REZCR: A Zero-shot Character Recognition Method via Radical Extraction

Xiaolei Diao\textsuperscript{1,2}, Daqian Shi\textsuperscript{2}, Hao Tang\textsuperscript{3}, Lei Wu\textsuperscript{1}, Yanzeng Li\textsuperscript{1}, Hao Xu\textsuperscript{1}

\textsuperscript{1} Jilin University, China
\textsuperscript{2} the University of Trento, Italy
\textsuperscript{3} ETH Zurich, Switzerland

Abstract

The long-tail effect is a common issue that limits the performance of deep learning models on real-world datasets. Character image dataset development is also affected by such unbalanced data distribution due to differences in character usage frequency. Thus, current character recognition methods are limited when applying to real-world datasets, in particular to the character categories in the tail which are lacking training samples, e.g., uncommon characters, or characters from historical documents. In this paper, we propose a zero-shot character recognition framework via radical extraction, i.e., REZCR, to improve the recognition performance of few-sample character categories, in which we exploit information on radicals, the graphical units of characters, by decomposing and reconstructing characters following orthography. REZCR consists of an attention-based radical information extractor (RIE) and a knowledge graph-based character reasoner (KGR). The RIE aims to recognize candidate radicals and their possible structural relations from character images. The results will be fed into KGR to recognize the target character by reasoning with a pre-designed character knowledge graph. We validate our method on multiple datasets, REZCR shows promising experimental results, especially for few-sample character datasets.

Introduction

Developments in Optical Character Recognition (OCR) technology offer new solutions for learning, managing, and utilizing character resources. Current OCR methods (Zhang, Bengio, and Liu 2017) are mainly based on deep learning models, which place high demands on the quantity and quality of data. Due to differences in character usage frequency, the long-tail effect exists as a common issue in character datasets. Such character dataset often contains categories with few samples, especially for datasets with un producible or inaccessible samples, such as historical character (Huang et al. 2019) and calligraphic character (Lyu et al. 2017) dataset. Fig. 1(a) shows the distribution of character samples for each category in the oracle bone dataset (Wu 2012). The distribution demonstrates a significant long-tail effect, where over half of the character categories have five samples or fewer, and there are also categories with a single sample. Therefore, current OCR methods achieve limited performance on such datasets (Anderson 2006), especially when recognizing character categories with few samples.

Figure 1: Analysis of character datasets; (a) statistics of each character category in (Wu 2012); (b) examples of oracle bone character categories with the unique sample; (c) examples of simplified Chinese characters; (d) examples of Korean characters. Identical radicals are marked with a box of the same colour.

A common solution to alleviate the limitations caused by long-tail effects is employing data enhancement methods to balance data categories, including re-sampling (Mahajan et al. 2018) and re-weighting (Cui et al. 2019). However, these methods are rarely applied to data categories with few samples or a single sample for preventing overfitting. Although the compromise solution tends to discard categories with few samples (Géron 2019), we consider such categories are valuable and irreplaceable for real-world tasks. For example, as shown in Fig. 1(b), we have only one sample for each character on unearthed oracle bone, which is the unique evidence for understanding this character by archaeologists. Thus, it is a challenge to achieve character recognition on few-sample datasets while keeping data integrity.

The orthography-based strategy of learning East Asian characters in terms of radicals and character structure is the
most frequent and effective way for human beings (Shen 2005). We observe the orthographic peculiarities of East Asian characters: characters are composed with a set of radicals organized by a specific structure, and a radical can be shared by different characters (Myers 2016). Fig. 1(b), (c) and (d) show our observation on oracle bone, Chinese and Korean character datasets, where we highlight identical radicals by boxes with specific colours. Statistically, 6,763 Chinese characters can be represented by 485 radicals and 10 character structures in a Chinese character dataset (Liu et al. 2011). In Fig. 1(a), we demonstrate some character samples that contain the identical radical "yang" (in red boxes), where these samples tend to distribute randomly in the statistics. This offers the possibility to learn from enough radical samples and to alleviate the problem of insufficient training samples for the characters. Therefore, we intend to utilize the information of radicals to address the problem of identifying character categories with few samples by decomposing and reconstructing characters following orthography.

Based on the above discussion, we propose a novel framework for zero-shot character recognition via radical extraction, called REZCR (see Fig. 3). This framework consists of an attention-based radical information extractor (RIE) and a knowledge graph-based character reasoner (KGR). The former aims to extract candidate radicals and their possible structural relationships by a radical attention blocks (RAB)-based model, which is introduced to mitigate the interference by background. The results of RIE will be fed into the KGR, where a weight fusion-based algorithm is proposed to achieve zero-shot character recognition by reasoning with a pre-designed character knowledge graph (CKG). The CKG organizes relations between radicals, structures, and characters under the guidance of linguists.

The main contributions are summarized as follows:

- We propose a novel zero-shot character recognition framework, namely REZCR, which exploits radical extraction and knowledge graph reasoning to achieve character recognition on few-sample categories.
- We introduce RIE to extract comprehensive radical information, where we propose RAB to mitigate background interference for obtaining candidate radicals, and feature combination to extract the structural relations between radicals.
- Our method is validated on multiple datasets, including a newly constructed dataset, namely OracleRC. REZCR achieves promising results on character recognition, especially outperforming state-of-the-art methods on few-sample cases.

Related Work

Character Recognition

Character-based Methods. Early work on character recognition is divided into feature extraction-based methods (Su and Wang 2003) and statistical-based methods (Shanthi and Duraiswamy 2010). CNNs were gradually applied to character recognition tasks with the development of deep learning. (Yuan et al. 2012) applied an improved LeNet-5 CNN model to recognise offline English characters. As the first method to apply CNNs into Chinese character recognition, MCDNN (Ciresan and Meier 2015) was trained by an embedding framework with eight networks and obtained human-comparable results. Furthermore, DirectMap (Zhang, Bengio, and Liu 2017) combines a direction-decomposed feature map with a deep CNN for training, and introduces an adaptation layer to reduce the mismatch between training and test data.

Radical-based Methods. The huge number of Chinese character categories and the reliance of the above-described methods on a large number of training samples limit the accuracy of character-based methods. Researchers have introduced domain knowledge related to Chinese characters and radical-based methods to overcome this issue. RAN (Zhang et al. 2018), Zhang, Du, and Dai (2020) and DenseRAN (Wang et al. 2018) use a CNN-based encoder to extract the radicals and their two-dimensional structure, and then generate radical sequences that represent characters by an RNN-based decoder for Chinese character recognition. HDE (Cao et al. 2020) recognises Chinese characters by performing vector matching in embedding space, introducing radical-level constitution to design a unique embedding vector for each Chinese character. (Chen, Li, and Xue 2021) decomposes Chinese characters into strokes, the smallest units in Chinese characters, then recognises characters by looking up stroke sequences in a dictionary.

The common idea of the above methods is to transform each character as a sequence of radicals or strokes. As the prerequisite of these methods, the alphabet needs to be present in advance, which contains each character and its corresponding sequence representation. The length of each sequence is not fixed, the maximum length can achieve 20 when the target character is complex. Thus, new challenges appear when applying the above methods. To obtain a correct recognition result, the generated sequence is required to be matched with the alphabet completely, both for each element and their order. That means the recognition performance will be significantly limited when applying to characters with a long sequence.

Zero-shot Learning

Zero-shot learning aims to classify unseen samples by learning from existing samples and knowledge supplemented by additional information (Wang et al. 2019b). In recent years, zero-shot learning has been considered from a variety of perspectives. For instance, embedding-based methods (Akata et al. 2015, Xie et al. 2019) propose to perform image embedding and label semantic embedding in the same space, and extend to unseen categories by a compatibility function. Furthermore, attribute-based methods (Lampert, Nickisch, and Harmeling 2009, 2013) manually design attributes in different ways and represent categories with multidimensional attribute vectors, which are introduced to train the visual classifier by vector mapping. Additionally, knowledge graph-based methods (Rohrbach, Stark, and Schiele 2011) Wang, Ye, and Gupta (2018) apply knowledge graphs as additional information to guide zero-shot learning, as the knowledge graph has the ability to convey categories and
the relationships between categories. In this paper, we intend to exploit knowledge graphs for providing additional information in the framework, since it is more explanatory than embedding-based methods and express structural features to the additional information compared to attribute-based methods. We believe this will bring crucial advantages in character recognition, especially for dealing with unreproducible characters (e.g., oracle bone or calligraphic characters) with historical and literary value.

The Proposed REZCR

The intuition of our method is to exploit radicals, one of the basic features to learn a character, for knowledge graph-based reasoning to achieve character recognition on few-sample categories. As a result, we propose a zero-shot learning framework that decomposes characters into radicals and their internal structural relations, aiming to satisfy demands on the model training and achieve high recognition performance on few-sample or unseen characters. In this section, we firstly discuss the reason for utilizing radical information and its superiority for recognizing characters. Then, we present a detailed description of the proposed zero-shot character recognition framework, namely REZCR, and its two main components, radical information extractor (RIE) and knowledge graph reasoner (KGR).

Problem Formulation

Most alphabet-based languages are only composed of a few dozen characters, while some other languages have a larger number of characters: for example, the total number of recorded Chinese characters exceeds 80,000 (Feng 2014). Such large number of categories is difficult to be managed for the OCR task. In fact, characters are not the smallest unit in such language systems, since they can be decomposed into radicals and even strokes. A Chinese character is shown in Fig. 2(a) as an example, where the first row is its radical composition and the second row is its strokes composition. As the smallest unit of each Chinese character, there are few types of strokes, thus the stroke sequence corresponding to the character is usually long and many-to-one relationships between stroke sequences and characters are common (Chen, Li, and Xue 2021). Besides, strokes in characters often appear in combination, accompanied by connections and intersections. These properties add extra difficulties to stroke-based character recognition.

The number of radicals, which is the direct composition of characters, is higher than strokes, but much smaller than characters. Indeed, over 6,000 common Chinese characters can be composed of less than 500 radicals. In addition, due to the complex form and structural relations, radicals remain semantic information that is not available in strokes. It follows that converting training on character categories to training on radicals can significantly reduce the difficulty of the classification task while maintaining the semantics of the characters. Therefore, we intend to utilize radical information to complete character recognition.

For a character, radical information consists of two parts: the radical categories and the specific structure that organizes these radicals. In addition to the radical class itself, the specific structure is also crucial for characters, as the same radical composition will point to different candidate characters when the structure changes, two sets of examples are given in Fig. 2(b). More in detail, the character structure is the relative localization relation between all composing radicals, here we refer to this as structural relation and predefine them separately for different languages in this paper. Take the oracle bone character as an example, we have predefined 14 structural relations shown in Fig. 2(c). Note that we have devised a unique structure, named ‘Multiple’, which contains all characters containing four or more radicals. This is because characters containing many radicals (more than 3) have complex radical structural relations, but there is usually one unique candidate character for the same radical composition. Thus, ‘Multiple’ as a general structure for characters with multiple radicals saves labour and calculation costs while keeping accuracy.

The overall architecture diagram of our proposed REZCR is shown in Fig. 3. In the training phase, the input character images are fed to the RIE to generate the candidate radical sequences and their structural relations sequences. In the reasoning phase, these two sequences are fed together to the KGR module, where the proposed weight-fusion algorithm applies a knowledge graph for reasoning to obtain all candidate characters. Then rank them according to the confidences and output the recognition results. Details are described below.

Radical Information Extractor

In order to learn the radical information within the characters, we annotate the radicals contained in the characters under the guidance of linguists, including the radical category \( r_c \), the radical localization, and the structural relations among radicals \( s \). The location of the radical is identified by the coordinates \( (x, y, w, h) \) of the bounding box. The input character image \( I_C \in \mathbb{R}^{H \times W \times C} \), where the spatial resolution is \( H \times W \), and the number of channels is \( C \). The double-headed outputs of RIE are the two radical information, candidate radical sequences \( R_i \) and its structural relationship sequence \( S_j \).
Character images acquired in the real world are usually accompanied by noisy backgrounds, such as rubbing and corrosion traces in oracle bone characters and uneven backgrounds in street view characters, which will interfere with the learning of characters. In order to extract robust features for radicals, we introduce a radical attention block (RAB) in the RIE module to focus on the radicals in characters for mitigating the interference of background in character images. The RAB, whose backbone is ResNet [He et al. 2016], consist of two consecutive CBRs and a dual spatial attention layer (DSAL), as shown in Fig. 3. The CBR consists of a 3 × 3 CNN, a batch normalization, and ReLU as the unit of ResNet. The DSAL has two attention weight calculations, keeping the texture information of the radicals for feature selection via MaxPool and transferring the background information of the character image to maintain the integrity of the information. Element-by-element multiplication is used to calculate the correlation of two spatial vectors: the larger the product of the vectors, the stronger the correlation between the two vectors. The formula for calculating attention is as:

\[
A_1(F_i) = \sigma(f(M\text{axPool}(F_i)) \otimes F_i),
\]

\[
A_2(F_i) = \sigma(f(AvgPool(A_1(F_i))) \otimes F_i),
\]

where \(A\) denotes the attention calculation function, \(\sigma\) is the sigmoid function, and \(f\) denotes the convolution. \(F_i\) is the feature map, \(F_i \in \mathbb{R}^{H \times W \times C}\). The final attention vector is \(A_2\), which helps to keep radical features by mitigating the interference by background.

After RAB finishing feature extraction, we obtain \(F_r\), a deep feature that will be used to perform radical detection and subsequent extraction of structural relation among radicals. Next, the RIE will move towards two heads. The first head is designed to output radical categories, \(F_r\) goes through an FC layer, and the output dimension is \(K \times K \times (M \times (n_r + 5))\), where \(K\) represents dividing the input character image into \(K \times K\) grids, \(M\) represents that each grid needs to detect \(M\) anchor boxes, \(n_r\) is the number of radical categories in the dataset, and 5 represents the coordinate information \((x, y, w, h)\) for an anchor box and its confidence. In our experiment, we use \(K = 7\), and \(M = 3\). We output each radical whose confidence is within the threshold. For candidate boxes with similar centroid and area, we treat them as the same candidate box and merge the candidate radicals and predictions within them. The results are a set of candidate radical sequences (CRs). Depending on the localization of the candidate boxes, CRs contain \(n\) candidate radical sequences \(R_i\), where \(n\) is the number of possible radicals contained in the character, \(1 \leq i \leq n\), and \(R_i\) consists of all possible candidate radicals in this candidate box.

The second head is designed to output radical structural relations. In order to combine shallow features in character images with deep features learned from radicals, we integrate the output of the first RAB \(F_1\) with the output of the last RAB \(F_r\) for further radical structure relationship learning. When extracting structural features, we freeze all RABs and only perform parameter updates for subsequent convolutional layers. We output all structural relations and their predictions as sequences of structural relations (SRs), where the length of the sequence is \(n_s\), i.e. the number of pre-defined structural relations.

In RIE, the loss function consists of two components, the loss of candidate radicals extraction \(\mathcal{L}_{CR}\) and the loss of candidate structural relations recognition \(\mathcal{L}_{SR}\), so:

\[
\mathcal{L}_{RI} = \mathcal{L}_{CR} + \mathcal{L}_{SR},
\]

where \(\mathcal{L}_{CR}\) is made up of three components, it can be expressed as:

\[
\mathcal{L}_{CR} = \mathcal{L}_R + \mathcal{L}_{box} + \mathcal{L}_{isR},
\]

Specifically, \(\mathcal{L}_R\) is the loss for the radical category, \(\mathcal{L}_{box}\) is the loss for the bounding box introduced when locating the radical, which is calculated by the IoU of the candidate box and the real box. \(\mathcal{L}_{isR}\) is the loss for the bounding box...
confidence, where it consists of two parts, one is the loss value of the existence of the object, and the second part is the loss value of the absence of the object. They can be express as:

\[
\mathcal{L}_R = \lambda_r \sum_{i=0}^{K^2} \sum_{u=1}^{\#i_{\mathcal{R}}} [p_i(R) \log(p_i(R)) + (1 - p_i(R)) \log(1 - p_i(R))],
\]

\[
\mathcal{L}_{box} = \lambda_c \sum_{i=0}^{K^2} \sum_{j=0}^{M} \sum_{u=1, j}^{\#i_{\mathcal{R}}} [(2 - w_i h_i)(x_i - x^j)^2 + (y_i - y^j)^2]
\]

\[
+ \lambda_c \sum_{i=0}^{K^2} \sum_{j=0}^{M} \sum_{u=1, j}^{\#i_{\mathcal{R}}} [(w_i - w^j)^2 + (h_i - h^j)^2],
\]

\[
\mathcal{L}_{ISR} = \lambda_{ISR} \sum_{i=0}^{K^2} \sum_{u=1}^{\#i_{\mathcal{R}}} (R_i - \hat{R}_i)^2
\]

\[
+ \lambda_{ISR} \sum_{i=0}^{K^2} \sum_{j=0}^{M} \sum_{u=1, j}^{\#i_{\mathcal{R}}} (R_i - \hat{R}_i)^2,
\]

where \( R \) represents radical categories, \( p_i(R) \) is the true probability of the target radical, \( \hat{p}_i(R) \) is the predicted radical. \( \mathcal{L}_{SR} \) is the loss of radical structural relation recognition, which adopts the cross-entropy loss function to measure the difference between the true structural relation label and the predicted result, as:

\[
\mathcal{L}_{SR} = \frac{\lambda_s}{N} \sum_{i=0}^{N} q_i(S) \log(\hat{q}_i(S)).
\]

where \( N \) is the number of structure categories, \( q_i(S) \) is the true structure probability while \( \hat{q}_i(S) \) is the predicted structure.

**Knowledge Graph Reasoner**

We collected character images of Oracle, Bronze, simplified Chinese, and Korean characters, respectively, and annotated all characters in the dataset with radicals, as well as their structural relations under the guidance of linguists to construct character datasets and character knowledge graphs (CKG). As an initial version, the current CKG consists of 3 layers, including:

- **Characters**: all characters in the dataset. Note that ancient characters (oracle bone and Bronze characters) are associated with simplified Chinese characters for recognition.
- **Radicals**: components of characters. All character forms are associated with the corresponding radicals.
- **Structure**: the structure of a character, namely, the relative structural relation between all the radicals in a character.

We introduce an algorithm based on weight fusion to reason about characters, as shown in Algorithm 1. The input is sequences of CRs and SRs obtained in RIE. As a first step, we extract all possible sequences of character candidate radicals, reason all possible characters through the CKG, and calculate the prediction \( p^c_{i} \) for each candidate character by averaging the prediction of the candidate radicals. The formula is as follows:

\[
p^c_{i} = \frac{1}{n} \sum_{i=0}^{n} p^r_{j},
\]

where \( p_{r} \) is the prediction of each candidate radical, and \( n \) is the number of the candidate radicals.

In the second step, the radical structure relationship is applied to verify whether the reasoned character is correct. The final candidate character prediction \( p_{c} \) will be determined by a weighted fusion of the character prediction \( p^c_{i} \) from the first stage and the confidence \( p_{s} \) of the radical structure relationship as follows:

\[
p_{c} = \theta p^c_{i} + (1 - \theta)p_{s},
\]

where \( \theta \) is the weight of the first stage character prediction. The final candidate character with the highest \( p_{c} \) will be the character identified for recognition.

As long as the character is stored in the CKG, the REZCR is able to recognise it, even if it is unseen during training. It is worth mentioning that our method supports the addition of new character categories: when a new character category is encountered, there is no need to collect training samples or retrain the model, since it can be recognised by simply updating the knowledge graph.

**Experiments**

**Experimental Setups**

**Datasets.** Long-tail effect widely appears in real-world datasets, some existing character image datasets try to alleviate this issue by manually copying and simulating. However, these ideally constructed datasets are not satisfied with real-world data distribution (e.g., unrepeatable calligraphic characters), which will affect the training model and further limit the final performance. Therefore, we introduce a new character images dataset that conforms to the above.
Oracle character rubbing images are collected in this new dataset, called OracleRC, which is a few-samples dataset with long-tail effects. All oracle character images are collected from [Wu2012] and have been uniformly denoised. It contains a total of 2005 oracle characters, which contain 202 radicals and 14 structural relations. The maximum number of character samples in this dataset is 32, whereas the minimum is 1; the maximum number of radical samples is 206, whereas the minimum is 30. Both radicals and structural relations were manually annotated by archaeologists.

To demonstrate that our proposed REZCR is a general character recognition method, we conduct experiments on the simplified Chinese datasets ICDAR2013 [Yin et al. 2013], HWDB1.1 [Liu et al. 2013], CTW [Yuan et al. 2019], and the Korean dataset PE92 [Kim et al. 1996].

### Implementation Details

We implement our method with PyTorch. The size of the input image is normalised to $32 \times 32$, and data enhancement strategies such as translation, rotation, scaling, and background transformation are used. All experiments are conducted with Adadelta optimization where the hyperparameters were set to $\rho=0.95$ and $\epsilon=10^{-6}$.

### Experimental Results

#### Results on Few Samples Dataset

Experiments are conducted on the OracleRC dataset, a few-sample character image dataset with long-tail effects. From each category, a random selection of 75% of the data are used as the training set, and the remaining 25% are used as the test set. Regarding the category with only one sample, all samples are used as test samples, namely, no training samples: there are 518 in this dataset. The average accuracies of all methods on OracleRC are shown in Table 1. It can be seen that our method significantly outperforms all other methods from Top-1 to Top-10, including both traditional statistical-based and deep learning-based methods. The reason for this disparity is the insufficient amount of training data, which limits the performance of deep neural networks. In contrast, after we decompose characters into radicals, the number of training categories is significantly reduced and the number of samples per category is increased to at least 30 samples. This largely alleviates the difficulty brought by a few samples to the character recognition task.

It is worth noting that our method is effective in recognising the character categories that have not been trained. For the 518 unseen categories, it gets an accuracy of 72.01% for Top-10. This is a small reduction compared to the recognition accuracy of all character categories, which demonstrates the superiority of our method in the zero-shot learning task.

#### Results for Zero-shot Learning

In this section, we design a series of experiments to demonstrate the effectiveness of REZCR in identifying unseen characters. Character-based methods cannot recognize unseen characters, which means that they get 0% accuracy in this experiment. Therefore, we only make a comparison with radical-based and stroke-based methods on zero-shot learning. We applied three datasets, namely the scene character dataset CTW, and the handwritten Chinese character datasets HWDB1.1 and ICDAR2013. In order to conduct experiments on the Chinese character dataset, we annotate the Chinese characters with radicals and their structural relations under the guidance of linguists.

For street view characters. The CTW dataset contains 3650 character categories with a total of 812,872 samples. We sort all character categories according to the numbered positions in the national standard GB18030-2005, and collect samples from the first 500 classes as the test set. The next 500 categories will be used as the training set, where $n \in \{500, 1000, 1500, 2000, 3150\}$. For handwritten Chinese characters. In this experiment, we apply both HWDB1.1 and ICDAR2013 datasets. We select samples of the first n character categories from HWDB1.1 as the training set, where $n \in \{500, 1000, 1500, 2000, 2755\}$. Then we select the samples in the last 1000 character categories as the test set from ICDAR2013. Results are shown in Table 2 proving that our method outperforms other methods in different numbers of training sets.

More in detail, our method recognizes characters by decomposing them into radicals and structural relations, and by introducing a CKG as external knowledge for reasoning. While other methods also disassemble characters, they all use a set of ordered sequences (such as IDS sequences) to represent them: the number of elements in the sequence ranges from 2 to 16 (radical-based methods) or 1 to 24 (stroke-based methods). This places high demands on the recognition method: only if each element and its localisation in the sequence are identified accurately can the character category be correctly obtained. The longer the sequence of characters, the more inaccurate the prediction.

The above problem does not arise in our method, since we do not require that the radicals are recognised in a given order, and each radical is stored equally in the CKG. Meanwhile, our method outputs all possible classes of radicals, structural relations, and their precision in the first stage, and all possible combinations of radical information are utilized in the CKG for reasoning. Calculated with a weighted

### Table 1: Averaged accuracy of different methods on OracleRC.

| Method          | Top-1 | Top-3 | Top-5 | Top-10 |
|-----------------|-------|-------|-------|--------|
| HOG+SVM (Dalal and Triggs 2005) | 9.29% | 12.75% | 18.26% | 24.35% |
| AlexNet (Krizhevsky, Sutskever, and Hinton 2012) | 28.03% | 36.45% | 40.03% | 47.49% |
| VGG16 (Simonyan and Zisserman 2015) | 27.75% | 38.12% | 41.53% | 48.86% |
| ResNet [He et al. 2016] | 30.32% | 39.21% | 43.06% | 51.66% |
| Inception-v4 [Szegedy et al. 2017] | 30.32% | 49.96% | 52.13% | 58.72% |
| REZCR (Ours) | 60.28% | 69.74% | 72.66% | 82.45% |

### Table 2: Performance comparison of the zero-shot character recognition using different methods on the two sets of datasets. $n$ is the character category used for training, and the test character category was fixed at 500 and 1000, respectively.

| Method | CTW $n$ | ICDAR2013 $n$ |
|--------|---------|---------------|
| HOG+SVM (Dalal and Triggs 2005) | 54.82% | 71.76% |
| AlexNet (Krizhevsky, Sutskever, and Hinton 2012) | 79.11% | 95.01% |
| VGG16 (Simonyan and Zisserman 2015) | 81.78% | 97.15% |
| ResNet [He et al. 2016] | 82.34% | 97.41% |
| Inception-v4 [Szegedy et al. 2017] | 83.03% | 97.62% |
| REZCR (Ours) | 93.82% | 97.91% |

The link of the proposed public accessible character image dataset will be updated later due to blind review.
Table 3: Character recognition performance comparisons on two datasets with the state-of-art methods.

| Method               | ICDAR2013 | CTW  |
|----------------------|-----------|------|
| AlexNet (Krizhevsky, Sutskever, and Hinton 2012) | 89.99%    | 76.49%|
| ResNet (He et al. 2016) | 92.18%    | 79.46%|
| Inception-v1 (Szegedy et al. 2017) | 95.79%    | 82.28%|
| DenseNet (Huang et al. 2017) | 95.90%    | 80.00%|
| RAN (Zhang et al. 2018) | 93.79%    | 81.80%|
| DenseRAN (Wang et al. 2018) | 96.96%    | 85.56%|
| FewshotRAN (Wang et al. 2019a) | 96.97%    | 86.78%|
| HDE-Net (Cao et al. 2020) | 96.74%    | 89.25%|
| Stroke-to-Character (Chen, Li, and Xue 2021) | 96.28%    | 85.29%|
| REZCR (Ours)         | 96.35%    | 84.21%|

fusion-based algorithm, we obtain the precision for all candidate characters, which in turn gives the final prediction results. Based on this process, our method has a strong self-correcting capability and it is still possible to obtain the correct character categories even in the case of incorrect Top1 candidate radicals.

Results for Generalization. To demonstrate the generalization of REZCN, we perform experiments on the ICDAR2013 and CTW datasets in a general manner (Cao et al. 2020). We directly use the official training set and the test set of the dataset for experiments (taking CTW as an example, the training set contains 760,107 samples, while the test set contains 52,765). Experimental results are shown in Table 3. It is undeniable that character-based deep learning methods perform well with sufficient sample size, but our method also achieves competitive results. As a general character recognition method, our method not only outperforms on general datasets, but also has unique advantages on few-sample datasets.

Ablation Study

To further validate the effectiveness of the proposed method, an extensive ablation study was carried out on the OracleRC dataset for different components of the proposed REZCR.

Impact of Learning and Reasoning Strategies. To validate the radical-based character recognition framework, we conduct ablation experiments on the design of learning and reasoning strategies in two modules in REZCR, i.e., RIE and KGR, respectively. Results are shown in Table 4. “Without radicals” refers to the learning for character recognition without extracting the radical information; “Radicals without structures” is to learn the radical categories in character images without considering the radical structure relations; “Radical + structures (RIE)” is to learn both the radical categories and the structural relations in character images, namely our proposed RIE module. The latter four experiments were carried out for different reasoning strategies. “RIE + Top1 radicals” refers to using the radicals with the highest prediction confidence for reasoning. “RIE + Top1 radicals & structure” refers to using both the radicals and the structural relation with the highest prediction confidence for reasoning. “RIE + radical sequences” is to input the sequences of all candidate radicals obtained from the RIE for reasoning. “RIE + radical & structure sequences (REZCR)” is to utilize all candidate radicals sequences and candidate structural relations sequences obtained from RIE for reasoning in KGR, which applies the weight fusion-based algorithm, namely our proposed REZCR.

From the first two experiments, we find that learning characters directly is not a good choice for few-sample character datasets: these few samples are unable to support the model to get a reliable character distribution, especially for images with only one sample. In contrast, the strategy of learning radicals effectively improves the performance of character recognition, with an accuracy improvement of about 30%, which is a remarkable improvement. The comparison between (b) and (c), (d) and (e), and (f) and (g) shows that the extraction of radical structure relations results in about a 2% improvement. This indicates that structure relations further complements the radical information, and effectively addresses the problem brought by the fact that different characters are composed of the same several radicals. Observing the experiments in (d) and (f), and in (e) and (g), we can find that the reasoning strategy using sequences has an ac-

Table 4: Ablation study results on learning and reasoning strategies.

| Index | Method                                | Acc.  |
|-------|---------------------------------------|-------|
| (a)   | Without radicals                      | 26.93%|
| (b)   | Radicals without structures            | 54.87%|
| (c)   | Radicals + structures (RIE)            | 56.84%|
| (d)   | RIE + Top1 radicals                    | 52.93%|
| (e)   | RIE + Top1 radicals & structure        | 55.62%|
| (f)   | RIE + radical sequences                | 58.06%|
| (g)   | RIE + radical & structure sequences (REZCR) | 60.28%|

Table 5: Ablation study results on different attention block scheme.

| Attention layer | Acc.  |
|-----------------|-------|
| SSAL-a          | 59.23%|
| SSAL-b          | 59.34%|
| SSAL-c          | 59.79%|
| DSAL-a          | 60.03%|
| DSAL-b (Ours)   | 60.28%|
curacy improvement of about 5% compared to the strategy using Top1 results, which demonstrates the superiority of the weight fusion-based knowledge graph reasoning algorithm.

**Impact of Attention Layer Schemes.** In this paper, we propose a dual spatial attention layer (DSAL) to help with the extraction of radical information. To verify the effectiveness of DSAL, we experiment with five different attention blocks, as shown in Fig. 4. The single spatial attention layers (SSAL) are divided into three types depending on the pooling layer applied, called SSAL-a, SSAL-b and SSAL-c, respectively, while DSAL has two different connections schemes, labelled DSAL-a and DSAL-b, respectively. Results of the experiments are shown in Table 5. It is intuitive to find that DSAL outperformed the SSAL on the OracleRC dataset, and DSAL-b achieved the best performance, which is currently applied in REZCR.

**Conclusion**

In this paper, we discuss the peculiarities of character composition and the importance of radical information for character recognition. Then, we propose a novel character recognition method, REZCR, which consists of two novel modules, i.e., RIE and KGR. In the former, candidate radicals and their possible structural relations are extracted to offer radical information, where attention mechanisms are introduced for mitigating the background interference. Target characters are reasoned with the CKG in the latter, where we introduce a weight-fusion based algorithm. Experimental results show that the REZCR obtains promising performance on multiple character datasets, especially on few-sample datasets. The ablation study shows the effectiveness of each proposed improvement. Furthermore, the results based on different languages prove that REZCR has great generalization, future applications including oracle, Bronze, simplified Chinese, traditional Chinese, Korean, etc.

**References**

Akata, Z.; Perronnin, F.; Harchaoui, Z.; and Schmid, C. 2015. Label-embedding for image classification. IEEE TPAMI, 38(7): 1425–1438.

Anderson, C. 2006. *The long tail: Why the future of business is selling less of more.* Hachette Books.

Cao, Z.; Lu, J.; Cui, S.; and Zhang, C. 2020. Zero-shot Handwritten Chinese Character Recognition with hierarchical decomposition embedding. Elsevier PR, 107: 107488.

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

Cireşan, D.; and Meier, U. 2015. Multi-column deep neural networks for offline handwritten Chinese character classification. In IJCNN.

Cui, Y.; Jia, M.; Lin, T.-Y.; Song, Y.; and Belongie, S. 2019. Class-balanced loss based on effective number of samples. In CVPR.

Dalal, N.; and Triggs, B. 2005. Histograms of oriented gradients for human detection. In CVPR.

De Boer, P.-T.; Kroese, D. P.; Mannor, S.; and Rubinstein, R. Y. 2005. A tutorial on the cross-entropy method. Annals of operations research, 134(1): 19–67.

Feng, Z. 2014. Chinese Romanization and its application in HCI. In *HCI*.

Géron, A. 2019. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems.* O’Reilly Media.

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

Huang, G.; Liu, Z.; Van Der Maaten, L.; and Weinberger, K. Q. 2017. Densely connected convolutional networks. In CVPR.

Huang, S.; Wang, H.; Liu, Y.; Shi, X.; and Jin, L. 2019. OBC306: A large-scale Oracle Bone character recognition dataset. In ICDAR.

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

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. ImageNet classification with deep convolutional neural networks. In NeurIPS.

Lampert, C. H.; Nickisch, H.; and Harmeling, S. 2009. Learning to detect unseen object classes by between-class attribute transfer. In CVPR.

Lampert, C. H.; Nickisch, H.; and Harmeling, S. 2013. Attribute-based classification for zero-shot visual object categorization. IEEE TPAMI, 36(3): 453–465.

Liu, C.-L.; Yin, F.; Wang, D.-H.; and Wang, Q.-F. 2011. CASIA online and offline Chinese handwriting databases. In ICDAR.

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

Lyu, P.; Bai, X.; Yao, C.; Zhu, Z.; Huang, T.; and Liu, W. 2017. Auto-encoder guided GAN for Chinese calligraphy synthesis. In ICDAR.

Mahajan, D.; Girshick, R.; Ramanathan, V.; He, K.; Paluri, M.; Li, Y.; Bharamee, A.; and Van Der Maaten, L. 2018. Exploring the limits of weakly supervised pretraining. In ECCV.

Myers, J. 2016. Knowing Chinese character grammar. Elsevier Cognition, 147: 127–132.

Rohrbach, M.; Stark, M.; and Schiele, B. 2011. Evaluating knowledge transfer and zero-shot learning in a large-scale setting. In CVPR.

Shanthi, N.; and Duraiswamy, K. 2010. A novel SVM-based handwritten Tamil character recognition system. Springer Pattern Analysis and Applications, 13(2): 173–180.

Shen, H. H. 2005. An investigation of Chinese-character learning strategies among non-native speakers of Chinese. Elsevier System, 35(1): 49–68.

Simonyan, K.; and Zisserman, A. 2015. Very deep convolutional networks for large-scale image recognition. In ICLR.
Su, Y.-M.; and Wang, J.-F. 2003. A novel stroke extraction method for Chinese characters using Gabor filters. *Elsevier PR*, 36(3): 635–647.

Szegedy, C.; Ioffe, S.; Vanhoucke, V.; and Alemi, A. A. 2017. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*.

Wang, T.; Xie, Z.; Li, Z.; Jin, L.; and Chen, X. 2019a. Radical aggregation network for few-shot offline handwritten Chinese character recognition. *Elsevier PRL*, 125: 821–827.

Wang, W.; Zhang, J.; Du, J.; Wang, Z.-R.; and Zhu, Y. 2018. Denseran for offline handwritten character recognition. In *ICFHR*.

Wang, W.; Zheng, V. W.; Yu, H.; and Miao, C. 2019b. A survey of zero-shot learning: Settings, methods, and applications. *ACM TIST*, 10(2): 1–37.

Wang, X.; Ye, Y.; and Gupta, A. 2018. Zero-shot recognition via semantic embeddings and knowledge graphs. In *CVPR*.

Wu, Z. 2012. *Shang and Zhou Bronze Inscriptions and Image Integration*. Shanghai Classics Publishing House. ISBN 9787532564231.

Xie, G.-S.; Liu, L.; Jin, X.; Zhu, F.; Zhang, Z.; Qin, J.; Yao, Y.; and Shao, L. 2019. Attentive region embedding network for zero-shot learning. In *CVPR*.

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

Yuan, A.; Bai, G.; Jiao, L.; and Liu, Y. 2012. Offline handwritten English character recognition based on convolutional neural network. In *DAS*.

Yuan, T.-L.; Zhu, Z.; Xu, K.; Li, C.-J.; Mu, T.-J.; and Hu, S.-M. 2019. A large Chinese text dataset in the wild. *Springer JCST*, 34(3): 509–521.

Zhang, J.; Du, J.; and Dai, L. 2020. Radical analysis network for learning hierarchies of Chinese characters. *Elsevier PR*, 103: 107305.

Zhang, J.; Zhu, Y.; Du, J.; and Dai, L. 2018. Radical analysis network for zero-shot learning in printed Chinese character recognition. In *ICME*.

Zhang, X.-Y.; Bengio, Y.; and Liu, C.-L. 2017. Online and offline handwritten Chinese character recognition: A comprehensive study and new benchmark. *Elsevier PR*, 61: 348–360.