Available online at www.sciencedirect.com
ScienceDirect
Pattern Recognition 41 (2008) 2842–2854
PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY
www.elsevier.com/locate/pr

A study of regularized Gaussian classifier in high-dimension small sample set case based on MDL principle with application to spectrum recognition

Ping Guo[a,b,*], Yunde Jia[a], Michael R. Lyu[c]

[a]School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, PR China
[b]Laboratory of Image Processing and Pattern Recognition, Beijing Normal University, Beijing, 100875, PR China
[c]Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong SAR, PR China

Received 6 December 2005; received in revised form 24 January 2008; accepted 2 February 2008

Abstract

In classifying high-dimensional patterns such as stellar spectra by a Gaussian classifier, the covariance matrix estimated with a small-number sample set becomes unstable, leading to degraded classification accuracy. In this paper, we investigate the covariance matrix estimation problem for small-number samples with high dimension setting based on minimum description length (MDL) principle. A new covariance matrix estimator is developed, and a formula for fast estimation of regularization parameters is derived. Experiments on spectrum pattern recognition are conducted to investigate the classification accuracy with the developed covariance matrix estimator. Higher classification accuracy results are obtained and demonstrated in our new approach.

Keywords: Classification; Covariance matrix estimation; Discriminant analysis method; Regularization parameter selection; Minimum description length

1. Introduction

Spectrum recognition has a wide range of applications, such as chemical element identification, stellar classification, and matter structure analysis. For spectral data, the number of variables (wavelengths) is much higher than that of training samples; therefore, spectral data are severely ill-posed. Due to such high dimensionality, the common multivariate classification methods of linear discriminant analysis (LDA) and quadratic discriminant analysis (QDA) cannot be directly applied because of the matrix singularity problem [1].

Spectrum recognition is usually a high-dimensional small sample set classification problem. Generally speaking, classification has two aspects: supervised classification (discrimination or simply classification) and unsupervised classification (clustering). In recent years, several classification algorithms have been developed to partition a data set into pre-defined classes. When the data are viewed as arising from two or more clusters mixed in varying proportions, we can use the finite Gaussian mixture distribution to analyze the data set. The Gaussian mixture distribution analysis method has been employed widely in a variety of important practical situations, where the likelihood approach to the fitting of Gaussian mixture models has been utilized extensively [2–5].

When classifying data with the Gaussian mixture model, the mean vector and covariance matrix of each component are not known in advance, and they have to be estimated from the given data set. While a large-size data set is desirable for estimating the parameters more accurately, in the real world, often only a small-size data set can be obtained because of some restriction, e.g., high cost in collecting large-size data sets. For a relatively small-number sample data set, if the dimension d of variable $\mathbf{x}$ is comparable to the number of training samples n_j in class j, the problem may become poorly posed. Worse, if the number n_j of training samples is less than the dimensionality, the problem becomes ill-posed. In this case, not all parameters can be properly estimated, and the classification accuracy is degraded.

* Corresponding author at: School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, PR China. Tel.: +86 10 58806662; fax: +86 10 58809444.
E-mail addresses: pguo@ieee.org (P. Guo), lyu@cse.cuhk.edu.hk (M.R. Lyu).

0031-3203/$30.00 © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2008.02.004

There are two possible solutions for this kind of problem: one is dimensionality reduction [6,7], and the other is regularization [1,8]. Regularization is the procedure of allowing parameters bias toward what are thought to be more plausible values, which reduces the variance of the estimates at the cost of introducing bias. Besides the regularization techniques can be used to sparse nonparametric density estimation in high dimension case [9], the regularization techniques have been highly successful in classifying small-number data with some heuristic approximations [1,8,10,11]. However, these methods, such as regularized discriminant analysis (RDA) [10], require users to select regularization parameters (or called *model*) with some statistical techniques like leave-one-out cross-validation [11–14], which is computation-expensive. Furthermore, a recent study shows that cross-validation performance is not always good in the selection of linear models [15] in some cases. Therefore, it is worthy to further investigate this problem.

Originally proposed as an estimation criterion by Rissanen [16,17], the minimum description length (MDL) principle can be applied to universal coding, linear regression, and density estimation problems. *The central idea of this principle is to represent an entire class of probability distributions as models by a single "universal" representative model, such that it would be able to imitate the behavior of any model in the class. The best model class for a set of observed data is the one whose representative permits the shortest coding of the data. The MDL estimates of both the parameters and their total number are consistent; i.e., the estimates converge and the limit specifies the data generating model* [17]. The codelength[1] criterion of MDL involves in the Kullback–Leibler divergence [18,19]. MDL principle has a wide applications, such as clustering problem [20]. In this paper, based on the MDL principle with the mixture model analysis, we present the results of investigating covariance matrix estimation and regularization parameter selection in the Gaussian classifier for the small-sample set with high-dimension classification problem.

2. Theoretical background

2.1. Classification with finite Gaussian mixture model

In pattern recognition problem, we have a set of data samples, each consisting of measurements on a set of variables with associated labels, the class types. They are used as exemplars in the classifier design [21]. In clustering we need to estimate *prior* probability and *posterior* probability in the classifier design. If these probabilities are known, it becomes a classification problem. So clustering is more general than classification in the mixture model analysis case. Let us consider the general case first.

The data points $D = \{\mathbf{x}_i\}_{i=1}^{N}$ to be classified are assumed to be samples from a mixture of k Gaussian densities with joint probability density of which the mathematical expressions are shown as follows:

$$p(\mathbf{x}, \Theta) = \sum_{j=1}^{k} \alpha_j G(\mathbf{x}, \mathbf{m}_j, \mathbf{\Sigma}_j)$$

$$\text{with } \alpha_j \geqslant 0 \quad \text{and} \quad \sum_{j=1}^{k} \alpha_j = 1, \tag{1}$$

where

$$G(\mathbf{x}, \mathbf{m}_j, \mathbf{\Sigma}_j) = \frac{\exp[-\frac{1}{2}(\mathbf{x}-\mathbf{m}_j)^{\mathrm{T}}\mathbf{\Sigma}_j^{-1}(\mathbf{x}-\mathbf{m}_j)]}{(2\pi)^{d/2}|\mathbf{\Sigma}_j|^{1/2}} \tag{2}$$

is a general multivariate Gaussian density function, $\mathbf{x}$ denotes a random vector, d is the dimension of the $\mathbf{x}$, and parameter $\Theta = \{\alpha_j, \mathbf{m}_j, \mathbf{\Sigma}_j\}_{j=1}^{k}$ is a set of finite mixture model parameter vectors. Here α_j is the *prior* probability, $\mathbf{m}_j$ is the mean vector, and $\mathbf{\Sigma}_j$ is the covariance matrix of the jth component. Based on the given data set, these parameters can be estimated by the maximum likelihood (ML) method with expectation-maximum (EM) algorithm [22,23].

In the Gaussian mixture model case, the Bayesian decision rule is applied to classify the vector $\mathbf{x}$ into class j with the largest *posterior* probability. The *posterior* probability $P(j|\mathbf{x}, \Theta)$ represents the probability that the sample $\mathbf{x}$ belongs to class j. We use Bayesian decision $j^* = \arg\max_j P(j|\mathbf{x}, \Theta)$ to classify $\mathbf{x}$ into class j^*. The probability functions $P(j|\mathbf{x}, \Theta)$ are usually unknown and have to be estimated from the training samples. With the ML method estimated parameter $\widehat{\Theta}$, the *posterior* probability can be written in the form

$$P(j|\mathbf{x}, \widehat{\Theta}) = \frac{\widehat{\alpha}_j G(\mathbf{x}, \widehat{\mathbf{m}}_j, \widehat{\mathbf{\Sigma}}_j)}{p(\mathbf{x}, \widehat{\Theta})} \quad \text{with}$$
$$j = 1, 2, \ldots, k. \tag{3}$$

Taking the logarithm to the above equation and omitting the common factors of the classes, the classification rule becomes

$$j^* = \arg\min_j d_j(\mathbf{x}), \quad j = 1, 2, \ldots, k \tag{4}$$

with

$$d_j(\mathbf{x}) = (\mathbf{x}-\widehat{\mathbf{m}}_j)^{\mathrm{T}}\widehat{\mathbf{\Sigma}}_j^{-1}(\mathbf{x}-\widehat{\mathbf{m}}_j) + \ln|\widehat{\mathbf{\Sigma}}_j| - 2\ln\widehat{\alpha}_j. \tag{5}$$

This equation is often called the discriminant function for the jth class in the literature [1]. Furthermore, if the *prior* probability $\widehat{\alpha}_j$ is the same for all classes, the term $2\ln\widehat{\alpha}_j$ can be omitted and the discriminant function reduces to a simpler form [24].

2.2. Covariance matrix estimation

When the sample number is small, the sample-based estimation of class-specific covariance matrix becomes inaccurate, resulting in lowered classification accuracy. To solve this problem, several techniques are proposed, such as LOOC as well as its extensions bLOOC1 and bLOOC2 [11–14]. LOOC was proposed by Hoffbeck and Landgrebe [11], who examine the diagonal sample covariance matrix, the diagonal common covariance matrix, and some pair-wise mixtures of those matrices.

[1] A term codelength is just another way to express a probability distribution or a model.

They called the covariance matrix estimator as LOOC because the mixture parameter was optimized by leave-one-out cross-validation method. Bensmail and Celeux proposed eigenvalue decomposition discriminant analysis (EDDA) approach to solve covariance matrix estimation problem [25], the covariance matrix is constrained to some form, such as diagonal matrices. Later, a covariance estimator formulated under an empirical Bayesian setting was described in Ref. [12], which is named as BLOOC, and it can also be viewed as a compromise between the linear and quadratic classifiers. The maximization of leave-one-out average log likelihood is used as the criterion to select the appropriate mixture model. Note in Ref. [13], an improved regularized covariance estimator of each class with the advantages of LOOC and Bayesian LOOC (BLOOC) was presented. There are two forms of this covariance estimation depending on the form of covariance matrices used, namely bLOOC1 and bLOOC2. Because bLOOC1 and bLOOC2 suffer from drawbacks inherited from RDA, LOOC, and the empirical Bayesian covariance estimators, a new family of adaptive covariance estimators is presented in Ref. [14], which is produced by combining an adaptive classification process with various regularized covariance estimators such as LOOC, bLOOC1, and bLOOC2. The regularized parameters and supportive covariance matrices employed in a covariance mixture are determined based on both training samples and semi-labeled samples, and they are repeatedly updated until the highest classification accuracy is reached. These methods mainly concern various mixture of sample matrix and common matrix, where the regularized parameters are determined by maximizing average leave-one-out log likelihood criterion. Besides, the high variance problem can be addressed by ensemble-based discriminant analysis solutions [26]. More recently, Srivastava et al. [27] proposed a Bayesian QDA classifier termed BDA7. BDA7 differs from the previous Bayesian QDA methods in that the prior is selected by cross-validation from a set of data-dependent priors.

In this paper, we focus on the regularization method and address this problem based on MDL principle. The focus of interest in MDL principle is in various classes of probability distributions as models, which results in the modeling problems. Now we consider that a given sample data set $\mathbf{x}$ was generated from an unknown density $p(\mathbf{x})$, which can be modeled by a finite Gaussian mixture density $p(x, \Theta)$, where Θ is the parameter set. In the absence of knowledge of $p(\mathbf{x})$, it may be estimated by an empirical kernel density estimate $p_h(\mathbf{x})$ [28] obtained from the data set. Because these two probability densities describe the same unknown density $p(\mathbf{x})$, they should be best matched with proper mixture parameters and smoothing parameters. However, if we only make these two models close to each other without applying the given sample set, it may turn out that none of them is close to the true density underlying the sample.

According to MDL principle, the best model class for a set of observed data is the one whose representative permits the shortest coding of the data, and the system should be optimized with optimal or *ideal* code length. That is, the model parameters should be estimated with minimized Kullback–Leibler distance (also called divergence) $KL(h, \Theta)$ based on the given data set drawn from the unknown density $p(\mathbf{x})$. With the constraint of the given data set, the "distance" of these two probability densities can be measured with the following Kullback–Leibler divergence function in integration form [18,19],

$$KL(h, \Theta) = \int p_h(\mathbf{x}) \ln \frac{p_h(\mathbf{x})}{p(\mathbf{x}, \Theta)} \, \mathrm{d}\mathbf{x}, \tag{6}$$

which equals to zero only if $p_h(\mathbf{x}) = p(\mathbf{x}, \Theta)$. The above KL function, also called the system relative entropy or cost function in MDL principle, is described as the expected codelength difference (redundancy) and can be rewritten in the form

$$KL(h, \Theta) = -\int p_h(\mathbf{x}) \ln p(\mathbf{x}, \Theta) \, \mathrm{d}\mathbf{x} + \int p_h(\mathbf{x}) \ln p_h(\mathbf{x}) \, \mathrm{d}\mathbf{x}, \tag{7}$$

where

$$p_h(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} G(\mathbf{x}, \mathbf{x}_i, \mathbf{W}_h) = \frac{1}{N(2\pi)^{d/2}|\mathbf{W}_h|^{1/2}} \sum_{i=1}^{N} \times \exp\left[-\frac{1}{2} \sum_{j=1}^{d} \frac{(x_j - x_{i,j})^2}{h_j} \right] \tag{8}$$

is assigned as the Gaussian kernel density for the given samples. Here x_j is the jth component of the random variable $\mathbf{x}$, $x_{i,j}$ represents the jth component of the variable for data point i, and $\mathbf{W}_h$ is a $d \times d$ dimensional diagonal matrix with a general form,

$$\mathbf{W}_h = \mathrm{diag}(h_1, h_2, \ldots, h_d), \tag{9}$$

where $h_i, i = 1, 2, \ldots, d$ are smoothing parameters in the non-parametric kernel density. In the following we denote this set as $h = \{h_i\}_{i=1}^{d}$.

When we estimate parameter Θ, the second term in Eq. (7) can be neglected because it is not related to Θ. At the limit $h \to 0$, the kernel density function becomes a δ function, then Eq. (7) reduces to the negative log likelihood function. Minimizing KL function is equivalent to maximizing likelihood function in this case. The ordinary EM algorithm [22,23] can be re-derived based on the minimization of this KL divergence function with respect to mixture parameter Θ [21]. The ML-estimated covariance matrix $\widehat{\mathbf{\Sigma}}_j$ has the form

$$\widehat{\mathbf{m}}_j = \frac{\sum_{i=1}^{N} P(j|\mathbf{x}_i, \widehat{\Theta})\mathbf{x}_i}{\sum_{i=1}^{N} P(j|\mathbf{x}_i, \widehat{\Theta})} = \frac{1}{n_j} \sum_{i=1}^{N} P(j|\mathbf{x}_i, \widehat{\Theta})\mathbf{x}_i,$$

$$\widehat{\mathbf{\Sigma}}_j = \frac{1}{n_j} \sum_{i=1}^{N} P(j|\mathbf{x}_i, \widehat{\Theta})(\mathbf{x}_i - \widehat{\mathbf{m}}_j)(\mathbf{x}_i - \widehat{\mathbf{m}}_j)^{\mathrm{T}}. \tag{10}$$

Unlike the supervised learning, the ML with EM algorithm can be applied for a totally un-labeled training data set, for example, in an application of automatic image segmentation [29]. However, for small-number samples with high dimension, the ML-estimated covariance matrix $\widehat{\mathbf{\Sigma}}_j$ with the EM algorithm becomes singular when $n_j < d$, leading to an unstable

classification rate. To deal with this difficulty, one approach is *regularization*. In the following we address this problem based on Kullback–Leibler divergence with $h \neq 0$.

The smoothing parameter h in a Gaussian kernel density plays an important role in estimating the mixture model parameter. The most concerned problem is covariance matrix estimation in classification, where the mixture weight α_j and class mean $\mathbf{m}_j$ can be estimated with summation under the $h = 0$ approximation as in Eq. (10). Then we focus on covariance matrix estimation problem in the following. By minimizing Eq. (6) with respect to Σ, i.e., setting $\partial/\partial\Sigma_j KL(h, \Theta) = 0$, the following covariance matrix estimation formula can be obtained:

$$\widehat{\mathbf{\Sigma}}_j = \frac{\int p_h(\mathbf{x}) P(j|\mathbf{x}, \widehat{\Theta})(\mathbf{x} - \widehat{\mathbf{m}}_j)(\mathbf{x} - \widehat{\mathbf{m}}_j)^{\mathrm{T}} \, \mathrm{d}\mathbf{x}}{\int p_h(\mathbf{x}) P(j|\mathbf{x}, \widehat{\Theta}) \, \mathrm{d}\mathbf{x}}. \tag{11}$$

Note in the above equation the parameters still need an iterative estimation. There are several ways to evaluate the probability-type integration in each iterative estimation step. One of the techniques is the well-known *Monte Carlo integration* [30,31]. In the *Monte Carlo integration* approximation, we need to generate a huge number of samples in high-dimension setting. It seems that this method is very computation-expensive, although it can achieve a reasonable accuracy.

Another method to evaluate the integration is to apply the Taylor expansion approximation, which is employed in this paper. Because the $p_h(\mathbf{x})$ term is contained in the integration of Eq. (11), when $\mathbf{x}$ is far away from $\mathbf{x}_i$ and $p_h(\mathbf{x})$ is smooth near x_i, the function value becomes very small. In this case, we can use the Taylor expansion for $P(j|\mathbf{x}, \Theta)$ at $\mathbf{x} = \mathbf{x}_i$ with respect to $\mathbf{x}$ and take up to second order approximation:

$$\begin{aligned} P(j|\mathbf{x}, \widehat{\Theta}) \approx{}& P(j|\mathbf{x}_i, \widehat{\Theta}) + (\mathbf{x} - \mathbf{x}_i)^{\mathrm{T}} \nabla_x P(j|\mathbf{x}_i, \widehat{\Theta}) \\ &+ \tfrac{1}{2}(\mathbf{x} - \mathbf{x}_i)^{\mathrm{T}} \mathbf{H}_i(j)(\mathbf{x} - \mathbf{x}_i) \end{aligned} \tag{12}$$

with notation $\nabla_x P(j|\mathbf{x}_i, \widehat{\Theta}) = \nabla_x P(j|\mathbf{x}, \widehat{\Theta})|_{x=x_i}$. $\mathbf{H}_i(j)$ is the Hessian matrix of the *posterior* probability function, $\mathbf{H}_i(j) = \nabla_x \nabla_x P(j|\mathbf{x}, \widehat{\Theta})|_{x=x_i}$. It is computed as the following:

$$\begin{aligned} \mathbf{H}_i(j) = {}& P(j|\mathbf{x}_i)\{\mathbf{\Sigma}_j^{-1}(\mathbf{x}_i - \mathbf{m}_j)(\mathbf{x}_i - \mathbf{m}_j)^{\mathrm{T}}\mathbf{\Sigma}_j^{-1} \\ &- \sum_{l=1}^{k} P(l|\mathbf{x}_i)[\mathbf{\Sigma}_l^{-1}(\mathbf{x}_i - \mathbf{m}_l)(\mathbf{x}_i - \mathbf{m}_l)^{\mathrm{T}}\mathbf{\Sigma}_l^{-1}]\} \\ &+ P(j|\mathbf{x}_i)\left\{\sum_{l=1}^{k} P(l|\mathbf{x}_i)\mathbf{\Sigma}_l^{-1} - \mathbf{\Sigma}_l^{-1}\right\} \\ &+ 2P(j|\mathbf{x}_i)\left[\sum_{l=1}^{k} P(l|\mathbf{x}_i)\mathbf{\Sigma}_l^{-1}(\mathbf{x}_i - \mathbf{m}_l) \right. \\ &\left. -\mathbf{\Sigma}_l^{-1}(\mathbf{x}_i - \mathbf{m}_l)\right] \\ &\times \sum_{l=1}^{k} P(l|\mathbf{x}_i)(\mathbf{x}_i - \mathbf{m}_l)^{\mathrm{T}}\mathbf{\Sigma}_l^{-1}. \end{aligned} \tag{13}$$

On substituting the above equations into Eq. (11), Eq. (11) reduces to Gaussian type function integrals. These type integrals can be evaluated out (refer to [32, Appendix B]), leading to the following covariance matrix estimation formula:

$$\mathbf{\Sigma}_j^{(2)}(h) = (1 + \tfrac{1}{2}\mathrm{Trace}[\mathbf{W}_h\mathbf{H}(j)])\mathbf{W}_h + \frac{\mathbf{\Sigma_e}}{(1+\eta)} + \frac{\widehat{\mathbf{\Sigma}}_Q}{(1+\eta)}, \tag{14}$$

where $\mathbf{H}(j) = \frac{1}{n_j}\sum_{i=1}^{N} \mathbf{H}_i(j)$. The following notations are used in the above equations:

$$\begin{aligned} &\eta = \frac{1}{2n_j} S(h, j), \\ &S(h, j) = \sum_{i=1}^{N} \mathrm{Trace}[\mathbf{W}_h\mathbf{H}_i(j)], \\ &\mathbf{\Sigma_e} = \mathbf{W}_h\mathbf{H}_e\mathbf{W}_h \end{aligned} \tag{15}$$

$\mathbf{H}_e$ is a diagonal matrix in which the diagonal elements are the eigenvalues of $\mathbf{H}(j)$, and

$$\begin{aligned} \widehat{\mathbf{\Sigma}}_Q = \frac{1}{n_j}\sum_{i=1}^{N}[P(j|\mathbf{x}_i, \widehat{\Theta}) + \tfrac{1}{2}\mathrm{Trace}(\mathbf{W}_h\mathbf{H}_i(j))](\mathbf{x}_i - \widehat{\mathbf{m}}_j) \\ \times (\mathbf{x}_i - \widehat{\mathbf{m}}_j)^{\mathrm{T}}. \end{aligned}$$

Because this estimation is derived under the framework of Kullback–Leibler information measure, it is called as KLIM2 in this paper.

If we only consider the first order approximation, the estimate becomes

$$\mathbf{\Sigma}_j^{(1)}(h) = \mathbf{W}_h + \widehat{\mathbf{\Sigma}}_j. \tag{16}$$

This estimation is called as KLIM1 in this paper, in which $\widehat{\mathbf{\Sigma}}_j$ is the ML estimation at $h = 0$, taking the form of Eq. (10).

The quantity in the form $\sum_{j=1}^{k} P(j|\mathbf{x}, \Theta)Q(j)$ represents the weighted average value $Q(j)$ over all classes. The above Hessian equation reflects the difference between single class quantity and averaged class quantity. If there is only one class, this Hessian matrix will become a null matrix and $\mathbf{\Sigma}_j^{(2)}(h)$ reduces to $\mathbf{\Sigma}_j^{(1)}(h)$ automatically.

From the above, we can see that new kinds of regularized covariance matrices, thereof regularized Gaussian classifiers, are obtained based on MDL principle with Kullback–Leibler information measure. Note the smoothing parameter h controls the degree of regularization, and it plays a role of regularization parameter. Because multi-parameter optimization is more difficult than single parameter optimization, in this paper we only consider one special case that the smoothing parameters are the same for all dimensions. Namely,

$$\mathbf{W}_h = h\mathbf{I}_d, \tag{17}$$

where $\mathbf{I}_d$ is a $d \times d$ dimensional identity matrix.

In the next section we discuss how an optimal value of smoothing parameter h can be selected based on the training samples.

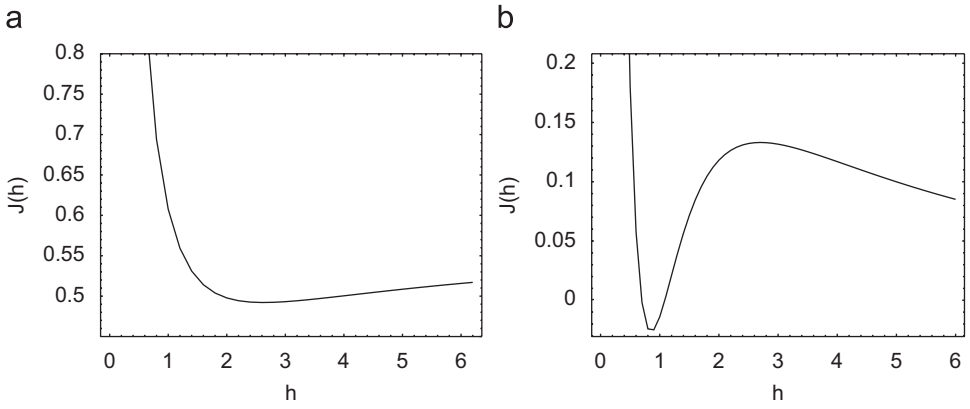

Fig. 1. The $J(h)$ function with some approximations. (a) $J(h)$ vs. h curve computed by *Monte Carlo integration* approximation. (b) $J(h)$ vs. h curve computed by second order approximation with Eq. (A.6).

3. Smoothing parameter selection

Different h will generate different models; therefore, to select the smoothing parameter is to select a model. There are several ways to select a proper smoothing parameter h. For example, with training samples we can use the statistical cross-validation technique to select the smoothing parameter based on an optimal classification rate, such as what is done in RDA [10] and in LOOC [11]. As we know, the goal in selecting the smoothing parameter is to produce a model for the probability density which is as close as possible to the unknown density $p(\mathbf{x}, \Theta)$ [32]. From the experience in empirical kernel density estimate, we know that when sample number $N \to \infty$, $p_h(\mathbf{x})$ will best describe the unknown density with $h \to 0$. When N becomes small, we should increase h in order to get a plausibly true density estimate. However, if h is too big, the estimate will become too smooth and far away from the true density. There should exist an optimal h value, but to select an optimal h for regularized estimate is a very difficult problem. In fact, we should choose it under some criteria to obtain a plausible estimate for the true density solution [33]. To this end, we propose to select a reasonable h parameter based on MDL principle with given data.

According to the principle of MDL, it should be with the shortest codelength to select a model [34]. When $h \neq 0$, the smooth parameter h can be estimated with the minimized KL divergence regarding h with ML estimated parameter Θ^*,

$$h^* = \arg\min J(h), \quad J(h) = KL(h, \Theta^*). \tag{18}$$

This is the MDL model selection criterion for the problem discussed here. In practical implementation, we can use an exhaustive search method to find h. That is, for each h, we compute the $J(h)$ function values to search the minima of $J(h)$, and choose h^* that minimizes $J(h)$. Note that in this approach all the samples can be applied to estimate h^*. Therefore, it is different from the cross-validation method which must split data into a training set and a validation set.

When selecting the optimal h, we have to evaluate the integration equation of $J(h)$. The integral can be approximated by the *Monte Carlo method* [29,31]. Using the *Monte Carlo integration* with the extrapolation method we can compute h with higher accuracy. However, the task is very computational-intensive, especially in the high dimensionality case which we deal with in this paper.

Now we propose to use a second order approximation for estimating the smoothing parameter h. The rough estimation formula is obtained as

$$h = \frac{d}{2J_r(\Theta)}. \tag{19}$$

Readers can refer to Appendix A for the details of deriving this formula. With this equation, h now can be estimated with less computation-expensive effort.

When the Taylor approximations are engaged, the computation cost will be significantly reduced. However, with rough approximations the obtained value is not as accurate as computing with an integration. Figs. 1a and 1b are typical $J(h)$ vs. h curves, showing the difference of the *Monte Carlo method* and Taylor approximations in estimating the smoothing parameter.

4. Approximations

4.1. Approximations for regularization term

In practice, the computation of η and Hessian matrix of Eq. (15) is still quite complicated, and some proper approximations should be adopted to simplify the calculation. Now let us consider a special approximation of matrix $\mathbf{H}(j)$.

The estimation of eigenvalue matrix of *posterior* probability Hessian is an iterative procedure, where initialization is necessary. At the beginning, we do not know the true distribution of the samples; therefore, the form of matrix $\mathbf{\Sigma}_j$ is also unknown. One of the assumptions is to let $\mathbf{\Sigma}_j = \mathbf{I}_d$ be the initial value. Under this assumption, the *posterior* probability Hessian

matrix becomes

$$\mathbf{H}(j) = \frac{1}{n_j}\sum_{i=1}^{N}\left\{P(j|\mathbf{x}_i,\Theta)\left\{(\mathbf{x}_i-\mathbf{m}_j)(\mathbf{x}_i-\mathbf{m}_j)^{\mathrm{T}} - \sum_{l=1}^{k}P(l|\mathbf{x}_i,\Theta)[(\mathbf{x}_i-\mathbf{m}_l)(\mathbf{x}_i-\mathbf{m}_l)^{\mathrm{T}}]\right\} + 2P(j|\mathbf{x}_i,\Theta)\left[\sum_{l=1}^{k}P(l|\mathbf{x}_i,\Theta)(\mathbf{x}_i-\mathbf{m}_l)\times-(\mathbf{x}_i-\mathbf{m}_l)\right] \times\sum_{l=1}^{k}P(l|\mathbf{x}_i,\Theta)(\mathbf{x}_i-\mathbf{m}_l)^{\mathrm{T}}\right\}. \quad (20)$$

Furthermore, if we regard $\sum_{l=1}^{k}P(l|\mathbf{x}_i,\Theta)\mathbf{m}_j=\mathbf{m}$ as an averaged mean, the above equation becomes

$$\mathbf{H}(j) = \widehat{\boldsymbol{\Sigma}}_j - \widehat{\boldsymbol{\Sigma}} + \frac{2}{n_j}\sum_{i=1}^{N}P(j|\mathbf{x}_i,\Theta)(\mathbf{x}_i-\mathbf{m})(\mathbf{x}_i-\mathbf{m})^{\mathrm{T}} - \frac{2}{n_j}\sum_{i=1}^{N}P(j|\mathbf{x}_i,\Theta)(\mathbf{x}_i-\mathbf{m}_j)(\mathbf{x}_i-\mathbf{m})^{\mathrm{T}}. \quad (21)$$

The last term represents the cross-variance between individual class and common class. If class overlapping part is very small, this term has very little significance, and can be omitted. The third term can be regarded as approximately equivalent to two times of the common covariance matrix. Under this approximation, the Hessian matrix can be written as

$$\mathbf{H}(j) = \widehat{\boldsymbol{\Sigma}}_j + \widehat{\boldsymbol{\Sigma}}. \quad (22)$$

When we rearrange the term in KLIM2, it leads to

$$\boldsymbol{\Sigma}_j^{(2)}(h) = \left(1+\frac{h}{2}\mathrm{Trace}[\widehat{\boldsymbol{\Sigma}}_j+\widehat{\boldsymbol{\Sigma}}]\right)h\mathbf{I}_d + \frac{h^2}{(1+\eta)}ev(\widehat{\boldsymbol{\Sigma}}_j+\widehat{\boldsymbol{\Sigma}}) + \frac{\widehat{\boldsymbol{\Sigma}}_Q}{(1+\eta)}, \quad (23)$$

where $ev(\boldsymbol{\Sigma})$ stands for a diagonal matrix in which the diagonal elements are the eigenvalues of $\boldsymbol{\Sigma}$.

From these approximations, it is known that the second order regularization involves calculation of the inverse covariance matrix. In the case of $n_j > d$, we can use Eq. (10) to estimate the initial value of $\boldsymbol{\Sigma}_j$. While for the case $n_j < d$, $\widehat{\boldsymbol{\Sigma}}_j$ becomes singular. With KLIM1 estimator, however, $\boldsymbol{\Sigma}_j$ is not singular as long as h is not too small. In this case, we adopt KLIM1 estimated covariance matrix as the initial value to calculate $\mathbf{H}(j)$ and η.

In fact, if we let the eigenvector and eigenvalue of a covariance matrix $\widehat{\boldsymbol{\Sigma}}_j$ be $\mathbf{u}_i$ and v_i, respectively,

$$\widehat{\boldsymbol{\Sigma}}_j\mathbf{u}_i = v_i\mathbf{u}_i, \quad \mathbf{u}_i^{\mathrm{T}}\mathbf{u}_j = \delta_{ij}, \quad (24)$$

where the inverse matrix of $\boldsymbol{\Sigma}_j$ in KLIM1 can be expressed as

$$\boldsymbol{\Sigma}_j^{-1} = (h\mathbf{I}_d + \widehat{\boldsymbol{\Sigma}}_j)^{-1} = \sum_{i=1}^{d}\frac{\mathbf{u}_i\mathbf{u}_i^{\mathrm{T}}}{v_i+h} \quad (25)$$

then,

$$\mathrm{Trace}[\boldsymbol{\Sigma}_j^{-1}] = \sum_{i=1}^{d}\frac{1}{v_i+h}. \quad (26)$$

If $\widehat{\boldsymbol{\Sigma}}_j$ is singular, then $|\widehat{\boldsymbol{\Sigma}}_j| = \mathbf{0}$ and hence $|v\mathbf{I} - \widehat{\boldsymbol{\Sigma}}_j| = \mathbf{0}$. This means that a singular matrix has at least one zero eigenvalue, resulting in one term being proportional to h^{-1} in Eq. (26). It is clearly seen that as long as h is not too small, $\boldsymbol{\Sigma}_j^{-1}$ exists with a finite value and the estimated classification rate is stable.

4.2. Approximations in regularization parameter selection

The function $J_r(\Theta)$ can be simplified with an approximation. From Eqs. (A.3),

$$J_r(\Theta) = -\frac{1}{2N}\mathrm{Trace}\sum_{i=1}^{N}[\nabla_x\nabla_x\ln p(\mathbf{x}_i,\Theta)] \quad (27)$$

while

$$-\sum_{i=1}^{N}\nabla_x\nabla_x\ln p(\mathbf{x}_i,\Theta) = \sum_{i=1}^{N}\left\{\sum_{j=1}^{k}P(j|\mathbf{x}_i,\Theta)[\boldsymbol{\Sigma}_j^{-1} - \boldsymbol{\Sigma}_j^{-1}(\mathbf{x}_i-\mathbf{m}_j)(\mathbf{x}_i-\mathbf{m}_j)^{\mathrm{T}} \times\boldsymbol{\Sigma}_j^{-1}] + \left[\sum_{j=1}^{k}P(j|\mathbf{x}_i,\Theta)[(\mathbf{x}_i-\mathbf{m}_j)^{\mathrm{T}}\boldsymbol{\Sigma}_j^{-1}]^{\mathrm{T}}\right] \times\left[\sum_{j=1}^{k}P(j|\mathbf{x}_i,\Theta)[(\mathbf{x}_i-\mathbf{m}_j)^{\mathrm{T}}\boldsymbol{\Sigma}_j^{-1}]\right]\right\}.$$

Using the approximations $\sum_{i=1}^{N}P(j|\mathbf{x}_i,\Theta)(\mathbf{x}_i-\mathbf{m}_j)(\mathbf{x}_i-\mathbf{m}_j)^{\mathrm{T}} \approx n_j\widehat{\boldsymbol{\Sigma}}_j$, $\sum_{i=1}^{N}P(j|\mathbf{x}_i,\Theta)\approx n_j$, and $\boldsymbol{\Sigma}_j^{-1}\widehat{\boldsymbol{\Sigma}}_j\approx\mathbf{I}_d$, the first term in Eq. (27) is equal to zero. In considering hard-cut case, the cross-terms in the second term of the above equation can be omitted. The equation is reduced to

$$J_r(\Theta) = \frac{1}{2N}\mathrm{Trace}\sum_{i=1}^{N}[-\nabla_x\nabla_x\ln p(\mathbf{x}_i,\Theta)] \approx\frac{1}{2N}\mathrm{Trace}\sum_{j=1}^{k}\sum_{i=1}^{n_j}\boldsymbol{\Sigma}_j^{-1}(\mathbf{x}_i-\mathbf{m}_j)(\mathbf{x}_i-\mathbf{m}_j)^{\mathrm{T}}\boldsymbol{\Sigma}_j^{-1} = \frac{1}{2}\mathrm{Trace}\left\{\sum_{j=1}^{k}\alpha_j\boldsymbol{\Sigma}_j^{-1}\right\}.$$

From Eq. (19), we get

$$h = \frac{d}{2J_r(\Theta)} \approx \frac{d}{\text{Trace}[\sum_{j=1}^{k} \alpha_j \boldsymbol{\Sigma}_j^{-1}]}. \tag{28}$$

Further, we can take the mean approximation [35], and let

$$\sum_{j=1}^{k} \alpha_j \boldsymbol{\Sigma}_j^{-1} = \boldsymbol{\Sigma}^{-1}. \tag{29}$$

To estimate the regularization parameter, we should avoid calculating the inverse of a covariance matrix in ill-posed case. We now take the following approximations: using the average eigenvalue to substitute each eigenvalue of a matrix $\boldsymbol{\Sigma}$, that is,

$$v_i = \bar{v}, \quad i = 1, 2, \ldots, d, \tag{30}$$

where

$$\bar{v} = \frac{1}{d} \sum_{i=1}^{d} v_i = \frac{1}{d} \text{Trace}[\boldsymbol{\Sigma}]. \tag{31}$$

Then we have

$$h \approx \frac{1}{\text{Trace}\,[\boldsymbol{\Sigma}^{-1}]} \approx \frac{1}{d/\bar{v}} = \frac{\bar{v}}{d}, \tag{32}$$

$$h = \frac{1}{d^2} \text{Trace}[\boldsymbol{\Sigma}]. \tag{33}$$

This means h can be estimated as $1/d$ average of the eigenvalues of the common matrix. Note that we can use the whole data points to estimate $\widehat{\boldsymbol{\Sigma}}$ as an approximation of $\boldsymbol{\Sigma}$,

$$\widehat{\boldsymbol{\Sigma}} = \frac{1}{N} \sum_{i=1}^{N} (\mathbf{x}_i - \widehat{\mathbf{m}})(\mathbf{x}_i - \widehat{\mathbf{m}})^{\mathrm{T}}, \tag{34}$$

$$\widehat{\mathbf{m}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i, \tag{35}$$

$$\begin{aligned} \text{Trace}[\widehat{\boldsymbol{\Sigma}}] &= \frac{1}{N} \sum_{i=1}^{N} \text{Trace}(\mathbf{x}_i - \widehat{\mathbf{m}})(\mathbf{x}_i - \widehat{\mathbf{m}})^{\mathrm{T}} \\ &= \frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{d} (x_{i,k} - m_k)^2 \\ &= \frac{1}{2N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2 \\ &= \frac{1}{2N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \|\mathbf{x}_i - \mathbf{x}_j\|^2. \end{aligned} \tag{36}$$

Then h in Eq. (33) can be estimated with the formula

$$h = \frac{1}{2d^2N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \|\mathbf{x}_j - \mathbf{x}_i\|^2. \tag{37}$$

Hence, we derive a simplified model selection criterion for the particular case studied in this paper.

4.3. Discussion of KLIM with RDA and LOOC

In this section we examine Eq. (23) in order to compare KLIM with other regularized matrix estimators. The term $h^2/2\,\text{Trace}[\widehat{\boldsymbol{\Sigma}}_j + \widehat{\boldsymbol{\Sigma}}])\mathbf{I}_d$ in the expression is very similar to the term $\gamma/d\,\text{Trace}[\boldsymbol{\Sigma}_j(\lambda)]\mathbf{I}_d$ of the RDA form in Ref. [10], but the coefficients are different. While the terms of diagonal matrices are similar to LOOC (especially if $\widehat{\boldsymbol{\Sigma}}$ and $\widehat{\boldsymbol{\Sigma}}_j$ are diagonal, the eigenvalue matrix $ev(\boldsymbol{\Sigma})$ is equal to diagonal($\boldsymbol{\Sigma}$)); again the coefficients are quite different. The most important difference between KLIM and RDA is that RDA uses two parameters to control regularization, while KLIM uses a single parameter h. On the other hand, LOOC uses a single parameter to control the mixing of two covariance matrices, while KLIM just uses the same portion by adding eigenvalue matrices of sample and common covariance matrices.

KLIM is derived under the framework of MDL principle, while RDA and LOOC are heuristically proposed. KLIM, RDA and LOOC are similar in that they all consider ML estimated covariance matrix with the addition of extra matrices. KLIM and RDA both engage an identity matrix multiplied by a scalar; however, the scalar term is different from each other. There is also a term in KLIM2 which is the eigenvalue matrix of *posterior* probability Hessian, while RDA considers it with LDA estimation, and LOOC considers it with the diagonal sample or common covariance matrix. Moreover, the ML estimate in KLIM2 has the regularization coefficient related to the difference between averaged class quantities and single class quantities, while RDA is simply related to the sample covariance matrix estimation.

KLIM is different from LOOC in that LOOC considers mixing sample covariance matrix and its diagonal or common covariance. However, the value of mixing parameter ξ_j in LOOC is still selected by using leave-one-out cross-validation statistical methods. In KLIM, on the other hand, the regularization parameter is the smoothing parameter in kernel density estimation, which can be selected based on the model selection criteria derived under the MDL principle with all samples. While in RDA, the regularization parameter is heuristically proposed, which must employ some statistical methods, such as bootstrap or leave-one-out cross-validation, for optimization. In this regard, RDA requires much more computation than KLIM.

In a special approximation we can get $h = 1/d\,\text{Trace}[\boldsymbol{\Sigma}]/d$. In RDA, if we let $\lambda = 1$ and $\gamma = 1$, the identity matrix coefficient will reduce to the same form as above except for a factor $1/d$. Here we can see that even in the first order approximation, KLIM and RDA still maintain certain relationship. This demonstrates that under the framework of MDL principle, we can establish an analytic link between RDA and KLIM.

5. Experiment results

The main objective of our experiments is to assess the performance of KLIM estimator and regularization parameter selection criterion. In order to verify the proposed methods, we applied two real-world spectral data sets and conducted a series of experiments.

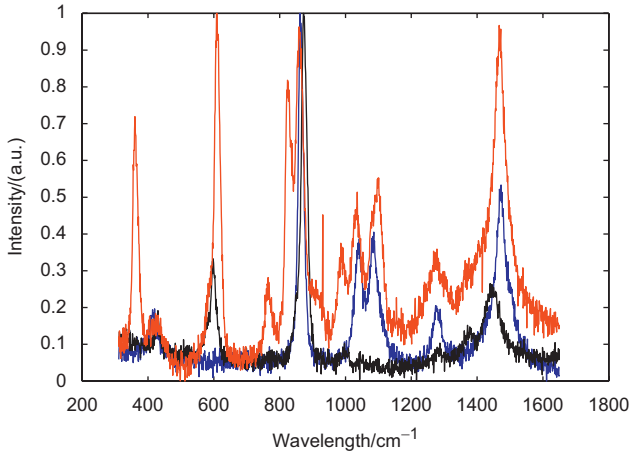

Fig. 2. Representative spectrum for each of the three Raman data classes.

Table 1
Raman data set used for classification experiments

Dimension d	Class 1 (ethanol)	Class 2 (acetic acid)	Class 3 (ethyl acetate)
134	30	50	290

5.1. Raman spectra data

5.1.1. Data description

This real-world data set was collected in the laboratory experiments from the physics department at Peking University. It includes 37 Raman spectra whose wave number ranges are from 320 to 1640 cm^{-1}, where each spectrum is measured by the charge-coupled device (CCD) detector with 1340 effective channels at the same time. The raw data set consists of three classes: 3 Raman spectra for ethanol, 5 for acetic acid, and 29 different time measures in synthesizing ethyl acetate. The dimension of the raw data set is 1340 and the sample number is 37. Fig. 2 shows the representative spectrum for each of the three classes.

Because the dimension of the raw data set is very high and the data number is too small, we first divide each sample into 10 sub-samples. From every 10 variables of the raw spectrum vector, one point is drawn and used to construct a new sample vector. By this method, each original sample is subdivided into 10 samples, and the dimension d from 1340 is reduced to 134. We also scale the intensity of Raman spectra to the range of [0,1].

Consequently, the data set we engaged for classification experiments is shown in Table 1. Here we should indicate that the classification results are comparable because we engage this same preprocessed data set for LDA, QDA, RDA, KLIM1, and KLIM2, and the results are not dependent on preprocessing of the data set.

5.1.2. Experiment

In order to study the performance of the regularized classifiers, we apply bootstrap technique [36] to conduct the experiments. The experiment for KLIM1 is described as follows:

Step 1: Randomly draw 20 training samples from each class, and apply them to estimate the mean and covariance matrices with Eq. (10).

Step 2: h is estimated with formula (37).

Step 3: Regularized covariance matrix is estimated with Eq. (16).

Step 4: The remaining samples are employed as test samples to verify classification accuracy. Using Eq. (4) to calculate the class label, then count all test samples to find correct classification rate.

The experiments are repeated 26 times, and the obtained results are the averaged values. The same data set is engaged with different classification methods. This is still an ill-posed problem because of n_j=20, which is much smaller than d=134. In this case, when we apply LDA[2] and QDA to this Raman spectra data set, they fail to give reliable classification results because the covariance matrix is singular.

On the other hand, the bootstrap experiments show that RDA gives an averaged classification accuracy of 99.27% with the standard deviation of 0.43, while the classification accuracy for KLIM1 and KLIM2 both reach 99.81% with the standard deviation of 0.28. The results also illustrate that these three classes are well separated from each other in the high-dimension space.

5.2. Stellar spectra data

5.2.1. Data description

There are seven main spectral types of stars in the order of decreasing temperature, namely: O—He II absorption, B—He I absorption, A—H absorption, F—Ca II absorption, G—strong metallic lines, K—bands developing, and M—very red. We employ 111 selected spectra from four spectral classes of stellar spectra as samples. Wavelength of the spectra ranges from 350 to 750 nm. Each of them contains a continuum spectrum and some absorption lines. The raw data set consists of four classes: B-type, A-type, F-type, and G-type. The representative spectrum for each of these classes is shown in Fig. 3.

Therefore, the data set we engaged for classification experiments is d = 745, N = 111, where class B-type has 23 samples, class A-type has 30 samples, and class F-type and class G-type each has 29 samples, respectively.

5.2.2. Experiment

For this data set, as the same as Raman spectra data, we also apply bootstrap technique to conduct the experiments with the same procedure. n_j = 15 training samples are randomly drawn from each class and applied to estimate the mean and covariance matrices. The remaining samples are employed as test samples to verify classification accuracy. The same data set is used with different classification methods. In the d=745 case, when we apply LDA and QDA to this stellar spectra data set,

[2] The terminology LDA used in this paper is from Ref. [1]. It is different from Refs. [6,37], in which LDA is usually called Fisher linear discriminant analysis (FDA) in the literature.

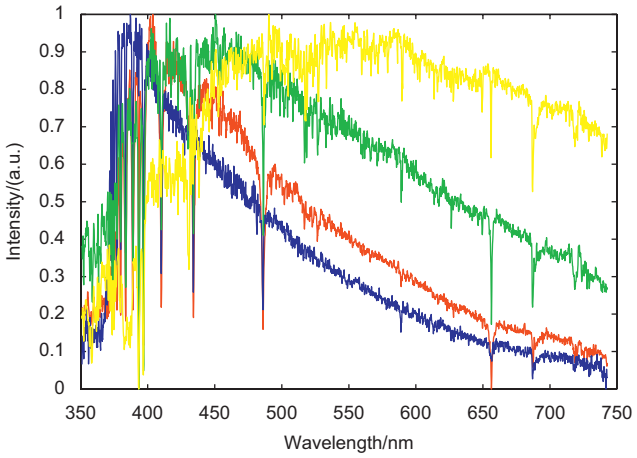

Fig. 3. Representative spectrum for each of the four stellar data classes.

Table 2
Mean classification accuracy for stellar eigen-spectra

	$d=6$	$d=10$	$d=20$	$d=40$
LDA	88.91 (4.26)	90.57 (3.23)	87.18 (4.41)	79.18 (7.27)
QDA	88.31 (3.11)	84.99 (4.43)	–	–
RDA	89.74 (4.12)	89.67 (4.06)	88.59 (3.17)	90.35 (4.03)
KLIM	89.97 (3.52)	90.80 (3.54)	91.25 (2.30)	91.55 (4.14)

they are unable to provide reliable classification results, again because the covariance matrix is singular. On the other hand, the bootstrap experiments show that RDA gives an averaged classification accuracy of 88.23% with the standard deviation of 3.37, while the classification accuracy for KLIM1 and KLIM2 are the same, both reaching 90.27% with the standard deviation of 3.48.

Because the dimension of Stellar spectra data is as high as $d=745$, when we try to compare the results with different dimensions, such as 6, 10, 20, and 40 dimensions, we need to adopt some dimension reduction methods to get the desired dimension from the original 745 dimension. Principal component analysis (PCA) [38] as the most used dimensionality reduction technique can be engaged directly for ill-posed problem by extracting principal component to form eigen-spectra, thus the number of variables in eigen-spectra can be made lower than the number of samples. In this work we adopt PCA for the purpose of getting a lower-dimension data set. In the experiments of eigen-spectra classification, the condition is the same as the above, while the dimension of eigen-spectra is reduced. The classification results for eigen-spectra are shown in Table 2 . In Table 2, d stands for the first d-dimension of the eigen-spectra, the classification accuracy is reported in percentage, and the value in parentheses represents the standard deviation. Furthermore, the dash lines represent that the covariance matrix is singular, in which case reliable results cannot be obtained.

With the developed fast regularization parameter estimation method, KLIM demonstrates a slightly higher classification accuracy than RDA. Besides, in all stellar eigen-spectra experiments, KLIM estimators are consistently better than LDA and QDA estimators. From the experiments, we can see that classification accuracy in high-dimension setting is usually higher than that in low-dimension. The results also illustrate that by only adopting PCA to reduce dimension can solve the ill-posed problem, but its classification accuracy will degrade [39]. The reason is that some samples can be separated in high-dimension space, but when projecting into reduced dimension space, they cannot be separated anymore.

5.3. Face image data

5.3.1. Data description

Face recognition is a typical small sample size recognition problem [40–42]; therefore, face data can be applied to assess the performance of KLIM estimator and regularization parameter selection criterion. ORL face database [43] is a popular database in face recognition research. This database consists of 40 persons, with each person's face appearing in 10 images, for the total of 400 images altogether. The images are taken at different time instances, with different lighting conditions, facial expressions (open/closed eyes, smiling/not smiling) and facial details (glasses/no glasses). All the images were taken against a dark homogeneous background with the faces in an upright, frontal position, including tolerance for some movement. All of the images are 112×92 in size. Fig. 4 shows samples of the face images for two persons taken at different time instances.

5.3.2. Experiment

We randomly select seven persons from the ORL database to construct the data set used in the experiments. Therefore, the data set we engaged for classification experiments is $d=10304$, $N=70$, where there are seven classes, each with 10 samples.

In the experiments, as the same as above, we still apply bootstrap technique. $n_j=6$ training samples are randomly drawn from each class and applied to estimate the mean and covariance matrices. The remaining samples are employed as test samples to verify the classification accuracy. The same data set is used with different classification methods. We also adopt PCA for the purpose of getting a lower dimension data set. The classification results for eigen-face are shown in Table 3. In Table 3, d stands for the first d-dimension of the eigen-face, the classification accuracy is reported in percentage, and the value in parentheses represents the standard deviation. Furthermore, the dash lines represent that the covariance matrix is singular, in which case reliable results cannot be obtained.

In this eigen-face experiment, KLIM estimators are consistently better than LDA and QDA estimators. From the experiments, we can find that classification accuracy of KLIM is better than that of LOOC, and is only a litter worse than that of RDA on average. However, the computation time to estimate regularization parameters for RDA and LOOC is much longer than that of KLIM.

5.4. Discussions

From these experiments and previous experiments with synthetic data [44], we see that the performance of various

Fig. 4. Examples of face images for two persons in the ORL face database.

Table 3
Mean classification accuracy for eigen-face data

	$d=5$	$d=10$	$d=20$	$d=40$
LDA	96.0 (0.18)	99.57 (0.01)	99.43 (0.02)	–
QDA	80.86 (1.4)	–	–	–
RDA	97.57 (0.19)	99.71 (0.02)	99.86 (0.0)	100 (0)
LOOC	93.0 (0.26)	98.0 (0.11)	98.29 (0.1)	98.14 (0.11)
KLIM	97.43 (0.13)	100 (0.0)	99.43 (0.02)	99.0 (0.03)

classification is generally data-dependent. For example, if all the classes have the same covariance matrix, LDA will lead to a higher classification accuracy than that of QDA. The synthetic data experiment illustrates that the classification accuracy strongly depends on the degree of overlapping between classes. If the classes are heavily overlapped with highly unequal ellipsoidal distribution, LDA's performance will be very poor. With properly selected smoothing parameter, however, KLIM1 is better than RDA except in one case ($d=20$). In the real-world spectra data experiments, KLIM's performance is better than all other estimators. These experimental results indicate that KLIM1 covariance matrix estimator can lead to a higher classification accuracy, suggesting that KLIM1 is simple and good-enough for most cases. According to the hypothesis tests (t-test) applied, there is no statistical evidence ($\alpha=0.05$) for stating the difference of performance between the RDA and KLIM1 with synthetic data set (p-value equals to 0.161). On the other hand, KLIM has the mean of the classification error significantly ($\alpha=0.05$) smaller than that of RDA with real-world data set (p-value equals to 0.012).

From these experiments, we also find that the smoothing parameter values for KLIM do not require stringent accuracy, as there exists a range of values in which a higher classification accuracy can be obtained. This range depends on training samples distribution. In most cases, however, the smoothing parameter selection methods employed in this paper work quite well.

In comparing KLIM1 and KLIM2, KLIM2 estimator leads to the same or a higher classification accuracy than KLIM1 in poorly posed problems, but its performance is not as good in ill-posed problems. One of the possible reason is that in ill-posed cases, the computation of covariance matrix $\widehat{\boldsymbol{\Sigma}}_j$ is highly varying, resulting in a large difference in value between averaged class quantity and single class quantity. This leads to strong regularization in estimating $\widehat{\boldsymbol{\Sigma}}_Q$, consequently deteriorating KLIM2 estimation. Nevertheless, this phenomenon occurs only in the cases where classes are heavily overlapped. When the classes are well separated from each other, the probability of $\mathbf{x}_i$ belonging to only one class will approach 1, resulting in $\eta \to 0$ and KLIM2 automatically reduces to KLIM1. The results in experiments with synthetic data and from the spectra data set, consequently, show that there is little or no significant difference between KLIM1 and KLIM2.

The regularization parameter selection is a crucial problem in regularized covariance matrix estimation. In the literature, the most applied method to selecting regularization parameters is with leave-one-out cross-validation statistical techniques [11–14]. The direct implementation of the leave-one-out likelihood function for each class with N_i training samples would require the computation of N_i matrix inverses and determinants at each value of the regularization parameter, which is computation-expensive. For the particular case studied in this paper, we derive a simplified regularization parameter selection criterion (Eq. (37)) based on MDL principle. It gives a theoretical guide to select regularization parameter, which is simple for users, and can significantly reduce computation time without loss of the classification accuracy. Using this formula to estimate regularization parameter obviously requires much less computation work than using the cross-validation method. If we count the order of computation time is $\mathcal{O}(N^2)$, while the cross-validation method requires at least an order of $\mathcal{O}(N^4)$ computation. In the face recognition experiments, the cross-validation method takes about 20.51 s for regularization parameters estimation in the case of $d=40$, while using the estimation formula to calculate the regularization parameter, it only takes about 0.0037 s, this is consistent with the above analysis.

Compared with the following regularization parameter estimation formula which we proposed heuristically in Ref. [45],

$$h=\frac{1}{dN^3}\sum_{i=1}^{N}\sum_{j=1}^{N}\|\mathbf{x}_j-\mathbf{x}_i\|^2, \tag{38}$$

we can find that it is a special case when $2d=N$ in Eq. (37). This clearly demonstrates that our heuristic method has established a theoretical basis.

6. Conclusions

In this paper, based on MDL principle, the KLIM covariance matrix estimation was derived and investigated for the

classification problem. An efficient smoothing parameter approximation formula was derived, and the approximation was found to be accurate for most cases in our experiments. With the *KL* information measure, total samples can be used to estimate the smoothing parameter, making it less computation-expensive than using leave-one-out cross-validation method proposed in the literature. With the MDL principle based estimation method, more than half of the experiments show that the obtained KLIM estimators work well, as they achieve higher classification accuracy than RDA. Besides, in all experiments, KLIM estimators are consistently better than QDA and LDA estimators. However, there are still some open problems to investigate as future work. For example, we can compare the dimensionality reduction with regularization technique in various high-dimension data sets. Also, we can study the kernel method combined with discriminant analysis as well as their applications to the face recognition problem. We can further study *Monte Carlo method* in the application of covariance matrix estimation in detail, and investigate the classification accuracy problem in data sets with heavily overlapped classes.

Acknowledgments

The authors wish to thank the anonymous reviewers for their useful suggestions and comments on the paper. And the authors also wish to express their thanks to Ms. Yunfei Jiang for her help on part experiment work.

This research work was fully supported by a grant from the National Natural Science Foundation of China (Project no. 60675011) and a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CUHK4150/07E).

Appendix A.

Derive the smooth parameter estimate formula with Taylor approximations.

Rewrite the integral in the form

$$J(h) = -\int p_h(\mathbf{x}) \ln p(\mathbf{x}, \Theta)\, d\mathbf{x} + \int p_h(\mathbf{x}) \ln p_h(\mathbf{x})\, d\mathbf{x} = J_0(h) + J_e(h), \tag{A.1}$$

where

$$J_0(h) \equiv -\int p_h(\mathbf{x}) \ln p(\mathbf{x}, \Theta)\, d\mathbf{x}$$

$$J_e(h) \equiv \int p_h(\mathbf{x}) \ln p_h(\mathbf{x})\, d\mathbf{x}.$$

We can apply the Taylor expansion of $\ln p(\mathbf{x}, \Theta)$ with respect to $\mathbf{x}$ to the function at $\mathbf{x} = \mathbf{x}_i$ and only keep the second order term. Replacing the integral by an average over the data $\mathbf{x}_i$, it results in the approximation of J_0,

$$J_0(h) \approx J_{01}(\Theta) + hJ_r(\Theta) \tag{A.2}$$

with

$$J_{01}(\Theta) = -\frac{1}{N}\sum_{i=1}^{N} \ln \sum_{j=1}^{k} \alpha_j G(\mathbf{x}_i, \mathbf{m}_j, \mathbf{\Sigma}_j),$$

$$J_r(\Theta) = -\frac{1}{2N}\sum_{i=1}^{N} \mathrm{Trace}[\nabla_x \nabla_x \ln p(\mathbf{x}_i, \Theta)]. \tag{A.3}$$

The logarithmic mixture density Hessian matrix can be computed as

$$\begin{aligned}\nabla_x \nabla_x \ln p(\mathbf{x}_i, \Theta) =& -\left\{\sum_{j=1}^{k} P(j|\mathbf{x}_i, \Theta)\{\mathbf{\Sigma}_j^{-1} - \mathbf{\Sigma}_j^{-1}(\mathbf{x}_i - \mathbf{m}_j)(\mathbf{x}_i - \mathbf{m}_j)^{\mathrm{T}}\mathbf{\Sigma}_j^{-1}\}\right. \\ &+ \left\{\sum_{j=1}^{k} P(j|\mathbf{x}_i, \Theta)[(\mathbf{x}_i - \mathbf{m}_j)^{\mathrm{T}}\mathbf{\Sigma}_j^{-1}]^{\mathrm{T}}\right\} \\ &\left.\times \left\{\sum_{j=1}^{k} P(j|\mathbf{x}_i)[(\mathbf{x}_i - \mathbf{m}_j)^{\mathrm{T}}\mathbf{\Sigma}_j^{-1}]\right\}\right\}. \end{aligned} \tag{A.4}$$

Similar to J_0, we can obtain the $J_e(h)$ term as follows:

$$J_e(h) \approx \frac{1}{N}\sum_{i=1}^{N} R(\mathbf{x}_i, h) + \frac{h}{2N}\sum_{i=1}^{N} \mathrm{Trace} \times [\nabla_x \nabla_x R(\mathbf{x}_i, h)], \tag{A.5}$$

where

$$R(\mathbf{x}_i, h) \equiv \ln p_h(\mathbf{x}_i).$$

$$\begin{aligned}\nabla_x \nabla_x R(\mathbf{x}_i, h) =& \frac{1}{h^2}\left\{\sum_{j=1}^{N} \beta(\mathbf{x}_i, \mathbf{x}_j)[(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^{\mathrm{T}} - h\mathbf{I}_d]\right. \\ &- \left[\sum_{j=1}^{N} \beta(\mathbf{x}_i, \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)\right] \\ &\left.\times \left[\sum_{j=1}^{N} \beta(\mathbf{x}_i, \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)\right]^{\mathrm{T}}\right\}\end{aligned}$$

with

$$\beta(\mathbf{x}, \mathbf{x}_i) = \frac{G(\mathbf{x}, \mathbf{x}_i, h\mathbf{I}_d)}{\sum_{j=1}^{N} G(\mathbf{x}, \mathbf{x}_j, h\mathbf{I}_d)}$$

and note that

$$\sum_{i=1}^{N} \beta(\mathbf{x}, \mathbf{x}_i) = \sum_{i=1}^{N} \frac{G(\mathbf{x}, \mathbf{x}_i, h\mathbf{I}_d)}{\sum_{j=1}^{N} G(\mathbf{x}, \mathbf{x}_j, h\mathbf{I}_d)} = 1.$$

With this relation, we get

$$\frac{h}{2N}\sum_{i=1}^{N}\mathrm{Trace}[\nabla_x\nabla_x R(\mathbf{x}_i,h)]$$
$$=\frac{1}{2Nh}\sum_{i=1}^{N}\left\{\sum_{j=1}^{N}\beta(\mathbf{x}_i,\mathbf{x}_j)[\|\mathbf{x}_i-\mathbf{x}_j\|^2]-\left\|\sum_{j=1}^{N}\beta(\mathbf{x}_i,\mathbf{x}_j)(\mathbf{x}_i-\mathbf{x}_j)\right\|^2\right\}-\frac{d}{2}.$$

Now the function $J(h)$ can be computed based on the original samples with summation instead of integration. That is,

$$J(h)=J_0(h)+J_e(h)\approx J_{01}(\Theta)+hJ_r(\Theta)+J_e(h). \quad (A.6)$$

For very sparse data distribution, we can use the following approximation:

$$p_h(\mathbf{x})\ln p_h(\mathbf{x})\approx\frac{1}{N}\sum_{i=1}^{N}G(\mathbf{x},\mathbf{x}_i,h\mathbf{I}_d)\ln\frac{1}{N}G(\mathbf{x},\mathbf{x}_i,h\mathbf{I}_d)$$
$$=\frac{1}{N}\sum_{i=1}^{N}G(\mathbf{x},\mathbf{x}_i,h\mathbf{I}_d)\left[-\frac{d}{2}\ln(2\pi h)-\frac{1}{2h}\|\mathbf{x}-\mathbf{x}_i\|^2-\ln N\right]$$

$$J_e(h)=\int p_h(\mathbf{x})\ln p_h(\mathbf{x})\,\mathrm{d}\mathbf{x}\approx-\frac{d}{2}\ln(2\pi h)-\frac{d}{2}-\ln N.$$

In this case we get the approximation formula for $J(h)$,

$$J(h)\approx J_{01}(\Theta)+hJ_r(\Theta)-\frac{d}{2}\ln(h)+C, \quad (A.7)$$

where C is a constant irrelevant to h.

Taking partial derivative of $J(h)$ to h and let it be equal to zero,

$$\frac{\partial}{\partial h}J(h)=J_r(\Theta)-\frac{d}{2h}=0$$

the rough estimation formula is then obtained as

$$h=\frac{d}{2J_r(\Theta)}. \quad (A.8)$$

References

[1] S. Aeberhard, D. Coomans, O. de Vel, Comparative analysis of statistical pattern recognition methods in high dimensional settings, Pattern Recognition 27 (8) (1994) 1065–1077.

[2] G.J. McLachlan, K.E. Basford, Mixture Models: Inference and Applications to Clustering, Dekker, New York, 1988.

[3] B.S. Everitt, D. Hand, Finite Mixture Distributions, Chapman and Hall, London, 1981.

[4] N.E. Day, Estimating the component of a mixture of normal distributions, Biometrika 56 (1969) 463–474.

[5] H.H. Bock, Probability models and hypotheses testing in partitioning cluster analysis, in: Clustering and Classification, World Scientific Press, Riverside, CA, 1996, pp. 377–453.

[6] E.K. Tang, P.N. Suganthan, X. Yao, A.K. Qin, Linear dimensionality reduction using relevance weighted LDA, Pattern Recognition 38 (4) (2005) 485–493.

[7] J. Lu, K.N. Plataniotis, A.N. Venetsanopoulos, Face recognition using LDA-based algorithms, IEEE Trans. Neural Networks 14 (1) (2003) 195–200.

[8] J. Lu, K.N. Plataniotis, A.N. Venetsanopoulos, Regularization studies of linear discriminant analysis in small sample size scenarios with application to fance recognition, Pattern Recognition Lett. 26 (2) (2005) 181–191.

[9] H. Liu, J. Lafferty, L. Wasserman, Sparse nonparametric density estimation in high-dimension using the Rodeo, in: M. Meila, X. Shen (Eds.), Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics, San Juan, Puerto Rico, March 21–24, 2007.

[10] J.H. Friedman, Regularized discriminant analysis, J. Am. Stat. Assoc. 84 (1989) 165–175.

[11] J.P. Hoffbeck, D.A. Landgrebe, Covariance matrix estimation and classification with limited training data, IEEE Trans. Pattern Anal. Mach. Intell. 18 (7) (1996) 763–767.

[12] S. Tadjudin, D.A. Landgrebe, Covariance estimation with limited training samples, IEEE Trans. Geosci. Remote Sensing 37 (4) (1999) 2113–2118.

[13] B.-C. Kuo, D.A. Landgrebe, A covariance estimator for small sample size classification problems and its application to feature extraction, IEEE Trans. Geosci. Remote Sensing 40 (4) (2002) 814–819.

[14] Q. Jackson, D.A. Landgrebe, An adaptive method for combined covariance estimation and classification, IEEE Trans. Geosci. Remote Sensing 40 (5) (2002) 1082–1087.

[15] I. Rivals, L. Personnaz, On cross validation for model selection, Neural Comput. 11 (1999) 863–870.

[16] J. Rissanen, Modeling by shortest data description, Automatica 14 (1978) 465–471.

[17] A. Barron, J. Rissanen, B. Yu, The minimum description length principle in coding and modeling, IEEE Trans. Inform. Theory 44 (6) (1998) 2743–2760.

[18] S. Kullback, Information Theory and Statistics, Wiley, New York, 1959.

[19] L. Devroye, A Course in Density Estimation, Birkhauser Publisher, Boston, 1987.

[20] P. Kontkanen, P. Myllymki, W. Buntine, J. Rissanen, H. Tirri, An MDL framework for data clustering, in: P. Grnwald, I.J. Myung, M. Pitt (Eds.), Advances in Minimum Description Length: Theory and Applications, MIT Press, Cambridge, MA, 2004.

[21] A.R. Webb, Statistical Pattern Recognition, Oxford University Press, London, 1999.

[22] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum-likelihood from incomplete data via the EM algorithm, J. R. Stat. Soc. B 39 (1977) 1–38.

[23] R.A. Redner, H.F. Walker, Mixture densities, maximum likelihood and the EM algorithm, SIAM Rev. 26 (1984) 195–239.

[24] P. Guo, M.R. Lyu, Classification for high-dimension small-sample data sets based on Kullback–Leibler information measure, in: H.R. Arabnia (Ed.), Proceedings of the 2000 International Conference on Artificial Intelligence, vol. III, Monte Carlo Resort, Las Vegas, Nevada, USA, CSREA Press, 2000, pp. 1187–1193.

[25] H. Bensmail, G. Celeux, Regularized Gaussian discriminant analysis through eigenvalue decomposition, J. Am. Stat. Assoc. 91 (1996) 1743–1748.

[26] J. Lu, K.N. Plataniotis, A.N. Venetsanopoulos, S.Z. Li, Ensemble-based discriminant learning with boosting for face recognition, IEEE Trans. Neural Networks 17 (1) (2006) 166–178.

[27] S. Srivastava, M.R. Gupta, B.A. Frigyik, Bayesian quadratic discriminant analysis, J. Mach. Learning Res. 8 (2007) 1277–1305.

[28] K. Fukunaga, Introduction to Statistical Pattern Recognition, second ed., Academic Press, Boston, 1990.

[29] P. Guo, M.R. Lyu, A study on color space selection for determining image segmentation region number, in: H.R. Arabnia (Ed.), Proceedings of the 2000 International Conference on Artificial Intelligence, Monte Carlo Resort, Las Vegas, Nevada, USA, CSREA Press, 2000, pp. 1127–1132.

[30] G.S. Fishman, Monte Carlo: Concepts, Algorithms, and Applications, Springer, New York, 1996.
[31] J.E. Gentle, Random Number Generation and Monte Carlo Methods, Springer, New York, 1998.
[32] C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, Oxford, 1995.
[33] P. Guo, M.R. Lyu, C.L.P. Chen, Regularization parameter estimation for feedforward neural networks, IEEE Trans. Syst. Man Cybern. Part B 33 (1) (2003) 35–44.
[34] F. Liang, A. Barron, Exact minimax strategies for predictive density estimation, data compression, and model selection, IEEE Trans. Inf. Theory 50 (11) (2004) 2708–2726.
[35] P. Guo, Spectral pattern recognition with regularized Gaussian classifier, in: Proceedings of the 2003 International Conference on Neural Networks and Signal Processing, vol. 1, 2003, pp. 727–730.
[36] B. Efron, R. Tibshirani, An Introduction to the Bootstrap, Chapman and Hall, London, 1993.
[37] J.R. Price, T.F. Gee, Face recognition using direct, weighted linear discriminant analysis and modular subspaces, Pattern Recognition 38 (2) (2005) 209–219.
[38] I.T. Jolliffe, Principal Component Analysis, Springer, New York, 1986.
[39] Y. Jiang, P. Guo, Regularization versus dimension reduction, which is better? in: Lecture Notes in Computer Science, vol. 4492, part II, 2007, pp. 474–482.
[40] J. Lu, K.N. Plataniotis, A.N. Venetsanopoulos, J. Wang, An efficient kernel discriminant analysis method, Pattern Recognition 38 (10) (2005) 1788–1790.
[41] J. Lu, K.N. Plataniotis, A.N. Venetsanopoulos, Face recognition using kernel direct discriminant analysis algorithm, IEEE Trans. Neural Networks 14 (1) (2003) 117–126.
[42] Y. Jiang, P. Guo, Comparative studies of feature extraction methods with application to face recognition, in: Proceedings of 2007 IEEE International Conference on System, Man and Cybernetics, 2007, pp. 3627–3632.
[43] ORL face dataset ⟨http://www.cl.cam.ac.uk/research/dtg/attarchive/facedata-base.html⟩.
[44] X. Wang, F. Xing, P. Guo, Comparison of discriminant analysis methods applied to stellar data classification, in: Proceedings of SPIE, vol. 5286, part II, 2003, pp. 758–763.
[45] P. Guo, C.L.P. Chen, M.R. Lyu, Cluster number selection for a small set of samples using the Bayesian ying-yang Model, IEEE Trans. Neural Network 13 (3) (2002) 757–763.

About the Author—PING GUO is currently a professor at the School of Computer Science and Technology in Beijing Institute of Technology, and the Computer Science Department of the Beijing Normal University. From 1993 to 1994 he was with the Department of Computer Science and Engineering at the Wright State University as a visiting faculty. From 2000 May to 2000 August he was with the National Laboratory of Pattern Recognition at Chinese Academy of Sciences as a guest researcher. He is the author or co-author of more than 160 papers. His current research interests include neural network, pattern recognition, image process, software reliability engineering, optical computing, and spectra analysis.
Dr. Guo received his M.S. degree in Physics from Peking University, his Ph.D. degree in Computer Science from The Chinese University of Hong Kong.

About the Author—YUNDE JIA is a professor in the department of computer science and engineering, currently a vice dean of the school of computer science at the Beijing Institute of Technology. He earned the bachelor's, master's and Ph.D. degrees in mechatronics from the Beijing Institute of Technology in 1983, 1986 and 2000, respectively. He found the computer vision lab in 1997 at the university right after working as a visiting researcher in the Carnegie Mellon University. His research interests include computer vision, media computing, and intelligent human–computer interaction.

About the Author—MICHAEL R. LYU is currently a professor in the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong. His research interests include software reliability engineering, distributed systems, fault-tolerant computing, wireless communication networks, Web technologies, digital libraries, and E-commerce systems. He has published around 300 refereed journal and conference papers in these areas. He served on the Editorial Boards of IEEE Transactions on Knowledge and Data Engineering, IEEE Transactions on Reliability, and Journal of Information Science and Engineering. Dr. Lyu is a fellow of IEEE and a fellow of AAAS.
Dr. Lyu received his B.S. degree in Electrical Engineering from National Taiwan University, Taipei, Taiwan, in 1981, his M.S. degree in Computer Engineering from University of California, Santa Barbara, in 1985, and his Ph.D. degree in Computer Science from University of California, Los Angeles, in 1988.