# Writer Adaptive Feature Extraction Based on Convolutional Neural Networks For Online Handwritten Chinese Character Recognition

Jun Du[1], Jian-Fang Zhai[1], Jin-Shui Hu[2], Bo Zhu[2], Si Wei[2], Li-Rong Dai[1]
[1]University of Science and Technology of China, Hefei, Anhui, P. R. China
[2]iFlytek Research, Hefei, Anhui, P. R. China
Emails: jundu@ustc.edu.cn, {jfzhai,jshu,bozhu,siwei}@iflytek.com ,lrdai@ustc.edu.cn

***Abstract*—This paper presents a novel approach to writer adaptation based on convolutional neural network (CNN) as a feature extractor and improved discriminative linear regression for online handwritten Chinese character recognition. First, the proposed recognizer consisting of CNN-based feature extractor and prototype-based classifier can achieve comparable performance with the state-of-the-art CNN-based classifier while it could be designed more compact and efficient as a practical solution. Second, the writer adaption is performed via a linear transformation of the extracted feature from CNN. The transformation parameters are optimized with a so-called sample separation margin based minimum classification error criterion, which can be further improved by using more synthesized adaptation data and a simple regularization method. The experiments on the data collected from user inputs of Smartphones with a vocabulary of 20,936 characters demonstrate that our writer adaptation approach can yield significant improvements of recognition accuracy over a high-performance baseline system and also outperform a state-of-the-art approach based on style transfer mapping especially with increased adaptation data.**

***Keywords*—convolutional neural network, discriminative linear regression, synthesized data, regularization, handwritten Chinese character recognition**

## I. Introduction

Online handwritten Chinese character recognition as an input mode on a mobile device (e.g., Smartphone, Tablet) becomes more and more popular in the mobile internet era. Several off-the-shelf solutions could be developed to build the recognizer for online handwritten Chinese characters (e.g., [1], [2], [3]). But the user experience is not always satisfactory in real applications due to the large variability of writing styles by different writers, especially the cursive writing style. one of solutions to address this problem is writer adaptation which aims to improve the recognition performance and user experience of a single writer by using the corresponding data samples to be recognized itself via an *unsupervised adaptation* strategy, or by using a small amount of adaptation data samples with labels collected from the target writer via a *supervised adaptation* strategy, which is also the focus of this study.

Many writer adaptation approaches have been proposed in the past several decades. For example, in [4], a writer-adaptable online character recognizer was designed via a time delay neural network where the last layer is served as a linear optimal hyperplane classifier for adapting to new writing styles. Platt et al. [5] placed a so-called output adaptation module using a radial basis function (RBF) network on top of standard neural networks. In [6], a hidden Markov model (HMM) based recognition system for cursive German script could be adapted to a new writing style using maximum likelihood linear regression (MLLR) or maximum a posteriori (MAP) criterion. Vuori and Korkeakoulu [7] proposed several strategies for adaptation of a prototype-based classifier, including adding new prototypes, reshaping existing prototypes, and inactivating poorly performing prototypes. In [8], a biased regularization for support vector machine (SVM) based classifier is adopted for personalization.

Most of these approaches are designed for the western languages, e.g. English, in which there are only dozens of character classes. In this paper, we focus on the writer adaptation techniques for the task consisting of thousands of character classes, namely online handwritten Chinese character recognition. One of the product solutions to build a Chinese handwriting recognizer is to train a prototype-based classifier as reported in [3] via a so-called sample separation margin based minimum classification error (SSM-MCE) criterion [9]. One advantage using such a classifier is that it can be made both compact [10] and efficient [11] in the recognition stage for the task with a large vocabulary of Chinese characters. One class of the adaptation methods based on the prototype-based classifier is to use a linear feature transformation for adapting the writing styles via different criteria, e.g., style transfer mapping (STM) with a least regularized weighted squared error criterion [12], [13], or discriminative linear regression (DLR) with SSM-MCE criterion [14], [15], [16]. But recently, convolutional neural network (CNN) originally proposed in [17] as a classifier makes a new milestone for both online and offline handwritten Chinese character recognition in terms of recognition accuracy [18], [19], [20]. In the 2013 Chinese handwriting recognition competition task, the system from University of Warwick [21] using a deep CNN achieved the best performance on the online character recognition task while the system from the Dalle Molle Institute for Artificial Intelligence (IDSIA) using multi-column deep CNNs [22] and the system from Fujitsu R&D Center using multiple CNNs [23] reported the two best results for the offline character recognition task. So the motivation of this study is to further improve the recognition accuracy on top of the high-performance CNN-based classifier by writer adaptation. First, our proposed recognizer can be considered as a hybrid version of CNN-based and prototype-based recognizers in which CNN

978-1-4799-1805-8/15/$31.00 ©2015 IEEE

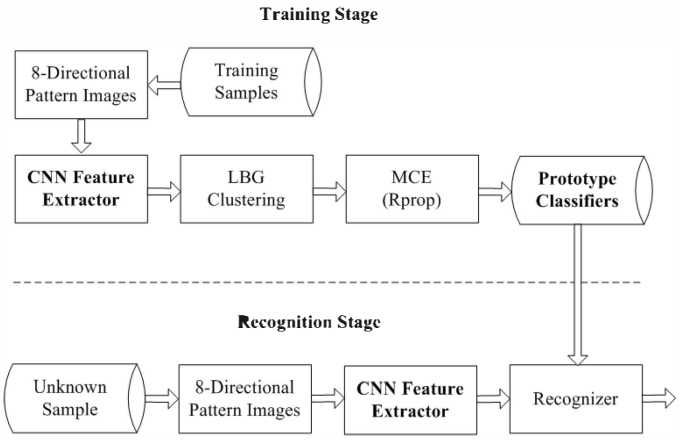


Fig. 1. Overall development flow and architecture.

with the convolutional layers is adopted as a feature extractor while the last fully connected layer is replaced by prototype-based classifier. This design can maintain the high recognition accuracy of CNN-based classifier while many well-established techniques associated with the prototype-based classifier can be adopted as a practical solution. Second, for writer adaptation, an improved DLR approach to prototype-based classifier is used. The main differences of this work in comparison to our recent work in [16] are: 1) We demonstrate that our writer adaptation approach can achieve consistently significant improvements over the new recognizer based on CNN feature extraction with a much higher baseline performance than those deep neural network (DNN) based recognizers in [16]; 2) The DLR based adaptation approach in [16] is improved by using more synthesized adaptation data and a simple regularization method; 3) Most of the training data are collected from mobile internet users rather than the in-house corpus in [16], which leads to a further improved baseline recognition accuracy and makes the adaptation more difficult. Furthermore the vocabulary of the recognition task is enlarged from 15,167 characters to 20,936 characters.

The remainder of the paper is organized as follows. In Section II, the detailed descriptions of recognizer with both the CNN-based feature extractor and prototype-based classifier are presented. In Section III, the writer adaptation approach is described. In Section IV, we report experimental results. Finally we conclude the paper in Section V.

## II. The Recognizer Description

An overall development flow and architecture of our system is illustrated in Fig. 1. In the training stage, first 8-directional pattern images are generated based on each training sample according to [24] as the input of CNN-based feature extractor. Then the final feature vector is extracted from the input of the last fully connected layer of CNN. Finally, based on the CNN features, a multi-prototype based classifier is constructed by using LBG clustering algorithm, which is then refined by SSM-MCE training with an Rprop algorithm. The main difference of this training procedure from [3] is that the conventional features based on linear discriminant analysis (LDA) transformation of 8-directional raw features are replaced by the highly nonlinear CNN-based features. At the recognition stage, with the CNN feature vector extracted from the unknown sample, the normal recognition is performed based on prototype-based classifier. In the following subsections, we elaborate on two highlighted modules, namely CNN-based feature extractor and prototype-based classifier training.

### A. CNN-based feature extractor

We adopt a modified CNN architecture with alternating convolutional and max-pooling layers proposed in the [21], which is inspired by [25], [26]. 8 directional pattern images plus one image with original ink trajectories are collected for the input layer, which is one alternative to the signature representation mentioned in [21]. Each node in the output layer corresponds to one character class. After CNN training, only the parameters before the fully connected layer are left to extract the final feature vector which is fed to the prototype-based classifier described in the next section. The CNN-based feature extractor which is irrelevant to the number of the character classes could be designed very compact.

### B. Prototype-based classifier training

Suppose our prototype-based classifier can recognize $M$ character classes denoted as $\{C_i|i = 1, ..., M\}$ and each class $C_i$ is represented by a set of $K_i$ prototypes $\lambda_i = \{\mathbf{m}_{ik} \in \mathcal{R}^D|k = 1, ..., K_i\}$, where $\mathbf{m}_{ik}$ is the $k^{\text{th}}$ prototype of the $i^{\text{th}}$ class with $D$ dimension. Let's use $\mathbf{\Lambda} = \{\lambda_i\}$ to denote the set of prototypes for all classes. In the classification stage, a feature vector $\mathbf{x} \in \mathcal{R}^D$ is first extracted from CNN. Then the measurement between $\mathbf{x}$ and each class $C_i$ is evaluated by a Euclidean distance based discriminant function as follows

$$g_i(\mathbf{x}; \lambda_i) = -\min_k \parallel \mathbf{x} - \mathbf{m}_{ik} \parallel^2 \ . \tag{1}$$

Finally, the class with the maximum discriminant function score is chosen as the recognized class $r(\mathbf{x}; \mathbf{\Lambda})$, i.e.,

$$r(\mathbf{x}; \mathbf{\Lambda}) = \arg\max_i g_i(\mathbf{x}; \lambda_i) \ . \tag{2}$$

In the training stage, given a set of training feature vectors $\mathcal{X} = \{\mathbf{x}_r \in \mathcal{R}^D|r = 1, ..., R\}$, first we initialize $\mathbf{\Lambda}$ by LBG clustering [28]. Then $\mathbf{\Lambda}$ can be refined by minimizing the following SSM-MCE objective function:

$$l(\mathcal{X}; \mathbf{\Lambda}) = \frac{1}{R} \sum_{r=1}^{R} \frac{1}{1 + \exp[-\alpha d(\mathbf{x}_r; \mathbf{\Lambda}) + \beta]} \tag{3}$$

where $\alpha$, $\beta$ are two control parameters, and $d(\mathbf{x}_r; \mathbf{\Lambda})$ is a misclassification measure defined by a so-called sample separation margin (SSM) as follows [9]:

$$d(\mathbf{x}_r; \mathbf{\Lambda}) = \frac{-g_p(\mathbf{x}_r; \lambda_p) + g_q(\mathbf{x}_r; \lambda_q)}{2 \parallel \mathbf{m}_{p\hat{k}} - \mathbf{m}_{q\bar{k}} \parallel} \tag{4}$$

where $q$ is the most competing class label excluding the true label $p$ of $\mathbf{x}_r$. $\hat{k}$ and $\bar{k}$ are the indices with the minimum Euclidean distance between $\mathbf{x}_r$ and the prototypes of class $p$ and $q$, respectively. To optimize the objective function in Eq. (3) with the parameter set $\mathbf{\Lambda}$, the Rprop algorithm is adopted as described in [3].

## III. Writer Adaptive Feature Extraction

In this work, we focus on the supervised writer adaptation as in most cases the labels of adaptation samples could be provided by the users when they select the true label from the recognition candidates displayed on the interface of the handwriting application. Suppose we are given a set of labeled adaptation data $\mathcal{Y} = \{\mathbf{y}_r \in \mathcal{R}^D | r = 1, ..., R'\}$ collected from a single writer. Then a linear feature transformation of the CNN-based feature vector is adopted for writer adaptation:

$$\mathbf{x}_r = \mathcal{F}(\mathbf{y}_r; \mathbf{\Theta}) = \mathbf{A}\mathbf{y}_r + \mathbf{b} \tag{5}$$

where $\mathbf{\Theta} = \{\mathbf{A}, \mathbf{b}\}$ denotes the set of transform parameters. $\mathbf{A}$ is a $D \times D$ nonsingular matrix and $\mathbf{b}$ is a $D$-dimensional bias vector. $\mathbf{y}_r$ and $\mathbf{x}_r$ are the $r^{\text{th}}$ $D$-dimensional input and transformed feature vectors, respectively. In the recognition stage, the estimated parameters $\{\mathbf{A}, \mathbf{b}\}$ are used to transform the feature vector of each unknown sample first, which is then fed to the classifier for recognition. In the following subsections, three approaches are elaborated to learn the parameters of the linear transformation.

### A. Style transfer mapping

One popular approach is the style transfer mapping proposed in [12], [13]. In STM, first we define the source point set as the set of feature vectors of adaptation samples (i.e., $\mathcal{Y}$), and the target point set as the set of the corresponding prototypes with the minimum Euclidean distances to those features vectors. Then a style transfer matrix $\mathbf{A}^{\text{STM}}$ could be found to solve the following optimization problem with a least regularized squared error criterion:

$$\min_{\mathbf{A}^{\text{STM}}} \sum_{r=1}^{R'} \|\mathbf{A}^{\text{STM}}\mathbf{s}_r - \mathbf{t}_r\|_2^2 + \beta_1 \|\mathbf{A}^{\text{STM}} - \mathbf{I}\|_2^2 \tag{6}$$

where the $r^{\text{th}}$ source point $\mathbf{s}_r$ is transformed to the target point $\mathbf{t}_r$. The hyperparameter $\beta_1$ is set as

$$\beta_1 = \frac{\tilde{\beta}_1}{2D} \text{tr}\left(\sum_{\text{r}} (\mathbf{s}_\text{r} + \mathbf{t}_\text{r})\mathbf{s}_\text{r}^\top\right) \tag{7}$$

where $\text{tr}(\cdot)$ is the trace of a matrix, and $\tilde{\beta}_1$ takes a value between 0 and 3. A closed-form solution of the above problem can be obtained as follows:

$$\mathbf{A}^{\text{STM}} = \left[\sum_r \mathbf{t}_r\mathbf{s}_r^\top + \beta_1\mathbf{I}\right]\left[\sum_r \mathbf{s}_r\mathbf{s}_r^\top + \beta_1\mathbf{I}\right]^{-1} . \tag{8}$$

### B. Discriminative linear regression

STM-based approach is quite effective when there is only a small amount of adaptation data. But the recognition performance is easily saturated with increased adaptation data. Another approach, namely the discriminative linear regression, which is first proposed in [15] for Chinese OCR adaptation, is also verified to be effective for handwriting recognition [16], especially with a large amount of adaptation data. To learn the transformation $\mathbf{\Theta}^{\text{DLR}} = \{\mathbf{A}^{\text{DLR}}, \mathbf{b}^{\text{DLR}}\}$, the same SSM-MCE objective function as the criterion of the classifier training is defined as follows:

$$l(\mathcal{Y}; \mathbf{\Lambda}, \mathbf{\Theta}^{\text{DLR}}) = \frac{1}{R'} \sum_{r=1}^{R'} \frac{1}{1 + \exp[-\alpha d(\mathbf{y}_r; \mathbf{\Lambda}, \mathbf{\Theta}^{\text{DLR}}) + \beta]} \tag{9}$$

where

$$d(\mathbf{y}_r; \mathbf{\Lambda}, \mathbf{\Theta}^{\text{DLR}}) = \frac{-g_p(\mathbf{x}_r; \lambda_p) + g_q(\mathbf{x}_r; \lambda_q)}{2 \parallel \mathbf{m}_{p\hat{k}} - \mathbf{m}_{q\bar{k}} \parallel} . \tag{10}$$

$\mathbf{x}_r$ in Eq. (10) is defined in Eq. (5). The optimization procedure for $\mathbf{\Theta}^{\text{DLR}}$ is the same as in [15]. From the viewpoint of classification measure, SSM-MCE seems a more reasonable objective function to learn the feature transform compared with STM, which is also confirmed by our experiments.

### C. Improved discriminative linear regression (IDLR)

To improve the regularization of DLR-based adaptation and fully tap its potential with large adaptation data, two strategies are presented. The first one is a simple linear interpolation between STM-based and DLR-based transformations as a regularization to DLR:

$$\mathbf{A}^{\text{IDLR}} = \beta_2\mathbf{A}^{\text{DLR}} + (1 - \beta_2)\mathbf{A}^{\text{STM}} \tag{11}$$

where the factor $\beta_2$ is adaptive with the number of adaptation data $R'$ and can be empirically set as:

$$\beta_2 = 0.5 + 0.1 * \log_2 \frac{R'}{N_T} . \tag{12}$$

$N_T$ is a threshold for the number of adaptation data $R'$. Eq. (12) only holds for $\beta_2$ in the range [0,1]. If $\beta_2 < 0$, then it will be set to 0, which implies that we only use the STM-based transformation with a very few amount of adaptation data. If $\beta_2 > 1$, then it will be set to 1, which indicates that only the DLR-based transformation is adopted with quite a large amount of adaptation data.

The second strategy is a simple perturbation, which can synthesize more adaptation data by using deformations of handwriting characters. This is motivated by the recent success of using distorted samples to improve the generalization performance of DNN-based classifier [29] and other classifiers [30]. We believe that the perturbation could also work for writer adaptation by carefully designing the deformations. In this work, the distorted operations including rotation, shearing, and scaling are randomly performed on the original adaptation samples.

## IV. Experiments

The experiments are conducted on the task of recognizing isolated online handwritten Chinese characters with a vocabulary of 20,936 character classes. For training, we use about 1,000 samples per character class. The training data is collected from a large amount of real users, which is quite different from the in-house corpus used in our recent work [16]. As for the adaptation and test data, the samples of 200 new users collected in several months are used. For each writer, one half of samples are randomly selected as adaptation data to learn the feature transformation parameters while the other half is used as test data. The configuration of CNN is that the input layer is 48×48×9 pattern images, the dimension of the final feature vector from the fully connected layer is 100, and the output layer is 20,936 character classes. For Rprop-based SSM-MCE training and adaptation, the setting of the control parameters can refer to [3], [15]. For IDLR, $N_T$ is set to 2,048 and we use 50-fold distorted samples for perturbation. The number

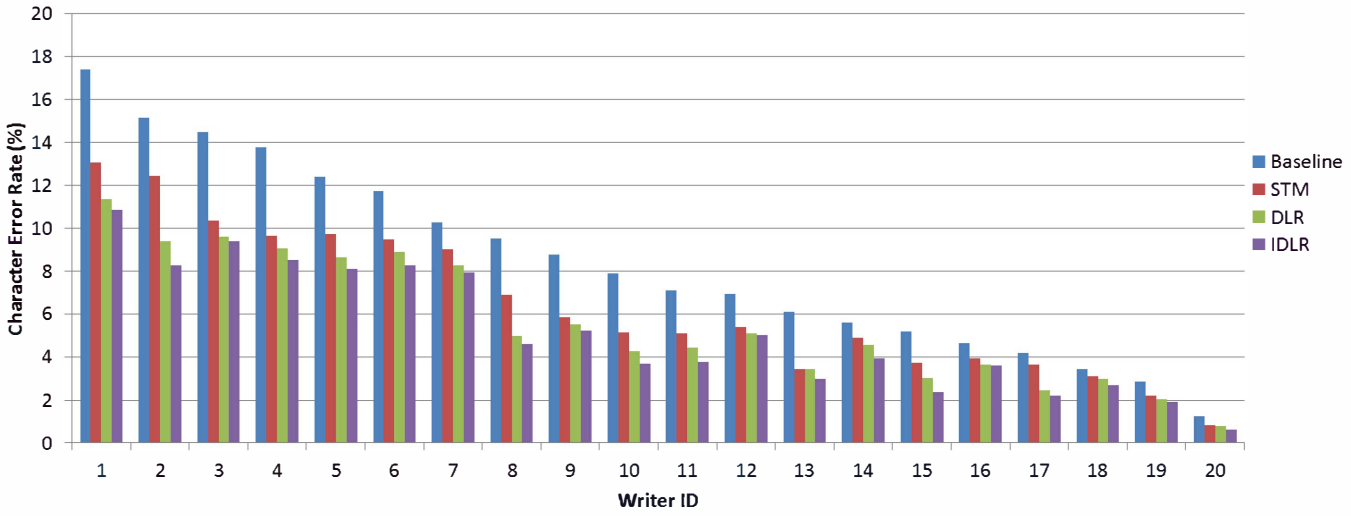


Fig. 2. Performance (character error rate in %) comparison of different adaptation approaches for each test set of 20 selected writers.

TABLE I. PERFORMANCE (CHARACTER ERROR RATE IN %) COMPARISON OF TWO CNN-BASED SYSTEMS AVERAGED ON ALL 200 WRITERS OF THE TEST SET WITHOUT WRITER ADAPTATION.

| System | CNN-MP | CNN |
|---|---|---|
| CER | 5.83 | 5.63 |

TABLE II. PERFORMANCE (CHARACTER ERROR RATE IN %) COMPARISON OF THREE ADAPTATION APPROACHES WITH DIFFERENT AMOUNT OF ADAPTATION DATA ON THE TEST SET.

| | 200 | 500 | 1000 | 6000 |
|---|---|---|---|---|
| STM | 4.8 | 4.6 | 4.5 | 4.43 |
| DLR | 4.77 | 4.5 | 4.31 | 3.91 |
| IDLR | 4.7 | 4.42 | 4.18 | 3.68 |

of prototypes for each character class is non-uniformly set as the maximum of 4 for commonly used characters and the minimum of 1 for uncommonly used characters. To handle the large-scale training data, the computations of LBG clustering, SSM-MCE training and adaptation with Rprop algorithm are parallelized on the CPU cluster while CNN training is performed on GPUs.

Table I shows a performance comparison of two CNN-based systems averaged on all 200 writers of the test set without writer adaptation. The "CNN-MP" system uses the multi-prototype based classifier with CNN-based feature extraction proposed in this work. "CNN" system is a conventional purely CNN-based classifier. Our proposed "CNN-MP" system is slightly worse than "CNN" system but can be designed more compact and efficient as a practical solution.

Table II lists a performance comparison of three adaptation approaches with different amount of adaptation data averaged on 200 writers of the test set. The baseline system without adaptation is "CNN-MP" in Table I. Four configurations with different number of adaptation samples are compared, namely 200, 500, 1,000, and 6,000. First, on top of the high-performance baseline with a 5.83% CER, the STM-based writer adaptation with more than 500 adaptation samples can achieve more than 20% relative error rate reduction. Second, with the increased amount of adaptation data, the performance of STM-based approach is quickly saturated while the error rates of both DLR and IDLR are still significantly decreased. We can imagine that the performance gap between IDLR and STM will be larger with more adaptation data, which results in a better user experience. Finally, the effectiveness of IDLR can be verified by the observation that the relative performance gain of IDLR from DLR is consistently comparable to that between DLR and STM for different amount of adaptation data. Overall, a 16.9% relative character error rate reduction is yielded by IDLR over STM with 6,000 adaptation samples.

Fig. 2 gives a performance comparison of different adaptation approaches for each test set of 20 selected writers sorted by the baseline recognition performance. For each writer, no more than 6,000 adaptation samples are used. In general, the similar observations as in Table II for each writer can be made with the same order sorted by character error rate, namely Baseline>STM>DLR>IDLR. There is only one exception on No.13 writer where the performance of DLR is the same as STM. Although the baseline recognition error rate varies from 17.37% to 1.27% for different writers, IDLR-based adaptation achieves consistently the best recognition performance, with the relative character error rate reductions of 54% at most and 22% at least over the baseline system.

## V. CONCLUSION

In this work, we investigate on the writer adaptation for a high-performance and well-designed CNN-based recognizer. The experiments are performed on real user data including training, adaptation, and test sets. STM-based adaptation is quite effective with a small amount of adaptation data while DLR approach can significantly improve the recognition accuracy over STM approach with more adaptation data. Furthermore, our proposed IDLR approach by using distorted samples and a simple regularization can yield consistently performance gains over DLR approach. As for the future work, we aim to incorporate the adaptation into the designing of new CNN architectures to further improve the writer adaptation performance.

## VI. Acknowledgment

This work was supported by the National Natural Science Foundation of China under Grants No. 61305002. We also thank Jia-Jia Wu from iFlytek Research for the help of training the CNN.

## References

[1] C.-L. Liu, H. Sako, and H. Fujisawa, "Discriminative learning quadratic discriminant function for handwriting recognition," *IEEE Trans. on Neural Networks*, Vol. 15, No. 2, pp.430-444, 2004.

[2] Y.-Q. Wang and Q. Huo, "Building compact recognizers of handwritten Chinese characters using precision constrained Gaussian model, minimum classification error training and parameter compression," *International Journal on Document Analysis and Recognition*, Vol. 14, No. 3, pp.255-262, 2011.

[3] J. Du and Q. Huo, "Designing compact classifiers for rotation-free recognition of large vocabulary online handwritten Chinese characters," *Proc. ICASSP-2012*, pp.1721-1724.

[4] N. Matić, I. Guyon, J. Denker, and V. Vapnik, "Writer adaptation for on-line handwritten character recognition," *Proc. ICDAR-1993*, pp.187-191.

[5] J. C. Platt and N. P. Matić, "A constructive RBF netwrok for writer adaptation," *Proc. NIPS-1997*.

[6] A. Brakensiek, A. Kosmala, and G. Rigoll, "Comparing adaptation techinieuqs for on-line handwriting recognition," *Proc. ICDAR-2001*, pp.486-490.

[7] V. Vuori and T. Korkeakoulu, *Adaptive methods for online recognition of isolated handwritten characters*, PhD thesis, Helsinki University of Technology, 2002.

[8] W. Kienzle and K. Chellapilla, "Personalized handwriting recognition via biased regularization," *Proc. ICML-2006.*

[9] T. He and Q. Huo, "A study of a new misclassification measure for minimum classification error training of prototype-based pattern classifiers," *Proc. ICPR-2008*.

[10] Y.-Q. Wang and Q. Huo, "A study of designing compact recognizers of handwritten Chinese characters using multiple-prototype based classifiers," *Proc. ICPR-2010*, pp.1872-1875.

[11] Z.-D. Feng and Q. Huo, "Confidence guided progressive search and fast match techniques for high performance Chinese/English OCR," *Proc. ICPR-2002*, pp.III-89-92.

[12] X.-Y. Zhang and C.-L. Liu, "Style transfer matrix learning for writer adaptation," *Proc. CVPR-2011*, pp.393-400.

[13] X.-Y. Zhang and C.-L. Liu, "Writer adaptation with style transfer mapping," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 35, No. 7, pp.1773-1787, 2013.

[14] J. Du and Q. Huo, "A discriminative linear regression approach to OCR adaptation," *Proc. ICPR-2012*, pp.629-632.

[15] J. Du and Q. Huo, "A discriminative linear regression approach to adaptation of multi-prototype based classifiers and its applications for Chinese OCR," *Pattern Recognition*, Vol. 46, No. 8, pp.2313-2322, 2013.

[16] J. Du, Jin-Shui Hu, Bo Zhu, Si Wei, and Li-Rong Dai, "Writer adaptation using bottleneck features and discriminative linear regression for online handwritten Chinese character recognition," *Proc. ICFHR-2014*.

[17] L. Yann, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, Vol. 11, pp.2278-2324, 1986.

[18] D. C. Ciresan, U. Meier, L. M. Gambardella, and J. Schmidhuber, "Convolutional neural network committees for handwritten character classification," *Proc. ICDAR-2011.*

[19] C.-L. Liu, F. Yin, Q.-F. Wang, and D.-H. Wang, "ICDAR 2011 Chinese handwriting recognition competition," *Proc. ICDAR-2011*, pp.1464-1469.

[20] F. Yin, Q.-F. Wang, X.-Y. Zhang, and C.-L. Liu, "ICDAR 2013 Chinese handwriting recognition competition," *Proc. ICDAR-2013*, pp.1464-1470.

[21] B. Graham, "Sparse arrays of signatures for online character recognition," *Technical Report*, University of Warwick, 2013.

[22] D. C. Ciresan and J. Schmidhuber, "Multi-column deep neural networks for offline handwritten Chinese character classification," *Preprint arXiv:1309.0261*, 2013.

[23] C. Wu, W. Fan, Y. He, J. Sun, and S. Naoi, "Handwritten character recognition by alternately trained relaxation convolutional neural network," *Proc. ICFHR-2014*.

[24] Z.-L Bai and Q. Huo, "A study on the use of 8-directional features for online handwritten Chinese character recognition," *Proc. ICDAR-2005*, pp.262-266.

[25] D. C. Ciresan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, "Flexible, high performance convolutional neural networks for image classification," *Proc. IJCAI-2011*, pp.1237-1242.

[26] D. Scherer, A. Müller, and S. Behnke, "Evaluation of pooling operations in convolutional architectures for object recognition," *Proc. International Conference on Artificial Neural Networks*, 2010.

[27] M. Lin, Q. Chen, and S.-C Yan, "Network in network," *Preprint arXiv:1312.4400v3*, 2014.

[28] Y. Linde, A. Buzo and R. Gray, "An algorithm for vector quantizer design," *IEEE Trans. on Communications*, Vol. 28, No. 1, pp.84-95, 1980.

[29] D. Ciresan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," *Proc. CVPR-2012*, pp.3643-3649.

[30] F. Yin, M.-K. Zhou, Q.-F. Wang, and C.-L. Liu, "Style consistent perturbation for handwritten Chinese character recognition," *Proc. ICDAR-2013*, pp.1051-1055.