# RADICAL ANALYSIS NETWORK FOR ZERO-SHOT LEARNING IN PRINTED CHINESE CHARACTER RECOGNITION

*Jianshu Zhang, Yixing Zhu, Jun Du and Lirong Dai*

National Engineering Laboratory for Speech and Language Information Processing
University of Science and Technology of China, Hefei, Anhui, P. R. China
xysszjs@mail.ustc.edu.cn, zyxsa@mail.ustc.edu.cn, jundu@ustc.edu.cn, lrdai@ustc.edu.cn

## ABSTRACT

Chinese characters have a huge set of character categories, more than 20,000 and the number is still increasing as more and more novel characters continue being created. However, the enormous characters can be decomposed into a compact set of about 500 fundamental and structural radicals. This paper introduces a novel radical analysis network (RAN) to recognize printed Chinese characters by identifying radicals and analyzing two-dimensional spatial structures among them. The proposed RAN first extracts visual features from input by employing convolutional neural networks as an encoder. Then a decoder based on recurrent neural networks is employed, aiming at generating captions of Chinese characters by detecting radicals and two-dimensional structures through a spatial attention mechanism. The manner of treating a Chinese character as a composition of radicals rather than a single character class largely reduces the size of vocabulary and enables RAN to possess the ability of recognizing unseen Chinese character classes, namely zero-shot learning.

***Index Terms***— Chinese characters, radical analysis, zero-shot learning, encoder-decoder, attention

## 1. INTRODUCTION

The recognition of Chinese characters is an intricate problem due to a large number of existing character categories (more than 20,000), increasing novel characters (e.g. the character "Duang" created by Jackie Chan) and complicated internal structures. However, most conventional approaches [1] can only recognize about 4,000 commonly used characters with no capability of handling unseen or newly created characters. Also each character sample is treated as a whole without considering the similarity and internal structures among different characters.

It is well-known that all Chinese characters are composed of basic structural components, called radicals. Only about 500 radicals [2] are adequate to describe more than 20,000 Chinese characters. Therefore, it is an intuitive way to decompose Chinese characters into radicals and describe their spatial structures as captions for recognition. In this paper,
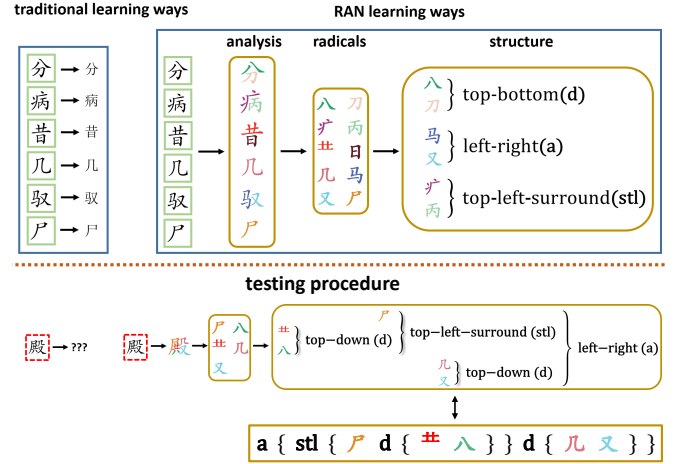


**Fig. 1**: The comparison between the conventional whole-character based methods and the proposed RAN method.

we propose a novel radical analysis network (RAN) to generate captions for recognition of Chinese characters. RAN has two distinctive properties compared with traditional methods: 1) The size of the radical vocabulary is largely reduced compared with the character vocabulary; 2) It is a novel zero-shot learning of Chinese characters which can recognize unseen Chinese characters in the recognition stage because the corresponding radicals and spatial relationships are learned from other seen characters in the training stage. Fig. 1 illustrates a clear comparison between traditional methods and RAN for recognizing Chinese characters. In the traditional methods, the classifier takes the character input as a single picture and tries to learn a mapping between the input picture and a pre-defined class. If the testing character class is unseen in training samples, like the character in red dashed rectangle in Fig. 1, it will be misclassified. While RAN attempts to imitate the manner of recognizing Chinese characters as Chinese learners. For example, before asking children to remember and recognize Chinese characters, Chinese teachers first teach them to identify radicals, understand the meaning of radicals and grasp the possible spatial structures between them. Such learning way is more generative and helps im-

prove the memory ability of students to learn so many Chinese characters, which is perfectly adopted in RAN. As the example in top-right part of Fig. 1 shows, the six training samples contain ten different radicals and exhibit top-bottom, left-right and top-left-surround spatial structures. When encountering the unseen character during testing, RAN can still generate the corresponding caption of that character (i.e. the bottom-right rectangle of Fig. 1) as it has already learned the essential radicals and structures. Technically, the enormous Chinese characters, as well as the newly created characters can all be identified by a compact set of radicals and spatial structures learned in the training stage.

In the past few decades, lots of efforts have been made for radical-based Chinese character recognition. [3] over-segmented characters into candidate radicals and could only handle the left-right structure. [4] first detected separate radicals and then employed a hierarchical radical matching method to recognize a Chinese character. Recently, [5] also tried to detect position-dependent radicals using a deep residual network with multi-labeled learning. Generally, these approaches have difficulties when dealing with the complicated 2D structures between radicals and do not focus on the recognition of unseen character classes.

Regarding to the network architecture, the proposed RAN is an improved version of the attention-based encoder-decoder model in [6]. We employ convolutional neural networks (CNN) [7] as the encoder to extract high-level visual features from input Chinese characters. The decoder is recurrent neural networks with gated recurrent units (GRU) [8] which converts the high-level visual features into output character captions. We adopt a coverage based spatial attention model built in the decoder to detect the radicals and internal two-dimensional structures simultaneously. The GRU based decoder also performs like a potential language model aiming to grasp the rules of composing Chinese character captions after successfully detecting radicals and structures. The main contributions of this study are summarized as follows:

- We propose RAN for the zero-shot learning of Chinese character recognition, solving the problem of handling unseen or newly created characters.

- We describe how to caption Chinese characters based on detailed analysis of Chinese radicals and structures.

- We experimentally demonstrate how RAN performs on recognizing seen/unseen Chinese characters and show its efficiency through attention visualization.

## 2. RADICAL ANALYSIS

Compared with enormous Chinese character categories, the amount of radical categories is quite limited. It is declared in GB13000.1 standard [2], which is published by National Language Committee of China, that 20902 Chinese characters are composed of 560 different radicals. Following the strategy
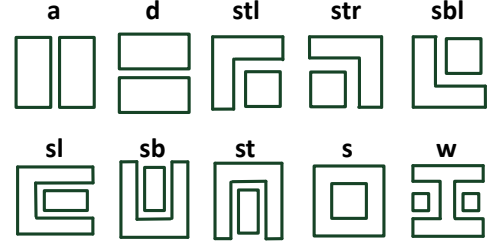


**Fig. 2**: Graphical representation of ten common spatial structures between Chinese radicals.

in cjk-decomp [1], we decompose Chinese characters into corresponding captions. Regarding to spatial structures among radicals, we show ten common structures in Fig. 2 where "**a**" represents a left-right structure, "**d**" represents a top-bottom structure, "**stl**" represents a top-left-surround structure, "**str**" represents a top-right-surround structure, "**sbl**" represents a bottom-left-surround structure, "**sl**" represents a left-surround structure, "**sb**" represents a bottom-surround structure, "**st**" represents a top-surround structure, "**s**" represents a surround structure and "**w**" represents a within structure.

We use a pair of braces to constrain a single structure in character caption. Taking "stl" as an example, it is captioned as "stl { radical-1 radical-2 }". Usually, like the common instances mentioned above, a structure is described by two different radicals. However, as for some unique structures, they are described by three or more radicals.

## 3. NETWORK ARCHITECTURE OF RAN

The attention based encoder-decoder model first learns to encode input into high-level representations. A fixed-length context vector is then generated via weighted summing the high-level representations. The attention performs as the weighting coefficients so that it can choose the most relevant parts from the whole input for calculating the context vector. Finally, the decoder uses this context vector to generate variable-length output sequence word by word. The framework has been extensively applied to many applications including machine translation [9], image captioning [10, 11], video processing [12] and handwriting recognition [13–15].

### 3.1. CNN encoder

In this paper, we evaluate RAN on printed Chinese characters. The inputs are greyscale images and the pixel value is normalized between 0 and 1. The overall architecture of RAN is shown in Fig. 3. We employ CNN as the encoder which is proven to be a powerful model to extract high-quality visual features from images. Rather than extracting features after a fully connected layer, we employ a CNN framework containing only convolution, pooling and activation layers, called
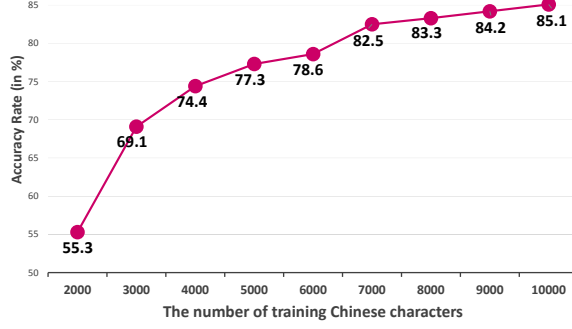
---

[1]https://github.com/amake/cjk-decomp

**Fig. 3**: The overall architecture of RAN for printed Chinese character recognition. The purple, blue and green rectangles are examples of attention visualization when predicting the words that are of the same color in output caption.

fully convolutional neural networks. It does make sense because the subsequent decoder can selectively pay attention to certain pixels of an image by choosing specific portions from all the extracted visual features.

Assuming that the CNN encoder extracts high-level visual representations denoted by a three-dimensional array of size $H \times W \times D$, then the CNN output is a variable-length grid of $L$ elements, $L = H \times W$. Each of these elements is a $D$-dimensional annotation vector corresponding to a local region of the image.

$$\mathbf{A} = \{\mathbf{a}_1, \ldots, \mathbf{a}_L\} \ , \ \mathbf{a}_i \in \mathbb{R}^D \tag{1}$$

### 3.2. Decoder with spatial attention

#### 3.2.1. Deocder

As illustrated in Fig. 3, the decoder aims to generate a caption of input Chinese character. The output caption $\mathbf{Y}$ is represented by a sequence of one-hot encoded words.

$$\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_C\} \ , \ \mathbf{y}_i \in \mathbb{R}^K \tag{2}$$

where $K$ is the number of total words in the vocabulary which includes the basic radicals, spatial structures and a pair of braces, $C$ is the length of caption.

Note that, the length of annotation sequence ($L$) is fixed while the length of caption ($C$) is variable. To address the problem of associating fixed-length annotation sequences with variable-length output sequences, we attempt to compute an intermediate fixed-size vector $\mathbf{c}_t$ which will be described in Section 3.2.2. Given the context vector $\mathbf{c}_t$, we utilize unidirectional GRU to produce captions word by word. The GRU is an improved version of simple RNN as it alleviates the problems of the vanishing gradient and the exploding gradient as described in [16, 17]. The probability of each predicted word is computed by the context vector $\mathbf{c}_t$, current GRU hidden state $\mathbf{s}_t$ and previous target word $\mathbf{y}_{t-1}$ using the following equation:

$$\mathbf{P}(\mathbf{y}_t | \mathbf{y}_{t-1}, \mathbf{X}) = g\left(\mathbf{W}_o h\left(\mathbf{E}\mathbf{y}_{t-1} + \mathbf{W}_s \mathbf{s}_t + \mathbf{W}_c \mathbf{c}_t\right)\right) \tag{3}$$

where $g$ denotes a softmax activation function over all the words in the vocabulary, $h$ denotes a maxout activation function, $\mathbf{W}_o \in \mathbb{R}^{K \times \frac{m}{2}}$, $\mathbf{W}_s \in \mathbb{R}^{m \times n}$, $\mathbf{W}_c \in \mathbb{R}^{m \times D}$, and $\mathbf{E}$ denotes the embedding matrix, $m$ and $n$ are the dimensions of embedding and GRU decoder.

The decoder adopts two unidirectional GRU layers to calculate the hidden state $\mathbf{s}_t$:

$$\hat{\mathbf{s}}_t = \text{GRU}\left(\mathbf{y}_{t-1}, \mathbf{s}_{t-1}\right) \tag{4}$$

$$\mathbf{c}_t = f_{\text{catt}}\left(\hat{\mathbf{s}}_t, \mathbf{A}\right) \tag{5}$$

$$\mathbf{s}_t = \text{GRU}\left(\mathbf{c}_t, \hat{\mathbf{s}}_t\right) \tag{6}$$

where $\mathbf{s}_{t-1}$ denotes the previous hidden state, $\hat{\mathbf{s}}_t$ is the prediction of current GRU hidden state, and $f_{\text{catt}}$ denotes the coverage based spatial attention model (Section 3.2.2).

#### 3.2.2. Coverage based spatial attention

Intuitively, for each radical or structure, the entire input character is not necessary to provide the useful information, only a part of input character should mainly contribute to the computation of context vector $\mathbf{c}_t$ at each time step $t$. Therefore, the decoder adopts a spatial attention mechanism to know which part of input is suitable to attend to generate the next predicted radical or structure and then assigns a higher weight to the corresponding local annotation vectors $\mathbf{a}_i$ (e.g. in Fig. 3, the higher attention probabilities are described by the lighter red color). However, there is one problem for the classic spatial attention mechanism, namely lack of coverage. Coverage means the overall alignment information that indicating whether a local part of input has been attended or not. The overall alignment information is especially important when recognizing Chinese characters because in principle, each radical or structure should be decoded only once. Lacking coverage will lead to misalignment resulting in over-parsing or under-parsing problem. Over-parsing implies that some radicals and structures have been decoded twice or more, while under-parsing denotes that some radicals and structures have never been decoded. To address this problem, we append a coverage vector to the computation of attention. The coverage vector aims at tracking the past alignment information. Here, we parameterize the coverage based attention model as a multi-layer perceptron (MLP) which is jointly trained with the encoder and the decoder:

$$\mathbf{F} = \mathbf{Q} * \sum_{l=1}^{t-1} \boldsymbol{\alpha}_l \tag{7}$$

$$e_{ti} = \boldsymbol{\nu}_{\text{att}}^{\text{T}} \tanh(\mathbf{W}_{\text{att}} \hat{\mathbf{s}}_t + \mathbf{U}_{\text{att}} \mathbf{a}_i + \mathbf{U}_f \mathbf{f}_i) \tag{8}$$

$$\alpha_{ti} = \frac{\exp(e_{ti})}{\sum_{k=1}^{L} \exp(e_{tk})} \tag{9}$$

where $e_{ti}$ denotes the energy of annotation vector $\mathbf{a}_i$ at time step $t$ conditioned on the current GRU hidden state prediction $\hat{\mathbf{s}}_t$ and coverage vector $\mathbf{f}_i$. The coverage vector is initialized as a zero vector and we compute it based on the summation of all

**Fig. 4**: The performance comparison of RAN for recognizing unseen Chinese characters with respect to the number of training Chinese characters.

past attention probabilities. $\alpha_{ti}$ denotes the spatial attention coefficient of $\mathbf{a}_i$ at time step $t$. Let $n'$ denote the attention dimension and $M$ denote the number of feature maps of filter $\mathbf{Q}$; then $\boldsymbol{\nu}_{\text{att}} \in \mathbb{R}^{n'}$, $\mathbf{W}_{\text{att}} \in \mathbb{R}^{n' \times n}$, $\mathbf{U}_{\text{att}} \in \mathbb{R}^{n' \times D}$ and $\mathbf{U}_f \in \mathbb{R}^{n' \times M}$. With the weights $\alpha_{ti}$, we compute the context vector $\mathbf{c}_t$ as:

$$\mathbf{c}_t = \sum_{i=1}^{L} \alpha_{ti} \mathbf{a}_i \tag{10}$$

which has a fixed-length one regardless of input image size.

## 4. EXPERIMENTS

The training objective of RAN is to maximize the predicted word probability in Eq. (3) and we use cross-entropy (CE) as the loss function. The CNN encoder employs VGG [18] architecture. We propose to use two different VGG architectures, namely VGG14-s and VGG14, due to the different size of training set. The "VGG14" means that the two architectures are both composed of 14 convolutional layers, which are divided into 4 blocks. The number of stacked convolutional layers in each block is (3, 3, 4, 4) and a max-pooling function is followed after each block. In VGG14-s, the number of output channels in each block is (32, 64, 128, 256). While in VGG14, the number of output channels in each block is (64, 128, 256, 512). Note that, the number of output channels of each convolutional layer in the same block remains unchanged. We use VGG14-s for experiments of recognizing unseen characters because the size of training set is relatively small, the suffix "s" means "smaller". While the VGG14 is used for experiments of recognizing seen characters with various font styles. The decoder is a single layer with 256 forward GRU units. The embedding dimension $m$ and GRU decoder dimension $n$ are set to 256. The attention dimension $n'$ is set to the annotation dimension $D$. The convolution kernel size for computing coverage vector is set to $(5 \times 5)$ and the number of feature maps $M$ is set to 256. We utilize the adadelta algorithm [19] with gradient clipping for optimization.

In the decoding stage, we aim to generate a most likely
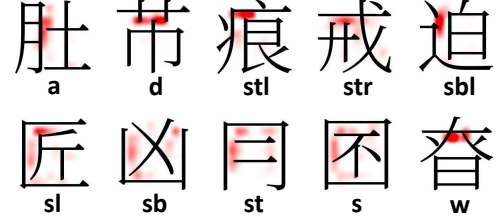


**Fig. 5**: Attention visualization of identifying ten common structures of Chinese characters.
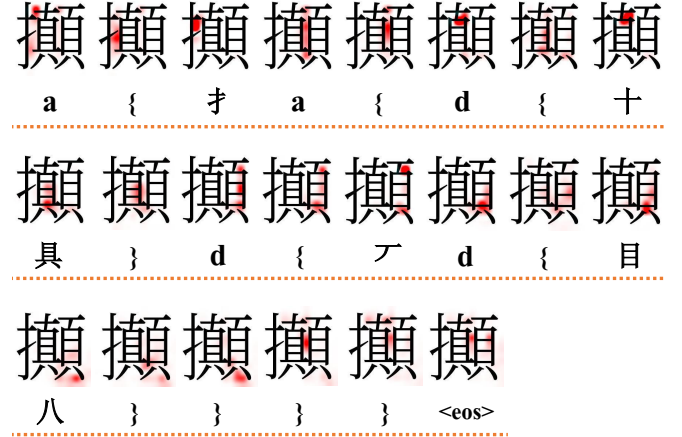


**Fig. 6**: Attention visualization of recognizing an unseen Chinese character.

caption string given the input character. The beam search algorithm [13] is employed to complete the decoding process because we do not have the ground-truth of previous predicted word during testing. The beam size is set to 10.

### 4.1. Experiments on recognition of unseen characters

In this section, we show the effectiveness of RAN on identifying unseen Chinese characters through accuracy rate and attention visualization. A test character is considered as successfully recognized only when its predicted caption exactly matches the ground-truth.

In this experiment, we choose 26,079 Chinese characters in Song font style which are composed of only 361 radicals and 29 spatial structures. We divide them into training set, validation set and testing set. We increase the training set from 2,000 to 10,000 Chinese characters to see how many training characters are enough to train our model to recognize the unseen 16,079 characters. As for the unseen 16,079 Chinese characters, we choose 2,000 characters as the validation set and 14,079 characters as the testing set. We also employ an ensemble method during testing procedure [13] because the performances vary severely due to the small training set. We illustrate the performance in Fig. 4, where 2,000 training Chinese characters can successfully recognize 55.3% unseen 14,079 Chinese characters and 10,000 training Chinese
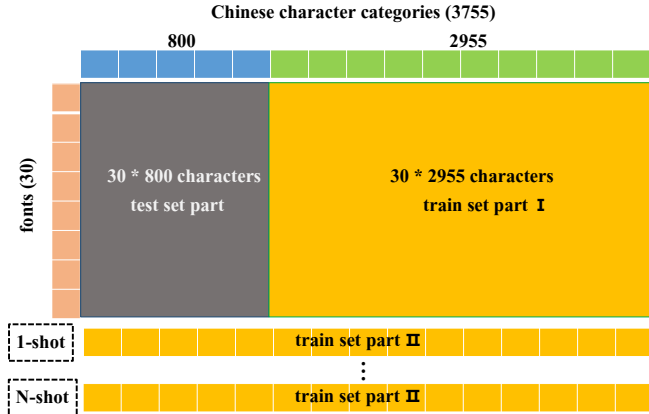
**Fig. 7**: Description of dividing training set and testing set for experiments of recognizing seen Chinese characters.

characters can successfully recognize 85.1% unseen 14,079 Chinese characters. Actually, only about 500 Chinese characters are adequate to cover overall Chinese radicals and spatial structures. However, our experiments start from 2,000 training characters because RAN is hard to converge when the training set is too small.

To generate a Chinese character caption, it is essential to identify the structures between isolated radicals. As illustrated in Fig. 5, we show ten examples on how RAN identifies common structures through attention visualization. The red color in attention maps represents the spatial attention probabilities, where the lighter color describes the higher attention probabilities and the darker color describes the lower attention probabilities. Taking "a" structure as an example, the decoder mainly focuses on the blank region between two radicals, indicating a left-right direction.

More specifically, in Fig. 6, we show that how RAN learns to recognize an unseen Chinese character from an image into a character caption step by step. When encountering basic radicals, the attention model well generates the alignment strongly corresponding to the human intuition. Also, it successfully generates the structure "a" and "d" when it detects a left-right direction and a top-bottom direction. Immediately after detecting a spatial structure, the decoder generates a pair of braces "{}", which is employed to constrain the structure in Chinese character caption.

### 4.2. Experiments on recognition of seen characters

In this section, we show the effectiveness of RAN on recognition of seen Chinese characters by comparing it with traditional whole-character modeling methods. The training set contains 3755 common used Chinese character categories and the testing set contains 800 character categories. The detailed implementation of organizing training set and testing set is illustrated in Fig. 7. We design this experiment like few-shot learning of Chinese character recognition. We divide 3755
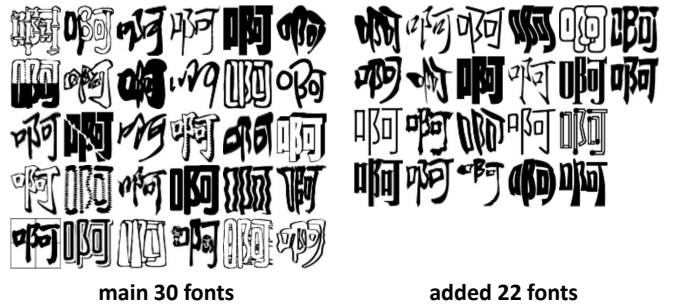


**Fig. 8**: Visualization of font styles.

**Table 1**: Comparison of Accuracy Rate (AR in %) between RAN with traditional whole-character based approaches.

|  | 1-shot | 2-shot | 3-shot | 4-shot |
|---|---|---|---|---|
| Zhong [20] | 4 | 16.3 | 43.7 | 62.4 |
| VGG14 | 23.4 | 58.4 | 74.3 | 82.2 |
| **RAN** | **80.7** | **85.7** | **87.9** | **88.3** |

characters into 2955 characters and other 800 characters. The 800 characters with 30 various font styles form the testing set and the other 2955 characters with the same 30 font styles become a part of training set. Additionally, we use 3755 characters with other font styles as a second part of training set. When we add 3755 characters with other N font styles as the second training set, we call this experiment N-shot. The number of font styles of the second part of training set increased from 1 to 22. The description of main 30 font styles and newly added 22 font styles are visualized in Fig. 8.

The system "Zhong" is the proposed system in [20]. Also, we replace the CNN architecture in "Zhong" with VGG14 and keep the other parts unchanged, we name it as system "VGG14". Table 1 clearly declares that RAN significantly outperforms the traditional whole-character based approaches on seen Chinese characters, when the training samples of these character classes are very few. Fig. 9 illustrates the comparison when the number of training samples of seen classes increases and RAN still consistently outperforms whole-character based approaches.

### 5. CONCLUSIONS AND FUTURE WORK

In this paper, we introduce a novel model named radical analysis network for zero-shot learning of Chinese characters recognition. We show from experimental results that RAN is capable of recognizing unseen Chinese characters with visualization of spatial attention and outperforms traditional whole-character based approaches on seen Chinese character recognition. In future work, we plan to investigate RAN's ability in recognizing handwriting Chinese characters or Chinese characters in natural scene. We will also explore the
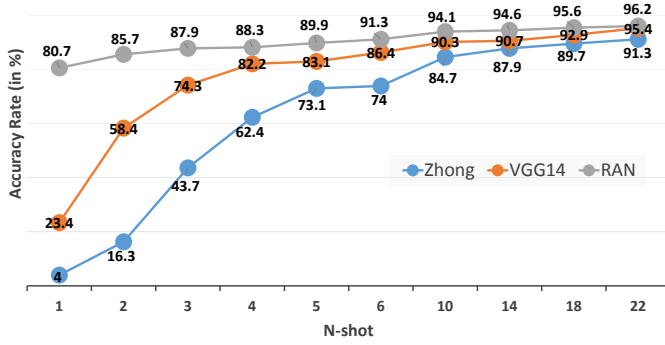
**Fig. 9**: The performance comparison among Zhong, VGG14 and RAN with respect to the number of newly added training fonts N.

effect of mapping relationship between structure/radical and Chinese character in improving the performance of RAN.

## 6. ACKNOWLEDGMENTS

## 7. REFERENCES

[1] Cheng-Lin Liu, Stefan Jaeger, and Masaki Nakagawa, "Online recognition of Chinese characters: the state-of-the-art," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 2, pp. 198–213, 2004.

[2] Xiaotong Li and Xiaoheng Zhang, "The writing order of modern Chinese character components," *The journal of modernization of Chinese language education*, 2013.

[3] Long-Long Ma and Cheng-Lin Liu, "A new radical-based approach to online handwritten Chinese character recognition," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*. IEEE, 2008, pp. 1–4.

[4] An-Bang Wang and Kuo-Chin Fan, "Optical recognition of handwritten Chinese characters by hierarchical radical matching method," *Pattern Recognition*, vol. 34, no. 1, pp. 15–35, 2001.

[5] Tie-Qiang Wang, Fei Yin, and Cheng-Lin Liu, "Radical-based Chinese character recognition via multi-labeled learning of deep residual networks," in *Document Analysis and Recognition (ICDAR), 2017 14th International Conference on*. IEEE, 2017.

[6] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks,"

[8] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[9] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[10] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3156–3164.

[11] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International Conference on Machine Learning*, 2015, pp. 2048–2057.

[12] Lianli Gao, Zhao Guo, Hanwang Zhang, Xing Xu, and Heng Tao Shen, "Video captioning with attention-based LSTM and semantic consistency," *IEEE Transactions on Multimedia*, vol. 19, no. 9, pp. 2045–2055, 2017.

[13] Jianshu Zhang, Jun Du, Shiliang Zhang, Dan Liu, Yulong Hu, Jinshui Hu, Si Wei, and Lirong Dai, "Watch, attend and parse: An end-to-end neural network based approach to handwritten mathematical expression recognition," *Pattern Recognition*, vol. 71, pp. 196–206, 2017.

[14] Jianshu Zhang, Jun Du, and Lirong Dai, "A gru-based encoder-decoder approach with attention for online handwritten mathematical expression recognition," in *Document Analysis and Recognition (ICDAR), 2017 14th International Conference on*. IEEE, 2017.

[15] Jianshu Zhang, Jun Du, and Lirong Dai, "Multi-scale attention with dense encoder for handwritten mathematical expression recognition," *arXiv preprint arXiv:1801.03530*, 2018.

[16] Yoshua Bengio, Patrice Simard, and Paolo Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994.

[17] Jianshu Zhang, Jian Tang, and Li-Rong Dai, "RNN-BLSTM based multi-pitch estimation," in *INTERSPEECH 2016*, 2016, pp. 1785–1789.

[18] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

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

[20] Zhuoyao Zhong, Lianwen Jin, and Ziyong Feng, "Multi-font printed Chinese character recognition using multi-pooling convolutional neural network," in *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*. IEEE, 2015, pp. 96–100.