

Combining Generative and Discriminative Learning for Face Recognition

Shaokang Chen Brian C. Lovell Ting Shan
School of Information Technology and Electrical Engineering
The University of Queensland
Brisbane, Australia
{shaokang, lovell, shanting}@itee.uq.edu.au

Abstract

Face recognition is a very complex classification problem and most existing methods are classified into two categories: generative classifiers and discriminative classifiers. Generative classifiers are optimized for description and representation which is not optimal for classification. Discriminative classifiers may achieve less asymptotic errors but are inefficient to train and may overfit to training data. In this paper, we present a hybrid learning algorithm that combines both generative learning and discriminative learning to find a trade-off between these two approaches. Experiments on Asian Face Database show a reduction in classification error rate for our hybrid learning method.

1 Introduction

Face recognition is a very challenging task and has attracted considerable attention from psychophysicists, neuroscientists and engineers for more than 50 years. Various techniques has been applied in automatic face recognition, such as Principal Component Analysis (PCA) [17], Linear Discriminant Analysis (LDA) [1, 13], Hidden Markov Models (HMMs) [16], Neural Networks [11, 5] and Support Vector Machines (SVMs) [6]. These methods are generally classified into two categories: generative classifier and discriminative classifier.

Generative and discriminative learning are two paradigms of machine learning. Generative learning is the main approach for pattern classification, artificial intelligence and perception. It focuses on generative description of samples and tend to synthesize configurations from them. Consequently, corresponding classifiers, detectors and predictors can be built based on the generative model. PCA [17], LDA [1, 13] and HMMs [16] are typical generative classifiers that produce a probability density model for face recognition. Discriminative learning attempts to compute the mapping for classification from

input to output directly without modelling the underlying distributions. It normally achieves superior performance than generative approach in many applications. Traditional Neural Networks [11, 5] and SVMs [6] are discriminative classifiers that attempt to maximize the classification boundary margin of classes (faces) for face recognition. Recent research on combining generative and discriminative learning has shown that proper combinations of two models outperforms pure generative or discriminative models [19, 2, 15, 14, 8, 18].

In 2004, we proposed a new method Adaptive Principal Component Analysis (APCA) [3, 12] for face recognition, which is robust to face image variations in illumination and expression. In this paper, we are going to introduce a hybrid learning method that combines generative learning and discriminative learning based on our proposed APCA method to further improve the recognition accuracy. In section 2, we briefly explain APCA method and knowledge of generative and discriminative learning. Then we discuss in details on the design of generative and discriminative learning algorithms and the combination of both paradigms based on APCA in section 3. Section 4 is devoted to the experimental results. Finally, we present conclusion and future work in section 5.

2 Preliminary Knowledge

2.1 Adaptive Principal Component Analysis (APCA)

Adaptive Principal Component Analysis [3, 12] is a linear pattern classification algorithm that inherit merits from both PCA and FLD by warping the face subspace according to the within-class and between-class covariance of samples. We first apply PCA on face images to extract eigenfaces. Consequently, every face image is projected into a face subspace with reduced dimensionality to form a m -dimensional feature vector $s_{j,k}$ with $k = 1, 2, \dots, K_j$ denoting the k^{th} sample of the class S_j . Then the face subspace

is warped by the following three steps:

- **Space Rotation:** The feature space is rotated according to the overall within-class covariance. The rotation matrix R is a set of eigen vectors obtained by applying singular value decomposition to the overall within-class covariance matrix. By space rotation, the representativeness of features are enhanced and features are either more discriminative or generative after rotation.
- **Whitening Transformation:** The subspace is whitened according to the eigen values λ_i ($i = 1, 2, \dots, m$) of the PCA extracted face subspace with a whitening power p . Each eigenface u_i is whitened according to the corresponding eigen-value λ_i with the power p . Consequently, the whitening matrix is:

$$Z = \text{diag}\{\lambda_1^p, \lambda_2^p, \dots, \lambda_m^p\}. \quad (1)$$

Whitening transformation is used to control the overall scatter of all samples and compensate for the overweighing of low frequency components.

- **Eigenface Filtering:** Eigen-features are weighted according to the identification-to-variation value ITV_i ($i = 1, 2, \dots, m$) with a filtering power q . The ITV is a ratio measuring the correlation with a change in person versus a change in variation for each of the eigenfaces. It is defined as the following:

$$\begin{aligned} ITV_i &= \frac{\text{BetweenClassCovariance}}{\text{WithinClassCovariance}} \\ &= \frac{\frac{1}{M} \sum_{j=1}^M \frac{1}{K} \sum_{k=1}^K |s_{i,j,k} - \varpi_{i,k}|}{\frac{1}{M} \sum_{j=1}^M \frac{1}{K} \sum_{k=1}^K |s_{i,j,k} - \mu_{i,j}|}, \\ \varpi_{i,k} &= \frac{1}{M} \sum_{j=1}^M s_{i,j,k}, \\ \mu_{i,j} &= \frac{1}{K} \sum_{k=1}^K s_{i,j,k}, \quad i = [1, \dots, m], \end{aligned} \quad (2)$$

where $s_{i,j,k}$ denotes the i_{th} element of the face vector of the k_{th} sample for class (person) S_j . The aim of eigenface filtering is to diminish the contribution of eigenfaces that are strongly affected by illumination and expression variations and enhance those features that capture the main differences between classes.

The cost function OPT is a combination of error rate and the ratio of between-class distance to within-class distance as the following:

$$OPT = \sum_{j=1}^M \sum_{k=1}^K \sum_m \left(\frac{d_{jj,k0}}{d_{jm,k0}} \right), \quad (3)$$

$$\forall m \in d_{jm,k0} < d_{jj,k0}, m \in [1 \dots m].$$

where $d_{jj,k0}$ is the within-class distance between the variant sample $s_{j,k}$ and the the standard reference image $s_{j,0}$

(typically the normally illuminated neutral image) for class S_j . Correspondingly, $d_{jm,k0}$ is the between-class distance between sample $s_{j,k}$ and the reference image $s_{m,0}$ for class S_m . The experimental results on face images in Asian Face Database [10] with both illumination and expression variations show that APCA performs much better than PCA and LDA. For more details of the APCA algorithm please refer to paper [3].

2.2 Generative and Discriminative Learning

Generative models estimate distributions of all inputs and outputs of the system and manipulate them to compute classification and regression functions. For pattern classification problem, generative classifiers produce a model of the joint possibility $p(x, y)$ of input x and output label y , then a posterior possibility can be generated according to marginal distributions, conditioning, and Bayes rules as follows:

$$\begin{aligned} p(y) &= \sum_x p(x, y), \\ p(x|y) &= \frac{p(x, y)}{p(y)}, \\ p(y|x) &= \frac{p(x|y)p(y)}{p(x)}. \end{aligned} \quad (4)$$

However, estimation of a joint possibility is a complex and difficult task. One improvement is to make an assumption on the conditional possibility $p(x|y)$ with a parametric model f with parameter θ . Then the posterior possibility is estimated as the following according to the Maximum Likelihood constraint:

$$\begin{aligned} p(y|x) &= p(y|x, \bar{\theta}), \\ \bar{\theta} &= \arg \max_{\theta} \sum_i p(y_i) f_{y_i}(x_i, \theta). \end{aligned} \quad (5)$$

By imposing possibility density models over all variables in the system, generative models provide the ability of representation, classification and prediction. However, generative models are optimized generically for description or representation which is not necessarily optimal for a specific task such as classification or regression [7, 9]. Therefore, discriminative learning is applied.

Unlike generative learning, discriminative learning models posterior possibilities directly or optimizing the mapping from input to output straightforward. Intermediate goals such as joint probabilities or conditional density functions are ignored. Therefore, constraints on optimization of parameters are different from generative learning. Normally, margin distances from the decision boundary to the nearest sample is considered. Only decision boundary or regression function approximations are adjusted to optimize

parameters. Hence, discriminative classifiers achieve better performance and less asymptotic errors compared to the generative paradigm. But discriminative models may be inefficient to train since they require simultaneous consideration of all data from all classes [4, 9, 2]. Moreover, when training data are limited, generative learning may outperform its discriminative counterpart [19].

A successful classifier should combine both generative and discriminative models. It may inherit versatility and flexibility from generative learning and take advantage of discriminative learning for its powerful classification ability. Research done in [2, 15, 14, 8, 18] proved that appropriate combination of both models is normally preferable, and may achieve higher accuracy.

3 Combinations of Generative and Discriminative Learning

3.1 Generative Learning Optimization

Our proposed APCA adapts the generative classifier — PCA by a whitening transformation to control the overall scatter of all samples and eigen filtering for weighing features. In order to maintain the generalization ability of the model and speed up the optimization procedure, we simplify the conditional density function to (6) under the limitation in the number of training samples as follows:

$$p(s|S_j) = \frac{1}{(2\pi)^{\frac{m}{2}} \prod_{i=1}^m \lambda_i^{-p} ITV_i^{-q}} \exp\left[-\frac{1}{2} \sum_{i=1}^m \frac{(s_i - \mu_{i,j})^2}{\lambda_i^{-2p} ITV_i^{-2q}}\right]. \quad (6)$$

Under this assumption, the whitening power p and filtering power q are two parameters that may affect the classification performance and need to be optimized. For a generative learning, the whitening power p and filtering power q are determined to maximize the likelihood as in (5). That is:

$$\langle \bar{p}, \bar{q} \rangle = \arg \max_{\langle p, q \rangle} \sum_{n=1}^N p(s_n | S_j, \langle p, q \rangle), \quad (7)$$

where N is the number of all training samples. With our estimation of conditional density function as in (6):

$$\langle \bar{p}, \bar{q} \rangle = \arg \max_{\langle p, q \rangle} \sum_{n=1}^N \frac{1}{(2\pi)^{\frac{m}{2}} \prod_{i=1}^m \lambda_i^{-p} ITV_i^{-q}} \exp\left[-\frac{1}{2} \sum_{i=1}^m \frac{(s_i - \mu_{i,j})^2}{\lambda_i^{-2p} ITV_i^{-2q}}\right]. \quad (8)$$

Normally, we optimize the logarithm of the above quantities which lead to the same optimization problem. Conse-

quently:

$$\begin{aligned} \langle \bar{p}, \bar{q} \rangle &= \arg \max_{\langle p, q \rangle} \sum_{n=1}^N \ln \left\{ \frac{1}{(2\pi)^{\frac{m}{2}} \prod_{i=1}^m \lambda_i^{-p} ITV_i^{-q}} \right. \\ &\quad \left. \exp\left[-\frac{1}{2} \sum_{i=1}^m \frac{(s_{i,n} - \mu_{i,j})^2}{\lambda_i^{-2p} ITV_i^{-2q}}\right] \right\} \\ &= \arg \max_{\langle p, q \rangle} \sum_{n=1}^N \left\{ \ln \left[\frac{1}{(2\pi)^{\frac{m}{2}} \prod_{i=1}^m \lambda_i^{-p} ITV_i^{-q}} \right] \right. \\ &\quad \left. + \ln \left(\exp\left[-\frac{1}{2} \sum_{i=1}^m \frac{(s_{i,n} - \mu_{i,j})^2}{\lambda_i^{-2p} ITV_i^{-2q}}\right] \right) \right\} \\ &= \arg \max_{\langle p, q \rangle} \sum_{n=1}^N \left\{ -\ln(2\pi)^{\frac{m}{2}} \right. \\ &\quad \left. - \ln \prod_{i=1}^m \lambda_i^{-p} ITV_i^{-q} \right. \\ &\quad \left. - \frac{1}{2} \sum_{i=1}^m \frac{(s_{i,n} - \mu_{i,j})^2}{\lambda_i^{-2p} ITV_i^{-2q}} \right\} \\ &= \arg \max_{\langle p, q \rangle} - \sum_{n=1}^N \left\{ \frac{m}{2} \ln(2\pi) \right. \\ &\quad \left. + \sum_{i=1}^m \ln(\lambda_i^{-p} ITV_i^{-q}) \right. \\ &\quad \left. + \frac{1}{2} \sum_{i=1}^m \frac{(s_{i,n} - \mu_{i,j})^2}{\lambda_i^{-2p} ITV_i^{-2q}} \right\} \\ &= \arg \min_{\langle p, q \rangle} \left\{ N \frac{m}{2} \ln(2\pi) + N(-p \sum_{i=1}^m \ln \lambda_i \right. \\ &\quad \left. + (-q) \sum_{i=1}^m \ln ITV_i) \right. \\ &\quad \left. + \frac{1}{2} \sum_{n=1}^N \sum_{i=1}^m \frac{(s_{i,n} - \mu_{i,j})^2}{\lambda_i^{-2p} ITV_i^{-2q}} \right\}. \end{aligned}$$

Note that the last part of the above equation is the sum of within-class distance of all training samples. This is also consistent with the nearest-neighbor rule for classification: the sample is assigned a label of the class whose distance to the sample is the minimal among all classes. It is obvious that in order to achieve the maximum likelihood, the within-class distances also tend to be minimal.

3.2 Discriminative Learning Optimization

On the contrary, for discriminative learning, optimization of the parametric model relies heavily on the definition of the loss function. Normally, p and q are determined to minimize the classification error. The optimization function *OPT* we use in section 2.1 is a combination of error rate

and ratio of between-class distance to within-class distance, which is a discriminative learning technique that achieves maximum separability of classes. We also propose another cost function which combines the error rate and overlap distance of the decision boundary.

$$\begin{aligned} \langle \bar{p}, \bar{q} \rangle &= \arg \min_{p, q} \sum_n \frac{d_j^2 - d_k^2}{\sum_{i=1}^m \frac{1}{\lambda_i^{-2p} ITV_i^{-2q}}}, \quad (10) \\ &\forall n \in d_k < d_j. \\ d_j &= \sqrt{\sum_{i=1}^m \frac{(s_{i,n} - \mu_{i,j})^2}{\lambda_i^{-2p} ITV_i^{-2q}}}. \\ d_k &= \sqrt{\sum_{i=1}^m \frac{(s_{i,n} - \mu_{i,k})^2}{\lambda_i^{-2p} ITV_i^{-2q}}}. \end{aligned}$$

The numerator is the overlap distance of decision boundary, where d_j is the within-class distance of sample s_n whose class label is j and d_k is the minimum between class distance between sample s_n and the closest class k . Note that the condition $d_k < d_j$ implies that there exist errors of classification of sample s_n according to the nearest neighbor rules, which demonstrates the overlap of decision boundary of classes. The denominator is the sum of weights of all eigen features, which is used to normalize the space so that whitening and filtering does not affect the absolute scale of the space. Hence, by minimizing the overlap distance $d_j^2 - d_k^2$, we can minimize the classification error.

3.3 Hybrid Learning Cost Function

Although discriminatively trained classifiers normally outperform generative classifiers, a hybrid model that moderately combines two strategies and inherits merits from both sides can achieve higher accuracy than pure generative or discriminative counterparts. Hence, ideally parameters are optimized to fulfil both the discriminative constraint in order to minimize classification error and the generative constraint to maximize likelihood. However, the optimal parameters of the two constraints are usually not identical and there might not exist ideal parameters that satisfy both requirements. Therefore, there has an trade-off between generative and discriminative approaches. A new cost function that combines both generative and discriminative constraints is necessary to determine the optimal parameters. We design a cost function that takes the form of the following:

$$f_{cost} = \eta f_{gen} + (1 - \eta) f_{dis}, \quad \eta \in [0, 1], \quad (11)$$

where f_{gen} is the cost function of a generative classifier and f_{dis} is the cost function of a discriminative classifier and η is a parameter that balances the importance of the two goals. If $\eta = 0$, we have a pure discriminative classifier and if $\eta = 1$,

we have a pure generative classifier. The optimization of p and q is achieved by minimizing this cost function.

Considering the optimization of the generative classifier, we set f_{gen} as:

$$\begin{aligned} f_{gen} &= N[(-p) \sum_{i=1}^m \ln \lambda_i + (-q) \sum_{i=1}^m \ln ITV_i] \quad (12) \\ &+ \frac{1}{2} \sum_{n=1}^N \sum_{i=1}^m \frac{(s_{i,n} - \mu_{i,j})^2}{\lambda_i^{-2p} ITV_i^{-2q}}. \end{aligned}$$

The discriminative cost function f_{dis} can take the form of the optimization function OPT . However, OPT represents the ratio of between-class distance to within-class distance and f_{gen} represent the within-class distance. Though combination of these two functions is possible, there is no clear meaning for this combination. Hence, we propose to use the function in (10). That is:

$$f_{dis} = \frac{\sum_{i=1}^m \frac{(s_{i,n} - \mu_{i,j})^2}{\lambda_i^{-2p} ITV_i^{-2q}} - \sum_{i=1}^m \frac{(s_{i,n} - \mu_{i,k})^2}{\lambda_i^{-2p} ITV_i^{-2q}}}{\sum_{i=1}^m \lambda_i^{2p} ITV_i^{2q}}. \quad (13)$$

The final cost function f_{cost} is the weighted sum of (12) and (13) which is the combination of within-class distance and overlap distance. The choice of η can be done empirically by searching in the whole range, but this may take a long time since the addition of one parameter may result in the calculation increasing multiplicatively. We propose to assign the weight according to the optimized value of f_{gen} and f_{dis} . Suppose we optimize the pure generative and discriminative classifiers with the training samples and achieve the minimum values G_{min} of f_{gen} and D_{min} of f_{dis} separately. Then if optimized parameters p and q of f_{gen} , f_{dis} and f_{cost} do not change significantly, when we set $\eta = \frac{1}{2}$ for f_{cost} , it is then reasonable to estimate the contribution of the two classifiers to the hybrid cost function as $\frac{1}{2}G_{min}$ and $\frac{1}{2}D_{min}$ respectively. This implies that the same change in f_{gen} and f_{dis} will affect f_{cost} differently. If we treat the two classifiers equally so that they maintain the same importance on the optimization, η can be determined by:

$$\eta = \frac{D_{min}}{G_{min} + D_{min}}. \quad (14)$$

Consequently, the cost function becomes:

$$f_{cost} = \frac{D_{min}}{G_{min} + D_{min}} f_{gen} + \frac{G_{min}}{G_{min} + D_{min}} f_{dis}. \quad (15)$$

Through this optimization to minimize f_{cost} , we not only minimize error rate by decreasing overlap distance between classes but also try to maximize the likelihood by reducing the within-class distance. Hence, classes are still highly isolated and within-class covariance is convergent which leads to better separability.

4 Experimental Results

We compare the performance of generative, discriminative and the hybrid learning on the Asian Face Database [10]. It consists of 856 facial images under 5 different standardized illuminations and 4 variant facial expressions corresponding to 107 subjects. The size of each image is 171×171 pixels with 256 grey levels per pixel. Face images are aligned so that eyes of faces are located at the same position of the image.

Figure 1 illustrates the error rate of three classifiers with different number of eigen features. From figure 1 we can

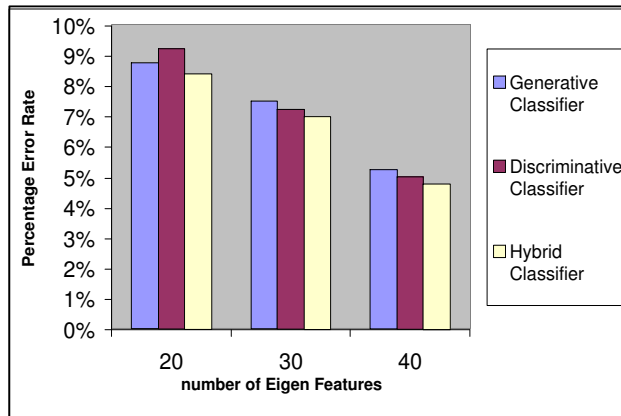


Figure 1. Error rate of generative, discriminative and hybrid classifiers.

see that error rate tends to decrease with the increase in the number of eigen features no matter what kind of learning schemes is applied, which is consistent with our experiments done in [3]. Among three schemes, hybrid learning achieves the best performance regardless of the number of features involved. Generative learning performs better than its discriminative counterpart when using only 20 eigen features. Discriminative learning outperforms the generative paradigm when more features are used to construct the face subspace. This is because in the frequency domain, the first few features are low frequency components, which are highly generative features that tend to be Gaussian distribution. Thus, generative learning may achieve better performance when limited training samples are available. With the increase in the number of features, much more discriminative features are involved in classification, so discriminative learning may reduce the error rate of classification. This effect is also corroborated in figure 2 which plots the *ITV* distribution of face images with illumination and expression variations in rotated face subspace constructed with different numbers of eigen features. *ITV* value is a ratio that describes the correlation with a change in person versus a

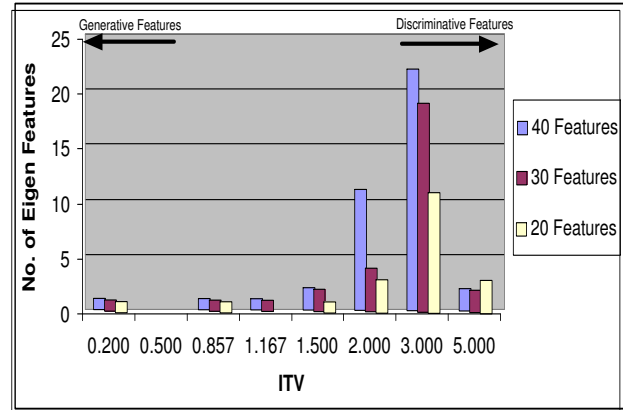


Figure 2. *ITV* distribution in rotated space with different number of features for face images with illumination and expression variations.

change in variation (illumination and expression) for each of the eigenfaces [3]. It is apparent from the figure that with the increase in the number of features, the number of generative features (whose *ITV* value is smaller than 1) does not change while the number of discriminative features ($ITV > 1.5$) increases dramatically.

Table 1 shows the minimum values obtained when trained with three different learning functions with different amount of features. The fifth row of the table is our estimated minimum values for hybrid learning with the corresponding weights and the sixth row lists corresponding η . The estimations of the minimum value are calculated according to (15) with the corresponding minimum values of generative cost function and discriminative cost function. That is:

$$\begin{aligned}
 Min_{estimation} &= \frac{D_{min}}{G_{min} + D_{min}} G_{min} \quad (16) \\
 &+ \frac{G_{min}}{G_{min} + D_{min}} D_{min} \\
 &= 2 \frac{D_{min} G_{min}}{G_{min} + D_{min}}.
 \end{aligned}$$

Comparing the actual minimum value and our estimation for hybrid learning, we can see that our estimation is very close to the actual value with an error less than 5%. Hence, the weights η and $1 - \eta$ we allocate to two classifiers are very likely to assign the same importance to generative and discriminative learning. Moreover, weight η decreases with a raise in the number of features. That means the effect of generative classifier on hybrid optimization is reducing when more features are counted. This is consistent with the fact that generative classifiers performs better than discriminative ones with less features hence it should have more

influence (higher η) on the optimization. As the amount of features increase, the discriminative classifier is more efficient, so it should be assigned a higher weight.

Table 1. Optimized values of generative, discriminative, and hybrid classifiers

| Number of Eigen Features | 20 | 30 | 40 |
|-----------------------------|--------------------|--------------------|--------------------|
| Generative Classifier | 3.83×10^4 | 5.54×10^4 | 7.20×10^4 |
| Discriminative Classifier | 6.79×10^5 | 3.12×10^5 | 1.72×10^5 |
| Hybrid Classifier | 7.44×10^4 | 9.54×10^4 | 1.08×10^5 |
| Estimation of Minimum Value | 7.26×10^4 | 9.42×10^4 | 1.02×10^5 |
| Weight η | 0.947 | 0.849 | 0.704 |

5 Conclusion and Future Work

In this paper, we develop a hybrid learning method for face recognition based on APCA [12, 3], which combines generative learning and discriminative learning. We first design a generative learning algorithm in an attempt to minimize the within-class distances and a discriminative learning algorithm in order to minimize the overlap distances. Then a hybrid learning algorithm is proposed by assigning different weights to generative and discriminative classifiers according to the corresponding minimum values of the cost function f_{gen} and f_{dis} . The experimental results show that our proposed hybrid learning method outperforms both generative and discriminative paradigms in the error rate of face classification. Our future work may involve searching in the whole range of η in interval $[0, 1]$ to determine the importance of generative and discriminative cost functions. We may also experiment other generative and discriminative cost functions and their combinations in the future.

References

[1] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711 – 720, 1997.

[2] G. Bouchard. The trade-off between generative and discriminative classifiers. In *Proceedings of Advances in Neural Information Processing Systems*, Vancouver, British Columbia, Canada, 2003.

[3] S. Chen and B. C. Lovell. Illumination and expression invariant face recognition with one sample image. In *Proceedings International Conference on Pattern Recognition*, Cambridge, UK., August 2004.

[4] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. John Wiley and Sons, 2001.

[5] M. J. Er, S. Wu, and J. Lu. Face recognition using radial basis function neural networks. In *Proceedings of the Conference on Decision and Control*, Phoenix, USA, 1999.

[6] G. Guo, S. Z. Li, , and K. Chan. Face recognition by support vector machines. In *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition 2000*, pages 196–201, Grenoble, France, 2000.

[7] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer-Verlag, 2001.

[8] T. S. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In *Proceedings of the conference on Advances in neural information processing systems II*, Vancouver, British Columbia, Canada, 1998.

[9] T. Jebara. Discriminative, generative and imitative learning. *PhD thesis, Massachusetts Institute of Technology*, 2002.

[10] I. M. Lab. Asian face image database pf01. <http://nova.postech.ac.kr/>.

[11] S. Lawrence, C. L. Giles1y, A. C. Tsoi, and A. D. Back. Face recognition: A convolutional neural network approach. *IEEE Transactions on Neural Networks, Special Issue on Neural Networks and Pattern Recognition*, 8(1):98–113, 1997.

[12] B. C. Lovell and S. Chen. *Robust Face Recognition for Data Mining in Encyclopedia of Data Warehousing and Mining*. Idea Group, 2005.

[13] A. M. Martinez and A. C. Kak. PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(2):228–233, 2001.

[14] L. Prevost, C. Michel-Sendis, A. Moises, L. Oudot, and M. Milgram. Combining model-based and discriminative classifiers : application to handwritten character recognition. In *Proceedings. Seventh International Conference on Document Analysis and Recognition*, Edinburgh, UK, 2003.

[15] R. Raina, Y. Shen, A. Y. Ng, and A. McCallum. Classification with hybrid generative/discriminative models. In *Proceedings of Neural Information Processing Systems*, Vancouver, British Columbia, Canada, 2004.

[16] F. Samaria and F. Fallside. Automated face identification using hidden markov models. In *Proceedings of the International Conference on Advanced Mechatronics*, Tokyo, Japan, 1993.

[17] M. A. Turk and A. P. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[18] R. R. Wang, T. Huang, and J. Zhong. Generative and discriminative face modelling for detection. In *5th IEEE International Conference on Automatic Face and Gesture Recognition (FGR 2002)*, Washington, D.C., USA, 2002.

[19] A. Y. Ng and M. I. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Proceedings of Advances in Neural Information Processing Systems*, Vancouver, British Columbia, Canada, 2002.