2014 14th International Conference on Frontiers in Handwriting Recognition

Irrelevant Variability Normalization via Hierarchical Deep Neural Networks for Online Handwritten Chinese Character Recognition

Jun Du
University of Science and Technology of China, Hefei, Anhui, P. R. China
jundu@ustc.edu.cn

Abstract

This paper presents a novel irrelevant variability normalization (IVN) approach via hierarchical deep neural networks (HDNNs) and prototype-based classifier for online handwritten Chinese character recognition. The recent insight of deep neural network (DNN) is the deep architecture with large training data can bring the best performance in many research areas. The architecture design of our proposed hierarchical deep neural networks focuses on both "depth" and "width" of artificial neural network. Specifically for the multivariate regression, HDNN consists of multiple subnets, which is empirically more powerful than DNN. In this work, HDNN is adopted as a nonlinear feature transform to normalize the feature vector of handwritten samples with irrelevant variabilities to a target prototype. The effectiveness of proposed method is verified on a Chinese handwriting recognition task. Furthermore, we have an very interesting observation that DNN-based IVN can not even bring performance gain over the prototype-based classifier while HDNN-based IVN yields significant improvements of recognition accuracy.

1. Introduction

In the mobile internet era, using online handwritten Chinese character recognition as an input mode on a portable device has been becoming increasingly popular. Several solutions have been developed to build product engines for online handwritten Chinese character recognition (e.g., [18, 6, 21]). But in real applications, there are many irrelevant variabilities (e.g., writing styles) in training/testing samples, which may lead to degradation of recognition performance. In this study, we adopt the concept of *irrelevant variability normalization* (IVN) [16] to tackle the above problem.

In [1], a so-called speaker adaptive training (SAT) approach was proposed to normalize speaker variability in training hidden Markov models (HMMs) for automatic speech recognition (ASR). The concept of SAT was generalized to deal with any variabilities irrelevant to phonetic classification in [16], therefore a term of IVN training was coined. Since then, many variants of IVN training methods have been tried in ASR area [26, 23]. Only recently, the concept of IVN was tried in the area of handwriting recognition. For example, in [4], writer adaptive training (WAT) using constrained maximum likelihood linear regression (CMLLR) [10] based feature transform was studied for an HMM-based Arabic handwriting recognition task. Region-dependent feature transform in [23] was also applied to HMM-based off-line handwriting recognition in [3]. In [24] a pattern field classification approach with style normalized transformation was proposed and demonstrated to be effective for several pattern recognition applications, including handwritten Chinese character recognition. More recently, we presented an IVN approach to jointly discriminative training of linear feature transforms and multi-prototype based classifier for recognition of online handwritten Chinese characters [8].

In this paper, we study a new IVN approach to normalize the generally irrelevant variabilities via a highly nonlinear feature transform, namely hierarchical deep neural networks (HDNNs), rather than the widely used linear transforms [25, 24, 5, 8, 7], for online handwritten Chinese character recognition. One of the state-of-the-art techniques to build a Chinese handwriting recognizer is to use a so-called sample separation margin based minimum classification error (SSM-MCE) criterion [11] to train a prototype-based classifier as reported in [20, 6]. In spite of the large vocabulary of Chinese characters, such a classifier can be made both compact [20] and efficient [9] in the recognition stage. In this work, based on this classifier, we propose to use HDNN for normalizing the irrelevant variabilities

2167-6445/14 $31.00 © 2014 IEEE
DOI 10.1109/ICFHR.2014.58

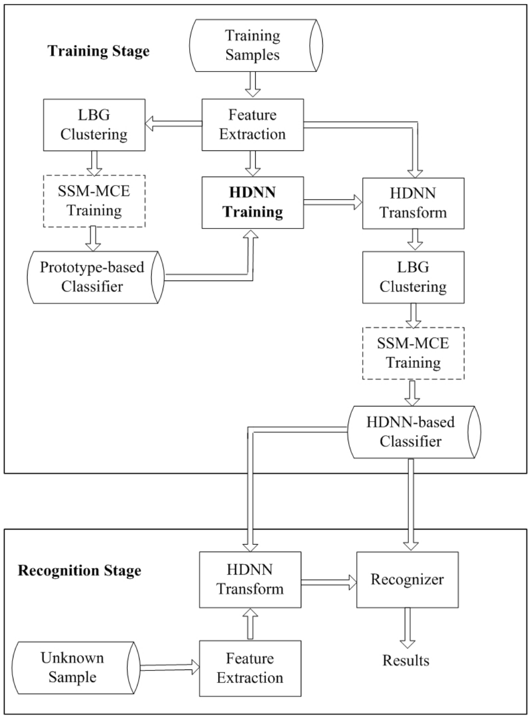

Figure 1. Overall development flow.

in handwritten samples. Recently, deep neural networks (DNNs) as classification or regression models are widely used in many areas, especially speech area [15, 14, 13, 22]. But our experiments show that DNN as a regression model for IVN can even lead to degradation of recognition performance mainly because DNN can not well learn the highly nonlinear regression function between high-dimensional feature vectors. Inspired by this, HDNN, consisting of multiple subnets which is empirically more powerful than DNN, is proposed and verified to be effective for IVN in Chinese handwriting recognition.

The remainder of the paper is organized as follows. In Section 2, we present the detailed description of prototype-based classifier. In Section 3, IVN via HDNN as a feature transform is described. In Section 4, we report experimental results. Finally we conclude the paper in Section 5.

2. System Description

An overall system development flow and architecture of IVN via HDNN is illustrated in Fig. 1. In the training stage, first a raw feature vector is extracted from each training sample [2], which is followed by LDA transformation to obtain a lower dimensional feature vector. After that, the prototype-based classifier is constructed by using LBG clustering algorithm [17], which can be refined by SSM-MCE training. With prototypes for each character class and feature vectors of training samples, HDNN training is performed to learn the mapping function between the feature vector of each character sample and its corresponding prototype. Then new transformed features are generated via HDNN, which are fed to the prototype-based classifier training. At the recognition stage, with the feature vector extracted from the unknown sample, feature transform via HDNN is conducted. Finally the transformed feature vector is fed to recognizer. The details of classifier training are elaborated in the following subsection while IVN via HDNN is described in Section 3.

2.1. Prototype-based classifier

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

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

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 [17]. Then $\mathbf{\Lambda}$ can be re-estimated 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 α, β are two control parameters, and $d(\mathbf{x}_r;\mathbf{\Lambda})$ is a misclassification measure defined by using a so-called sample separation margin (SSM) as follows [11]:

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

where

$$\begin{aligned} \hat{k} &= \arg\min_k \parallel \mathbf{x}_r - \mathbf{m}_{pk} \parallel^2 && (5) \\ q &= \arg\max_{i \in \mathcal{M}_r} g_i(\mathbf{x}_r;\lambda_i) && (6) \\ \bar{k} &= \arg\min_k \parallel \mathbf{x}_r - \mathbf{m}_{qk} \parallel^2 && (7) \end{aligned}$$

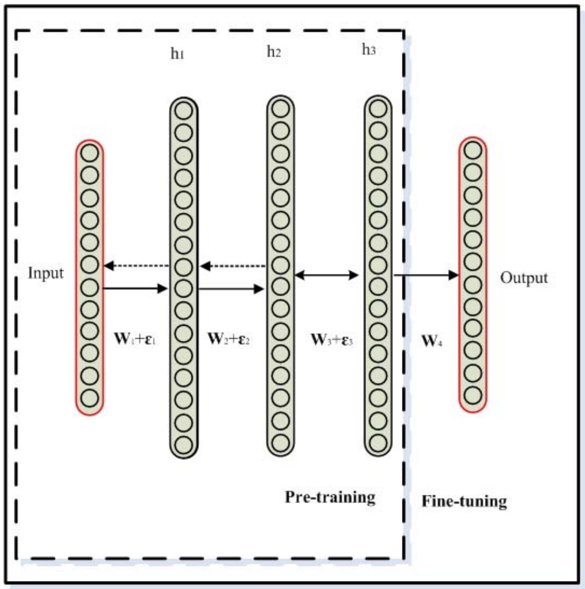

Figure 2. DNN.

and $\mathcal{M}_r$ is the hypothesis space for the r^{th} sample, excluding the true label p.

To optimize the objective function in Eq. (3), the same implementation of Quickprop algorithm as described in [20] is adopted here.

3. Irrelevant Variability Normalization via Hierarchical Deep Neural Networks

In this study, the concept of IVN is implemented by using feature transformation:

$$\mathbf{x}_r^{\text{ivn}} = \mathcal{F}(\mathbf{x}_r; \boldsymbol{\Theta}) \tag{8}$$

where $\mathbf{x}_r$ and $\mathbf{x}_r^{\text{ivn}}$ are the r^{th} D-dimensional input and transformed feature vectors, respectively. To learn the mapping function $\mathcal{F}$ with the set of parameters $\boldsymbol{\Theta}$, we aim at minimizing mean squared error function defined as:

$$E = \frac{1}{R}\sum_{r=1}^{R} \|\mathbf{x}_r^{\text{ivn}} - \mathbf{x}_r^{\text{ref}}\|_2^2 \tag{9}$$

where $\mathbf{x}_r^{\text{ref}}$ is the reference feature vector, which is set as the prototype with the smallest Euclidean distance to $\mathbf{x}_r^{\text{ivn}}$ for the corresponding character class. Ideally if the input feature vector can be transformed to the corresponding prototype in that class, then the recognition results are always correct. In the following subsections, two specific forms of $\mathcal{F}$, namely DNN and HDNN are introduced. Also we will discuss the implementation issues of HDNN.

3.1. Deep Neural Network

A deep neural network (DNN) is a feed-forward, artificial neural network that has more than one layer of hidden units between its inputs and outputs [13]. In this work, DNN is adopted as a multivariate regression model to learn the mapping function between the feature vector and the corresponding prototype. The DNN training is illustrated in Fig. 2, which consists of unsupervised pre-training and supervised fine-tuning.

The pre-training procedure treats each consecutive pair of layers as a restricted Boltzmann machine (RBM) [14] whose joint probability is defined as:

$$p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z}\exp\{-E(\mathbf{v}, \mathbf{h})\} \tag{10}$$

where $\mathbf{v}$ and $\mathbf{h}$ denote the observable variables and latent (hidden) variables, respectively. E is an energy function and Z is the partition function to ensure $p(\mathbf{v}, \mathbf{h})$ is a valid probability distribution. If both $\mathbf{v}$ and $\mathbf{h}$ are binary states, i.e., the Bernoulli-Bernoulli RBM, the energy function is given by

$$E(\mathbf{v}, \mathbf{h}) = -(\mathbf{b}_v^\top \mathbf{v} + \mathbf{b}_h^\top \mathbf{h} + \mathbf{v}^\top \mathbf{W}_{vh} \mathbf{h}) \tag{11}$$

where $\mathbf{b}_v$, $\mathbf{b}_h$ are bias vectors of $\mathbf{v}$ and $\mathbf{h}$ respectively, and $\mathbf{W}_{vh}$ is the weight matrix between them. If $\mathbf{v}$ is real-valued data and $\mathbf{h}$ is binary, i.e., the Gaussian-Bernoulli RBM, the energy function is:

$$E(\mathbf{v}, \mathbf{h}) = \frac{1}{2}(\mathbf{v} - \mathbf{b}_v)^\top(\mathbf{v} - \mathbf{b}_v) - \mathbf{b}_h^\top \mathbf{h} - \mathbf{v}^\top \mathbf{W}_{vh} \mathbf{h} \tag{12}$$

where we assume that the visible units follow the Gaussian noise model with an identity covariance matrix if the input data are pre-processed by mean and variance normalization.

The RBM parameters can be efficiently trained in an unsupervised fashion by maximizing the likelihood over training samples of visible units with the approximate contrastive divergence algorithm [14]. As for our DNN, a Gaussian-Bernoulli RBM is used for the first layer while a pile of Bernoulli-Bernoulli RBMs can be stacked behind the Gaussian-Bernoulli RBM. Then the parameters of RBMs can be trained layer-by-layer. Hinton *et al.* indicate that this greedy layer-wise unsupervised learning procedure always helps over the traditional random initialization.

After pre-training for initializing the weights of the first several layers, a supervised fine-tuning of the parameters in the whole neural network with the final output layer is performed. Then Eq. (9) can be specified as:

$$E = \frac{1}{R}\sum_{r=1}^{R} \|\mathcal{F}(\mathbf{x}_r; \mathbf{W}, \mathbf{b}) - \mathbf{x}_r^{\text{ref}}\|_2^2 \tag{13}$$

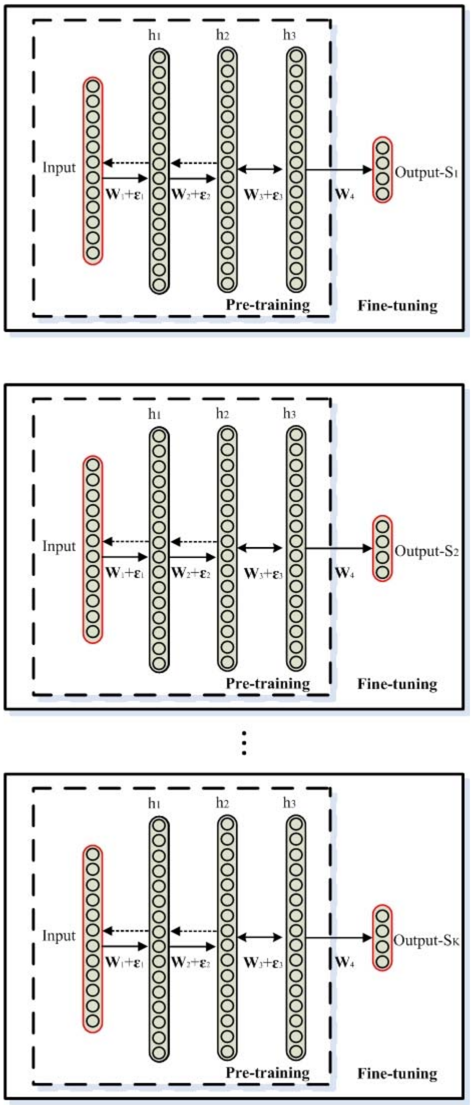

Figure 3. HDNN.

where $\mathbf{W}$ and $\mathbf{b}$ denote all the weight and bias parameters. The objective function is optimized using back-propagation procedure with conjugate gradient method in mini-batch mode.

3.2. Hierarchical Deep Neural Network

Although the powerful modeling capability of DNN has been demonstrated in many areas, in this work our experiments show that the use of DNN as a highly nonlinear regression function between two high-dimensional vectors on Chinese handwriting recognition task is not successful which still leads to the performance degradation even using deep architectures. So we propose a more powerful model called hierarchical deep neural network illustrated in Fig. 3, which achieves both deep and wide architectures of neural network. The main principle of HDNN is to divide the vector of the output layer in Fig. 2 into K sub-vectors, each associated with a DNN using the same input layer and hidden layers, which can make the learning of the regression function $\mathcal{F}$ easier. Then the formulation of HDNN is extended from Eq. (9):

$$\begin{aligned} E &= \frac{1}{R}\sum_{r=1}^{R}\|\mathbf{x}_r^{\text{ivn}} - \mathbf{x}_r^{\text{ref}}\|_2^2 \\ &= \frac{1}{R}\sum_{r=1}^{R}\sum_{k=1}^{K}\|\mathbf{x}_{r,k}^{\text{ivn}} - \mathbf{x}_{r,k}^{\text{ref}}\|_2^2 \\ &= \sum_{k=1}^{K} E_k \end{aligned} \tag{14}$$

and

$$E_k = \frac{1}{R}\sum_{r=1}^{R}\|\mathbf{x}_{r,k}^{\text{ivn}} - \mathbf{x}_{r,k}^{\text{ref}}\|_2^2 \tag{15}$$

where $\mathbf{x}_{r,k}^{\text{ivn}}$ and $\mathbf{x}_{r,k}^{\text{ref}}$ are the k^{th} subvectors of $\mathbf{x}_r^{\text{ivn}}$ and $\mathbf{x}_r^{\text{ref}}$, respectively. Obviously, the optimization of Eq. (14) can be divided into K subproblems as in Eq. (15), each associated with a DNN in Fig. 3. The initialization of the parameters in the first several hidden layers in each DNN can share the same pre-training procedure described in Section 3.1.

3.3. Implementation Issues of HDNN

In this work, by considering the peculiarity of LDA transformed feature vector, we design a specific implementation of HDNN. First, K is set to D which implies that each dimension of output is associated with a DNN. Then for the final transformed feature vector $\mathbf{x}_r^{\text{ivn}}$, only the first D_{sub} ($D_{\text{sub}} < D$) outputs of HDNN is used and the remaining $D - D_{\text{sub}}$ dimensions are set as the same values of the input feature vector. In other words, we only need to train D_{sub} sub DNNs for HDNN. This is inspired by the fact that the most useful information of LDA transformed feature vector lies in the first several dimensions while the remaining dimensions are noisy.

4. Experiments and Results

The experiments are conducted on the task of recognizing isolated online handwritten Chinese characters with a vocabulary of 3,926 character classes via a public database released by the Institute of Automation of Chinese Academy of Sciences (CASIA) [19]. There are

Table 1. Performance (character error rate in %) comparison of different systems using prototype-based classifiers with LBG clustering on the testing set.

Methods	Baseline	DNN-1L	DNN-2L	DNN-3L	HDNN-1L	HDNN-2L
CER(%)	16.13	29.26	23.30	25.63	13.44	12.37

939,561 samples in the training set and 234,798 samples in the testing set. For feature extraction, a 512-dimensional raw feature vector is extracted as described in [2], which is followed by LDA transformation to obtain a 128-dimensional feature vector. For Quickprop-based SSM-MCE training, the setting of the control parameters can refer to [20]. The number of prototype for each character class of the classifier used for DNN/HDNN training is set to 1. D_{sub} is set as 48. The tuning parameters of DNN are set according to [12]. The number of units in each hidden layer of DNN is 1024. To handle the large-scale training data, the computations of LBG clustering, SSM-MCE training with Quickprop algorithm are parallelized on the CPU cluster while DNN/HDNN training is implemented and optimized on GPUs.

Table 1 shows a performance (character error rate in %) comparison of different systems using prototype-based classifiers with LBG clustering on the testing set. “Baseline” denotes the system without using IVN. “DNN-1L” to “DNN-3L” represent the systems using DNN-based IVN with 1 hidden layer to 3 hidden layers, respectively. “HDNN-1L” and “HDNN-2L” refer to the systems using HDNN-based IVN with 1 hidden layer and 2 hidden layers for each sub DNN, respectively. Note that the prototype-based classifiers for both IVN and classification in Fig. 1 are generated using LBG clustering with 1 prototype. First, DNN-based IVN systems yield much worse performance over the baseline system, which indicates that DNN totally fails in learning the mapping function between the LDA transformed feature vector and the corresponding prototype even using deep architectures. Second, HDNN-based IVN systems achieve significant error reductions over the baseline system. In terms of recognition error rate, HDNN shows much more powerful learning capability than DNN. Furthermore, HDNN-2L system gives the best recognition performance with deeper architecture than HDNN-1L system.

Table 2. Performance (character error rate in %) comparison of systems using prototype-based classifiers with different features and different training criteria on the testing set.

	#prototype	LBG	SSM-MCE
Baseline	1	16.13	12.26
	4	13.68	11.64
HDNN (LBG)	1	12.37	11.64
	4	11.84	11.32
HDNN (SSM-MCE)	1	11.38	10.82
	4	10.96	10.61

Table 2 gives a performance (character error rate in %) comparison of systems using prototype-based classifiers with different features and different training criteria on the testing set. “HDNN(LBG)” and “HDNN(SSM-MCE)” denote HDNN-based IVN systems using two prototype-based classifiers with LBG clustering and SSM-MCE training for HDNN training, respectively. 2 hidden layers are used in each sub DNN of HDNN. Several observations can be made. First, all the HDNN-based IVN systems yield consistently significant performance gain over baseline system in the corresponding setting of different number of prototypes and training criteria, especially for prototype-based classifier trained using LBG clustering. Second, without SSM-MCE based discriminative training, namely, the system using prototype-based classifier trained by LBG clustering in the case of “HDNN(LBG)” can still generate comparable recognition accuracy with SSM-MCE trained baseline system, which implies that the feature space after transformation via HDNN-based IVN brings more discriminative information. Third, the HDNN-based IVN system using SSM-MCE generated prototypes for HDNN training can always outperform the system using LBG generated prototypes. Finally, the best HDNN-based IVN system can achieve about absolute 1% error reduction over the best baseline system under the setting of 4 prototypes and SSM-MCE training (from 11.64% to 10.61%).

5. Conclusion

In this work, we investigate to use a HDNN-based IVN approach via prototype-based classifier for online handwritten Chinese character recognition. It is verified that HDNN has a more powerful modeling capability than DNN, which brings significant error reductions.

As for future work, the discriminative training criterion will be further explored on top of the HDNN training using minimum mean squared error criterion.

6. Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grants No. 61305002, the Programs for Science and Technology Development of Anhui Province, China under Grants No. 13Z02008-4 and No. 13Z02008-5, and Youth Innovation Fund of USTC (University of Science and Technology of China) under Grants No. 2013.

References

[1] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul. A compact model for speaker-adaptive training. In *ICSLP*, pages 1137–1140, 1996.

[2] Z.-L. Bai and Q. Huo. A study on the use of 8-directional features for online handwritten Chinese character recognition. In *ICDAR*, pages 262–266, 2005.

[3] J. Chen, B. Zhang, H. Cao, R. Prasad, and P. Natarajan. Applying discriminatively optimized feature transform for HMM-based off-line handwriting recognition. In *ICFHR*, pages 219–224, 2012.

[4] P. Dreuw, D. Rybach, C. Gollan, and H. Ney. Writer adaptive training and writing variant model refinement for offline Arabic handwriting recognition. In *ICDAR*, pages 21–25, 2009.

[5] J. Du and Q. Huo. A discriminative linear regression approach to OCR adaptation. In *ICPR*, pages 629–632, 2012.

[6] J. Du and Q. Huo. Designing compact classifiers for rotation-free recognition of large vocabulary online handwritten Chinese characters. In *ICASSP*, pages 1721–1724, 2012.

[7] 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*, 46(8):2313–2322, 2013.

[8] J. Du and Q. Huo. An irrelevant variability normalization based discriminative training approach for online handwritten Chinese character recognition. In *ICDAR*, pages 69–73, 2013.

[9] Z.-D. Feng and Q. Huo. Confidence guided progressive search and fast match techniques for high performance Chinese/English OCR. In *ICPR*, pages III–89–92, 2002.

[10] M. J. F. Gales. Maximum likelihood linear transformations for HMM-based speech recognition. *Computer Speech and Language*, 12(2):75–98, 1998.

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

[12] G. Hinton. A practical guide to training restricted Boltzmann machines. Technical report, University of Toronto, 2010.

[13] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury. Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012.

[14] G. Hinton, S. Osindero, and Y. Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527–1554, 2006.

[15] G. Hinton and R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[16] Q. Huo and B. Ma. Irrelevant variability normalization in learning HMM state tying from data based on phonetic decision-tree. In *ICASSP*, pages 577–580, 1999.

[17] Y. Linde, A. Buzo, and R. Gray. An algorithm for vector quantizer design. *IEEE Trans. on Communications*, 28(1):84–95, 1980.

[18] C.-L. Liu, S. Jaeger, and M. Nakagawa. Online recognition of Chinese characters: the state-of-the-art. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(2):198–213, 2004.

[19] C.-L. Liu, F. Yin, D.-H. Wang, and Q.-F. Wang. Online and offline handwritten Chinese character recognition: benchmarking on new databases. *Pattern Recognition*, 46(1):155–162, 2013.

[20] Y.-Q. Wang and Q. Huo. A study of designing compact recognizers of handwritten Chinese characters using multiple-prototype based classifiers. In *ICPR*, pages 1872–1875, 2010.

[21] 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*, 14(3):255–262, 2011.

[22] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee. An experimental study on speech enhancement based on deep neural networks. *IEEE Signal Processing Letters*, 21(1):65–68, 2014.

[23] B. Zhang, S. Matasoukas, and R. Schwartz. Discriminatively trained region dependent feature transforms for speech recognition. In *ICASSP*, pages 1520–1523, 2006.

[24] X.-Y. Zhang, K. Huang, and C.-L. Liu. Pattern field classification with style normalized transformation. In *IJCAI*, pages 1621–1626, 2011.

[25] X.-Y. Zhang and C.-L. Liu. Style transfer matrix learning for writer adaptation. In *CVPR*, pages 393–400, 2011.

[26] Y. Zhang, J. Xu, Z.-J. Yan, and Q. Huo. A study of an irrelevant variability normalization based discriminative training approach for LVCSR. In *ICASSP*, pages 5308–5311, 2011.