshick, R., He, K.: Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297 (2020)
7. Chen, Z., Yang, W., Li, X.: Stroke-based autoencoders: self-supervised learners for efficient zero-shot Chinese character recognition. Appl. Sci. **13**(3), 1750 (2023)
8. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9726–9735 (2020)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
10. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2261–2269 (2017)
11. Huynh, T., Kornblith, S., Walter, M.R., Maire, M., Khademi, M.: Boosting contrastive self-supervised learning with false negative cancellation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 986–996 (2022)
12. Jin, L.W., Yin, J.X., Gao, X., Huang, J.C.: Study of several directional feature extraction methods with local elastic meshing technology for HCCR. In: Proceedings of the Sixth International Conference for Young Computer Scientist, pp. 232–236 (2001)
13. Johnson, J., Douze, M., Jégou, H.: Billion-scale similarity search with GPUs. IEEE Trans. Big Data **7**(3), 535–547 (2021)

14. Li, J., Zhou, P., Xiong, C., Hoi, S.: Prototypical contrastive learning of unsupervised representations. In: International Conference on Learning Representations (2021)
15. Liu, C.L., Yin, F., Wang, D.H., Wang, Q.F.: Online and offline handwritten Chinese character recognition: benchmarking on new databases. Pattern Recogn. **46**(1), 155–162 (2013)
16. Liu, H., et al.: Perceiving stroke-semantic context: hierarchical contrastive learning for robust scene text recognition. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, pp. 1702–1710 (2022)
17. Van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. J. Mach. Learn. Res. **9**, 2579–2605 (2008)
18. Noroozi, M., Favaro, P.: Unsupervised learning of visual representations by solving jigsaw puzzles. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9910, pp. 69–84. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46466-4_5
19. Oord, A.v.d., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)
20. Qian, R., Zhang, B., Yue, Y., Wang, Z., Coenen, F.: Robust Chinese traffic sign detection and recognition with deep convolutional neural network. In: 11th International Conference on Natural Computation (ICNC), pp. 791–796. IEEE (2015)
21. Su, Y.M., Wang, J.F.: A novel stroke extraction method for Chinese characters using gabor filters. Pattern Recogn. **36**(3), 635–647 (2003)
22. Tian, Y., Hénaff, O.J., Oord, A.v.d.: Divide and contrast: self-supervised learning from uncurated data. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10043–10054 (2021)
23. Wang, J., et al.: Towards robust visual information extraction in real world: new dataset and novel solution. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 2738–2745 (2021)
24. Wang, T., Xie, Z., Li, Z., Jin, L., Chen, X.: Radical aggregation network for few-shot offline handwritten Chinese character recognition. Pattern Recogn. Lett. **125**, 821–827 (2019)
25. Wang, T., Isola, P.: Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In: International Conference on Machine Learning, pp. 9929–9939. PMLR (2020)
26. Wang, W., Zhang, J., Du, J., Wang, Z.R., Zhu, Y.: DenseRAN for offline handwritten Chinese character recognition. In: 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), pp. 104–109. IEEE (2018)
27. Wu, C., Fan, W., He, Y., Sun, J., Naoi, S.: Handwritten character recognition by alternately trained relaxation convolutional neural network. In: 14th International Conference on Frontiers in Handwriting Recognition, pp. 291–296. IEEE (2014)
28. Wu, M., Mosse, M., Zhuang, C., Yamins, D., Goodman, N.: Conditional negative sampling for contrastive learning of visual representations. In: International Conference on Learning Representations (2021)
29. Xiao, Y., Meng, D., Lu, C., Tang, C.K.: Template-instance loss for offline handwritten chinese character recognition. In: International Conference on Document Analysis and Recognition (ICDAR), pp. 315–322. IEEE (2019)
30. Xie, J., Zhan, X., Liu, Z., Ong, Y.S., Loy, C.C.: Delving into inter-image invariance for unsupervised visual representations. Int. J. Comput. Vision **130**(12), 2994–3013 (2022)

31. Xu, Y., Chang, L.Y., Perfetti, C.A.: The effect of radical-based grouping in character learning in Chinese as a foreign language. Mod. Lang. J. **98**(3), 773–793 (2014)
32. Yang, M., et al.: Reading and writing: discriminative and generative modeling for self-supervised text recognition. In: Proceedings of the 30th ACM International Conference on Multimedia, pp. 4214–4223 (2022)
33. Yin, F., Wang, Q.F., Zhang, X.Y., Liu, C.L.: ICDAR 2013 Chinese handwriting recognition competition. In: 12th International Conference on Document Analysis and Recognition, pp. 1464–1470. IEEE (2013)
34. Zhang, J., Du, J., Dai, L.: Radical analysis network for learning hierarchies of Chinese characters. Pattern Recogn. **103**, 107305 (2020)
35. Zhang, X.Y., Bengio, Y., Liu, C.L.: Online and offline handwritten Chinese character recognition: a comprehensive study and new benchmark. Pattern Recogn. **61**, 348–360 (2017)
36. Zhong, Z., Jin, L., Xie, Z.: High performance offline handwritten Chinese character recognition using GoogLeNet and directional feature maps. In: 13th International Conference on Document Analysis and Recognition (ICDAR), pp. 846–850. IEEE (2015)
37. Zu, X., Yu, H., Li, B., Xue, X.: Chinese character recognition with augmented character profile matching. In: Proceedings of the 30th ACM International Conference on Multimedia, pp. 6094–6102 (2022)