2011 International Conference on Document Analysis and Recognition

Perceptron Learning of Modified Quadratic Discriminant Function

Tong-Hua Su, Cheng-Lin Liu, Xu-Yao Zhang
National Laboratory of Pattern Recognition
Institute of Automation, Chinese Academy of Sciences, Beijing 100190, P.R. China
Email: {thsu,liucl,xyz}@nlpr.ia.ac.cn

***Abstract*—Modified quadratic discriminant function (MQDF) is the state-of-the-art classifier in handwritten character recognition. Discriminative learning of MQDF can further improve its performance. Recent advances justify the efficacy of minimum classification error criteria in learning MQDF (MCE-MQDF). We provide an alternative choice to MCE-MQDF based on the Perceptron learning (PL-MQDF). For better generalization performance, we propose a new dynamic margin regularization. To relieve the heavy burden in training process, active set technique is employed, which can save most of the computation with negligible loss in accuracy. In experiments on handwritten digit datasets and a large-scale Chinese handwritten character database, the proposed PL-MQDF was demonstrated superior in both error reduction and training speedup.**

***Keywords*-Chinese handwritten character recognition, MQDF, Perceptron, dynamic margin, active set**

I. Introduction

Modified quadratic discriminant function (MQDF), originally proposed by Kimura [1], is a compact Gaussian classifier with the-state-of-the-art performance in handwritten character recognition. MQDF re-parameterizes the covariance matrix into eigenvalues and eigenvectors and truncates the small eigenvalues to denoise the unstable estimation. MQDF often generalizes better than regular quadratic discriminant function (QDF) and it runs faster and requires lower storage by product. Traditionally, the parameters of MQDF are derived from maximum likelihood estimator (MLE) which is a generative method, thus the learning process is not directly related to its classification performance.

Previously, minimum classification error (MCE) criterion is tried on MQDF [2], [3], [4]. The MCE criterion is well known in speech recognition [5] and it is used to adjust parameters of MQDF (MCE-MQDF). Liu et al [2] conduct thorough investigation on MCE-MQDF and the experiments on digit recognition justify its superior. Due to the intensive computation burden, MCE-MQDF is applied on Chinese handwritten character recognition where just the mean vectors and eigenvalues of MQDF are adjusted [3]. Recently, MCE-MQDF with a revised loss function is applied on Chinese handwritten character recognition with the help of parallelization programming and they only update the mean vectors [4]. Despite rapid advances in computer speed, heavy training burden is still one of the biggest problems for such discriminative learning methods.

To overcome the shortcomings of MCE-MQDF, we propose a new method for discriminative learning of MQDF in the context of large-category learning task. Our model is based on one of the oldest Perceptron algorithm [6], that is, we pursue for Perceptron learning of modified quadratic discriminant function (PL-MQDF). However, the direct use of Perceptron suffers from overfitting and slow training. Two endeavors are done to make Perceptron generalize better and scale better. First of all, we regularize the objective function through a dynamic margin constraint to achieve a better generalization performance. On the other hand, active set technique is equipped with Perceptron; most computation costs are saved meanwhile without much loss in accuracy.

It is worthy to further highlight our model from machine learning perspective. PL-MQDF is a quadratic Perceptron, which is different from the traditional nonlinear extension of linear models. Mostly, linear models are extended into their nonlinear counterpart via kernel trick or explicit mapping. When the dataset is large, kernel methods will produce many support vectors, which is time-consuming in evaluation. Explicit quadratic mapping requires a large memory to save the mapped features, and it can only learn a regular QDF. The MQDF cannot be equipped with these two tricks for its complex nonlinear structure. Therefore, PL-MQDF is irreplaceable by those two nonlinear extensions. What's more, PL-MQDF scales well to large-scale applications.

The paper is organized as follows. We describe background materials relating to MQDF and Perceptron algorithm in Sect. II. We then present the PL-MQDF in Sect. III. Margin regularization and active set technique are detailed in turn. The next section clarifies the relationship with the learning vector quantization (LVQ) family. Sect. V provides empirical studies on both handwritten digit databases and a Chinese handwritten character database. Concluding remarks are given in Sect. VI.

II. Background

A. MQDF

In Bayesian classification, we get the quadratic discriminant function (QDF) under the assumptions of multivariate Gaussian density and equal a priori probabilities:

$$d_Q^2(\mathbf{x},\omega_i) = (\mathbf{x}-\mu_i)^T\Sigma_i^{-1}(\mathbf{x}-\mu_i) + \ln|\Sigma_i| \quad , \quad (1)$$

1520-5363/11 $26.00 © 2011 IEEE
DOI 10.1109/ICDAR.2011.204

where μ_i and Σ_i are the mean vector and covariance matrix of class ω_i, respectively. The modified QDF (MQDF) further replaces the minor eigenvalues with a larger constant [1]:

$$d^2(\mathbf{x},\omega_i) = \sum_{j=1}^{k}\frac{1}{\lambda_{ij}}\left[\phi_{ij}^T(\mathbf{x}-\mu_i)\right]^2 + \sum_{j=1}^{k}\log\lambda_{ij} + \frac{1}{\delta_i}r_i(\mathbf{x}) + (d-k)\log\delta_i \quad , \quad (2)$$

where λ_{ij} is the j-th largest eigenvalue of Σ_i, ϕ_{ij} is the corresponding eigenvector, δ_i is the truncated minor eigenvalues, k denotes the number of principal axes and $r_i(\mathbf{x})$ is the residual of subspace projection:

$$r_i(\mathbf{x}) = \|\mathbf{x}-\mu_i\|^2 - \sum_{j=1}^{k}\left[\phi_{ij}^T(\mathbf{x}-\mu_i)\right]^2 \quad . \quad (3)$$

MQDF falls into generative model and its parameter set $\Theta_i = \{\lambda_{ij}, \delta_i, \mu_i, \phi_{ij}\}$ is commonly derived from the estimation of maximum likelihood.

B. Perceptron Learning

Hybrid generative and discriminative models will give higher classification accuracy, as well as the resistance to outliers [2]. Perceptron is used for discriminative learning of MQDF. Firstly, the misclassification measure on a pattern from class ω_c is defined, following Juang et al. [5]:

$$h(\mathbf{x},\Theta_c,\Theta_r) = d^2(\mathbf{x},\omega_c) - d^2(\mathbf{x},\omega_r) \quad , \quad (4)$$

where $d^2(\mathbf{x},\omega_r)$ is the score of the closest rival class: $d^2(\mathbf{x},\omega_r) = \min_{i\in\mathcal{M}} d^2(\mathbf{x},\omega_i)$ with $\mathcal{M}$ is the candidate set. Using Perceptron learning, the goal is to minimize:

$$\mathcal{L}_p = \sum_{\mathbf{x}_n\in\mathcal{E}} d^2(\mathbf{x}_n,\omega_c) - d^2(\mathbf{x}_n,\omega_r) \quad , \quad (5)$$

where $\mathcal{E} = \{\mathbf{x}|h(\mathbf{x},\Theta_c,\Theta_r) > 0\}$.

On instance $\mathbf{x}_n$, stochastic gradient descent (SGD) updates as:

$$\Theta_i \leftarrow \Theta_i - \eta\frac{\partial\{d^2(\mathbf{x}_n,\omega_c) - d^2(\mathbf{x}_n,\omega_r)\}}{\partial\Theta_i} \quad , \quad (6)$$

where η is the learning step.

III. Learning Model

The direct use of Perceptron suffers from overfitting and slow training. This section provides the essential building blocks for a effective and efficient PL-MQDF.

A. Dynamic Margin Regularization

Margin constraint is imposed to ensure a good generalization. If the difference between the squared distance from x to ω_r and that from x to ω_c does not exceed a margin, a margin error is triggered. That is, not only the misclassified samples, but also the correctly classified ones that near the decision boundary are used to update the parameters.

Analogous to [7], the objective function with a constant margin strength can be formulated as:

$$\mathcal{L}_p = \sum_{\mathbf{x}\in\mathcal{E}} d^2(\mathbf{x},\omega_c) - d^2(\mathbf{x},\omega_r) + \rho \quad , \quad (7)$$

where ρ is the margin strength and $\mathcal{E} = \{\mathbf{x}|d^2(\mathbf{x},\omega_c) - d^2(\mathbf{x},\omega_r) > -\rho\}$.

Considering the large divergence of different classes, fixed margin may loss its precision. On every instance $\mathbf{x}$, we seek a local margin proportional to the squared distance from $\mathbf{x}$ to ω_c. Then we would like to let all instances satisfy: $d^2(\mathbf{x},\omega_r) - d^2(\mathbf{x},\omega_c) > \rho d^2(\mathbf{x},\omega_c)$. Intuitively, $\mathbf{x}$ should be not only correctly classified but also kept away from the decision boundary. We renew the misclassification measure:

$$h'(\mathbf{x},\Theta_c,\Theta_r) = (1+\rho)d^2(\mathbf{x},\omega_c) - d^2(\mathbf{x},\omega_r) \quad (8)$$

$$= \rho d^2(\mathbf{x},\omega_c) + d^2(\mathbf{x},\omega_c) - d^2(\mathbf{x},\omega_r) \quad (9)$$

We first consider Eq. 8. The first term penalizes large distances from $\mathbf{x}$ to its true class. The gradient of this term generates a pulling force that attracts the true class. The second term penalizes small distances from $\mathbf{x}$ to its most confusable class. The gradient of this term generates a pushing force that repels the rival class. Moreover, the strength of pulling is larger than that of pushing and is determined by the intensity of ρ. We then consider Eq. 9. The role of the first term is the same as the term in Eq. 8. The remaining terms penalize the classification error. Their gradient generates a pushing force to make the difference of those squared distances larger. As a result, the empirical errors may be greatly decreased under these two terms.

Plugging in the dynamic margin, the objection function becomes:

$$\mathcal{L}'_p = \sum_{\mathbf{x}\in\mathcal{E}}(1+\rho)d^2(\mathbf{x},\omega_c) - d^2(\mathbf{x},\omega_r) \quad , \quad (10)$$

where $\mathcal{E} = \{\mathbf{x}|d^2(\mathbf{x},\omega_c) - d^2(\mathbf{x},\omega_r) > -\rho d^2(\mathbf{x},\omega_c)\}$. The elements in $\mathcal{E}$ incur a classification error or violate the margin error. Only in Eq. 10 can the margin constraint be viewed as a regularization to MLE. If ρ is 0, the model degenerates as a Perceptron without margin. On the contrary, we attain a maximum likelihood estimation when ρ approaching to ∞.

At the t-th iteration, Θ_c and Θ_r are updated on the erroneous instance $\mathbf{x}_n$ as:

$$\begin{cases}\Theta_c(t+1) = \Theta_c(t) - (1+\rho)\eta_t \bigtriangledown d^2(\mathbf{x}_n,\omega_c)|_{\Theta_c(t)} \\ \Theta_r(t+1) = \Theta_r(t) + \eta_t \bigtriangledown d^2(\mathbf{x}_n,\omega_r)|_{\Theta_r(t)}\end{cases} . \quad (11)$$

We should highlight that Eq. 7 does not affect the learning rate in Eq. 6, while Eq. 10 changes the learning rate of Θ_c from η_t to $(1+\rho)\eta_t$. If we specify the initial learning rate and the last one before halting as $1/t_0$ and $1/(m.t_0)$ respectively, then the learning rate at t-th step is set as:

$$\eta_t = \frac{T.S}{t_0(T.S + (m-1).(t-1))} \quad , \quad (12)$$

Figure 1. The switching order of active sets.

where T is the total number of training rounds, S is the sample size.

There are two distinctions than MCE learning. Firstly, the loss function is different. Perceptron employs a hinge-style loss; MCE uses a sigmoid function to approximate the 0-1 loss. Secondly, they vary in the update rule. Perceptron just considers the information from misclassified instances while MCE absorbs information from all instances. Although their distinctions, many implementation tricks of MCE can be seamlessly adopted to Perceptron such as parameter transformation, partial derivation of squared distance (Cf. [2] for more details).

B. Active Set Technique

We employ active set technique to speed up the training process of Perceptron algorithm. If a instance invokes a classification error or margin error, we say it "active". During the cycling once through the available training instances, all active instances form a active set for the subsequent rounds. Such procedure can be nested and active sets of different levels can be generated.

For demonstration purpose, we present two nested active sets similar to [8]. The first active set named AS_1 is composed of the instances from the full training dataset violating margin constraints. The second one named AS_2 is built from the active instances in AS_1. AS_2 is presented repetitively to the algorithm for N_2 passes. Then AS_1 comes back which is cycled once again to the algorithm. Each time AS_1 is under consideration, AS_2 is restarted. After N_1 times invoking of AS_1, we will resume to the full training dataset and the procedure starts all over again. The switching of above three sets is depicted in Fig. 1. The full training dataset is denoted as AS_0.

We give an concrete example to show that the active set technique will not excessively drop the samples that should be used for learning in our setting. We select the Chinese handwritten character database (Cf. Sect. V-B) as full training dataset and set $\{N_1 = 10, N_2 = 0\}$. During the first cycling through AS_0, we generate a AS_1. On each of the following 10 rounds, we first record all samples with margin error from AS_0 and the percentage of samples that can be found in AS_1 is called hit percentage. When the allowed rounds are exhausted, we generate a new AS_1 and restart the collection of hit percentage. After cycling 20 times, we stop it. The hit percentages of the process are shown in Fig. 2. We can see a high hit percentage and a large reduction in the size of active set.

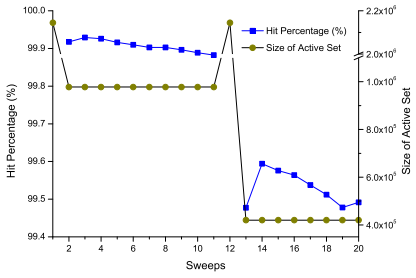


Figure 2. The hit percentage versus the size of active set.

IV. Interpretation

Our model is applicable to both MQDF and QDF. We show that the learning model of this paper is a general framework covering LVQ family. If we impose structural constraints on Σ_i, we obtain LVQ and its variants such as LVQ with class-independent weighting and LVQ with class-dependent weighting (Cf. [9]).

Case 1: Assuming $\forall i, \Sigma_i = \mathbf{I}$ ($\mathbf{I}$ is the identity matrix), Eq. 4 becomes:

$$\begin{aligned} h(\mathbf{x}, \Theta_c, \Theta_r) =& ||\mathbf{x} - \mu_c||_2^2 - ||\mathbf{x} - \mu_r||_2^2 \\ =& -2(||\mathbf{x} - \mu_c||_2 + ||\mathbf{x} - \mu_r||_2) \\ & \bullet \underbrace{\frac{1}{2}(||\mathbf{x} - \mu_r||_2 - ||\mathbf{x} - \mu_c||_2)}_{\text{hypothesis margin}} \end{aligned} . \quad (13)$$

Using SGD w.r.t. μ_i, we get the LVQ classifier. The regularization of dynamic margin in Eq. 10 is similar to [9] and helps to enlarge the hypothesis margin. As shown in [10], maximizing hypothesis margin provides a upper bound of generalization error for LVQ.

Case 2: Assuming $\forall i, \Sigma_i = \text{diag}\{w^1, ..., w^d\}$, Eq. 4 becomes:

$$h(\mathbf{x}, \Theta_c, \Theta_r) = \sum_{p=1}^{d} w^p (x^p - \mu_c^p)^2 - \sum_{p=1}^{d} w^p (x^p - \mu_r^p)^2 \quad (14)$$

and we get the LVQ with class-independent weighting. Even we can assume $\Sigma_i = \Sigma$ and fix it. We will run LVQ in a linear transformed space and all theoretical reasoning are still held.

Case 3: Assuming $\Sigma_i = \text{diag}\{w_i^1, ..., w_i^d\}$ and omitting logarithmic terms, Eq. 4 becomes:

$$h(\mathbf{x}, \Theta_c, \Theta_r) = \sum_{p=1}^{d} w_c^p (x^p - \mu_c^p)^2 - \sum_{p=1}^{d} w_r^p (x^p - \mu_r^p)^2 \quad (15)$$

and we get the LVQ with class-dependent weighting. If we further relax the structural constraints to just require $\Sigma_i \succeq 0$, we arrive at the learning model in this paper.

V. EXPERIMENTS

Our investigation includes the error reduction and training time of PL-MQDF. Evaluation is conducted on both small-category learning task and large-category learning task. All the experiments are run on a personal computer with a 3.0 GHZ CPU and a 2.0 GB physical memory. We first report results on MNIST and USPS; we then present results on Chinese handwritten character database.

A. On Digit Databases

As for MNIST database, the original 20×20 fine gray-scale images are used to extract features [11]. Three kinds of features are evaluated separately: *img*, *pca80*, *e-grg* (Cf. [11] for more details). *img* is derived from arranging the image pixel into a 400D feature vector. *pca80* is processed by principal component analysis on *img* (from 400D to 80D). *e-grg* is 8-direction gradient features (200D). Likely, *img* and *e-grg* are extracted and evaluated on USPS database.

The error rates and training time of Perceptron learning (PL-MQDF) are given in Table. I and II with $k = 30$. We try active set technique in two different ways. For one trial (PL-2AS), we construct two active sets exactly following Sect. III-B and we set $\{N_1 = 1, N_2 = 3\}$. For the other (PL-1AS), we just generate one active set by simple using $\{N_1 = 10, N_2 = 0\}$. We also take efforts to train the MQDF using MCE criteria as in [2]. A speedup trick is employed for MCE-MQDF. We first compute the distance differences as in Eq. 4 for all training samples using the initial MQDF parameters and their average are taken as a threshold. During MCE training, if the squared distance from $\mathbf{x}$ to ω_r is bigger than that from $\mathbf{x}$ to ω_c by above threshold, SGD step is bypassed. For a close comparison, we illustrate the error rates on each class in Fig. 3. PL-MQDF outperforms MCE-MQDF on both MNIST and USPS and there is no severe loss in accuracy by using active set technique. In terms of training time, both PL-1AS and PL-2AS bring great reduction. Consistently, PL-1AS works well with preferable performance.

Table I
ERROR RATES (%) AND TRAINING TIME (S) ON FINE IMAGE FEATURES OF MNIST DATABASE.

classifier	*img* (400D)		*pca80* (80D)		*e-grg* (200D)	
	Err	CPU	Err	CPU	Err	CPU
MQDF	4.17	–	4.07	–	0.94	–
PL	1.55	238.98	1.52	84.94	0.52	129.84
PL-2AS	1.52	136.92	1.58	25.66	0.54	44.09
PL-1AS	1.51	122.55	1.49	22.70	0.53	30.61
MCE	1.92	196.28	1.80	79.75	0.53	119.91

1) The Effect of Sample Size: We select MNIST as a subject. We initially just draw 2000 samples from training set, then increase the training volumes and evaluate the performance on test set. The results are given in Fig. 4. It shows that models using discriminative learning require more training samples for a robust learning.

Table II
ERROR RATES (%) AND TRAINING TIME (S) ON USPS DATABASE.

classifier	*img* (256D)		*e-grg* (128D)	
	Err	CPU	Err	CPU
MQDF	5.03	–	2.84	–
PL	3.94	21.63	2.19	14.80
PL-2AS	3.84	13.39	2.19	5.00
PL-1AS	3.94	12.00	2.24	3.55
MCE	4.09	15.69	2.49	22.39

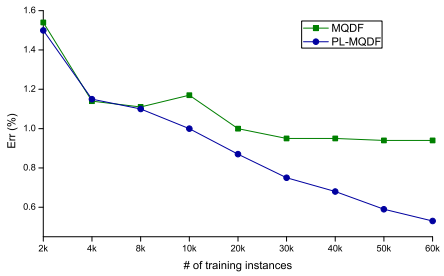


Figure 4. Effect of training volumes for PL-MQDF (on *e-grg* of MNIST).

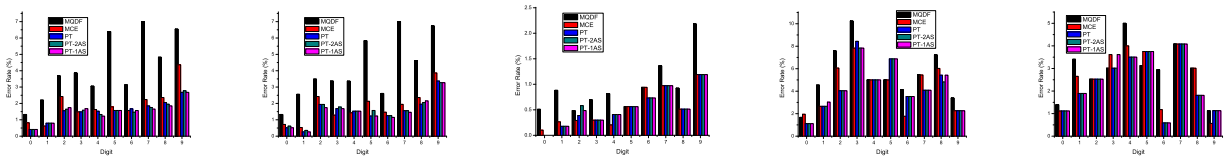
B. On Chinese Character Database

Due to the heavy burden in computation consumption, discriminative learning algorithm is seldom attempted on large-category tasks. We will show that active set technique will speed up PL-MQDF greatly and render its usability on regular personal computers.

CASIA-HWDB1.0 and CASIA-HWDB1.1 are used to verify the efficacy of PL-MQDF. There are 3,755 classes (GB1 character set) and each class has about 570 training instances. We have 2,144,749 training samples and 533,675 test samples. As recommended in [12], pseudo 2D LDI normalization is first executed and then NCGF features are extracted on gray-scale samples. Also LDA is employed to reduce the feature dimensionality from 512 to 160. In both experiments, we set $k = 50$.

We employ a two-stage classification process. In terms of testing, an LVQ classifier first ranks all classes and provides 100 candidate classes of leading rankings. Then quadratic models are incurred to re-rank the candidates and output the label of the first place. As for training process, training instances are cycled through repeatedly, and MQDF models are adjusted as needed. On each instance, LVQ classifier provides a candidate class set $\mathcal{M}$ ($|\mathcal{M}| = 10$ or $|\mathcal{M}| = 100$). If the true label of the instance is not included in $\mathcal{M}$, the SGD is bypassed. Otherwise, we only invoke SGD to update parameters each time violation of the margin constraint is found. We set $\rho = 0.05$ when dynamic margin is used and we cycle 20 epochs through the training data (or active set).

The results are provided in Table III. The MCE-MQDF is speeded up as mentioned in Sect. V-A. We just summarize three general trends as follows. Firstly, a small candidate



(a) *img* of MNIST (b) *pca80* of MNIST (c) *e-grg* of MNIST (d) *img* of USPS (e) *e-grg* of USPS

Figure 3. The error rates per class of different methods on MNIST and USPS.

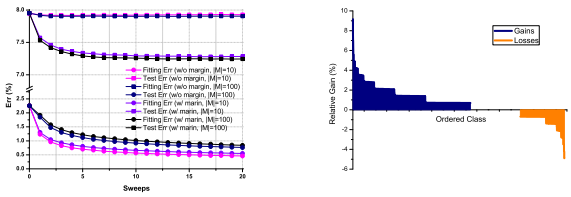
Table III
ERROR RATES (%) AND TRAINING TIME (HOUR) ON CASIA-HWDB1.0 AND CASIA-HWDB1.1 DATABASE.

classifier (ρ)	$\|\mathcal{M}\|$=10		$\|\mathcal{M}\|$=100	
	Err	CPU	Err	CPU
MQDF	7.95	–	7.95	–
PL(0)	7.93	15.89	7.91	29.75
PL(0.05)	7.28	16.26	7.24	30.05
PL-1AS(0.05)	7.29	6.74	7.25	13.01
MCE	7.30	16.27	7.26	30.33

set is enough for discriminative learning. There is at most 0.04% extra error reduction from 100 candidate classes than 10, however, the time consumption is nearly doubled.

Secondly, regularization via margin is indispensable to PL-MQDF. It is easily stuck in overfitting solution without margin regularization and 8.20% relative error reduction is yielded when $\rho = 0.05$. We plot the training process of PL-MQDF in Fig. 5(a).

Thirdly, PL-MQDF is comparative to MCE-MQDF. In particular, our model requires lower computation cost when equipped with active set technique. The active set method accelerates PL-MQDF greatly and the loss of accuracy is negligible. The ordered gains and losses w.r.t. all classes in accuracy are illustrated in Fig. 5(b). In all, the PL-MQDF achieves 8.30% error reduction than MQDF with acceptable training cost.



(a) (b)

Figure 5. Evaluation of PL-MQDF: (a) the effect of candidate size and margin regularization; (b) gains and losses against MQDF.

VI. CONCLUSION

We present a Perceptron based learning model for MQDF named PL-MQDF. The objective function of PL-MQDF cannot be solved by traditional extensions of linear Perceptron. A well-behaved solution can be attained through dynamic margin constraint during SGD procedures. Moreover, cost of the training time can be saved through active set technique meanwhile preserving its accuracy. Experiments in contexts of large-category learning task justify the efficacy and efficiency of PL-MQDF.

ACKNOWLEDGEMENTS

This work was supported by National Natural Science Foundation of China (NSFC) under grants no. 60825301 and no. 60933010, as well as China Postdoctoral Science Foundation under grant no. 20090450623.

REFERENCES

[1] F. Kimura, K. Takashina, S. Tsuruoka, Y. Miyake. Modified quadratic discriminant functions and the application to Chinese character recognition. *IEEE TPAMI*, 9: 149-153, 1987.
[2] C.-L. Liu, H. Sako, H. Fujisawa. Discriminative learning quadratic discriminant function for handwriting recognition. *IEEE TNN*, 15(2): 430-444, 2004.
[3] H. Liu, X. Ding. Handwritten Character recognition using gradient feature and quadratic classifier with multiple discrimination schemes. *ICDAR*, 2005.
[4] Y. Wang, Q. Huo. Sample-separation-margin based minimum classification error training of pattern classifiers with quadratic discriminant functions. *ICASSP*, 2010.
[5] B.-H. Juang, W. Chou, C.-H. Lee. Minimum classification error rate methods for speech recognition. *IEEE Trans. Speech and Audio Processing*, 5(3): 257-265, 1997.
[6] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65 (6):386-408, 1958.
[7] C. Cheng, F. Sha, L. Saul. A fast online algorithm for large margin training of continuous density hidden markov models. *Proceedings of Interspeech*, 2009.
[8] C. Panagiotakopoulos, P. Tsampouka. The margin Perceptron with unlearning. *ICML*, 2010.
[9] X.-B. Jin, C.-L. Liu, X.-W. Hou. Regularized margin-based conditional log-likelihood loss for prototype learning. *Pattern Recognition*, 43: 2428-2438, 2010.
[10] K. Crammer, R. Gilad-Bachrach, A. Navot. Margin analysis of the LVQ algorithm. *NIPS*, 2002.
[11] C.-L. Liu, K. Nakashima, H. Sako, H. Fujisawa. Handwritten digit recognition: benchmarking of state-of-the-art techniques. *Pattern Recognition*, 36: 2271-2285, 2003.
[12] C.-L. Liu, F. Yin, D.-H. Wang, Q.-F. Wang. Online and offline handwritten Chinese character recognition: Benchmarking on new databases. *CJKPR*, 2010.