Data Mapping by Probabilistic Modular Networks and Information-Theoretic Criteria

Yue Wang, Shang-Hung Lin, Huai Li, and Sun-Yuan Kung, *Fellow, IEEE*

***Abstract*—The quantitative mapping of a database that represents a finite set of *classified* and/or *unclassified* data points may be decomposed into three distinctive learning tasks:**

1) **detection of the structure of each class model with locally mixture clusters;**
2) **estimation of the data distributions for each induced cluster inside each class;**
3) **classification of the data into classes that realizes the data memberships.**

The mapping function accomplished by the probabilistic modular networks may then be constructed as the optimal estimator with respect to information theory, and each of the three tasks can be interpreted as an independent objective in real-world applications. We adapt a model fitting scheme that determines both the number and kernel of local clusters using information-theoretic criteria. The class distribution functions are then obtained by learning generalized Gaussian mixtures, where a *soft* classification of the data is performed by an efficient incremental algorithm. Further classification of the data is treated as a *hard* Bayesian detection problem, in particular, the decision boundaries between the classes are fine tuned by a reinforce or antireinforce supervised learning scheme. Examples of the application of this framework to medical image quantification, automated face recognition, and featured database analysis, are presented as well.

I. Introduction

THIS PAPER addresses the problem of mapping a database, given a finite set of data points (examples). The mapping function can therefore be interpreted as a quantitative representation of the contents (knowledge) contained in the database [1], [3], [4]. The data set may be a *classified* set, as in general clustering problems [2], [22], [25], it may be unclassified, as in unsupervised distribution learning [1], [12], [18], or it may be a partially classified set, as in pattern classification applications [5]–[7]. Instead of mapping the whole data set using a single complex network, in many applications, it is more practical to design a set of simple class subnets with locally mixture clusters, each one of which represents a specific region of the knowledge space. This is indeed the case, and in particular, inspired by the principle of divide-and-conquer in applied statistics, probabilistic modular neural networks have become increasingly popular in the machine learning research [1], [4]–[7], [17], [36]. In this paper, we present a particular application of the probabilistic modular networks to the problem of mapping from databases. We describe a constructive criterion for designing the network architecture and the learning algorithm, both of which are governed by information theory [37]. The motivation of this work comes from following considerations. First, the database (available knowledge) and the network (learning capability) have been traditionally treated as two separate components in neural system design, where the relationship between them is not explicit [36]. It is desirable to have a network mapping a database, thus allowing an efficient information representation [25]. Second, since the complex cluster patterns and distributions intrinsically exhibited in a database are generally not transparent to the user, it will be difficult to interpret the output of system, to analyze the course of error, and to evaluate the process of performance [4]. A high-resolution divide-and-conquer architecture, i.e., hierarchy, may be required. Finally, in many practical applications, data mapping means either supervised (with objective of data classification) [2], unsupervised (with objective of data quantification) [12], [22], or the combined learning [5]. A flexible but unified scheme should be explored.

The quantitative mapping of a database may be decomposed into three distinctive learning tasks:

1) detection of the structure of each class model with locally mixture clusters;
2) estimation of the data distributions for each induced cluster inside each class;
3) classification of the data into classes that realizes the data memberships.

Although many previously proposed approaches have led to quite impressive results, several fundamental issues remain unresolved in the application domain. For example, the finite mixture model has very appealing properties to class distribution learning; the number of local clusters and the kernel shapes of cluster distributions are often assumed to be known, which is far from being realized in most applications [2], [9], [13], [17], [22]. The data mapping will be, in general, difficult to interpret since imposing a simple parametric model for the class may prevent the correct identification of the data structure [25] and the accurate estimation of the class boundaries [1], [26]. If the local models are to map the structure of the class

Manuscript received July 24, 1997; revised January 5, 1998. This work was supported in part by the U.S. Army Medical Research and Materiel Command under Grant DAMD17-98-1-8046 and the National Institutes of Health under Grant 1R21RR12784-01. The associate editor coordinating the review of this paper and approving it for publication was Dr. Y. H. Hu.

Y. Wang is with the Department of Electrical Engineering and Computer Science, The Catholic University of America, Washington, DC 20064 USA (e-mail: wang@pluto.ee.cua.edu).

S.-H. Lin is with the Epson Palo Alto Laboratory, Palo Alto, CA 94304 USA (e-mail: shlin@epal.com).

H. Li is with the Department of Electrical Engineering, University of Maryland, College Park, MD 20742 USA (e-mail: huaili@eng.umd.edu).

S.-Y. Kung is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA (e-mail: kung@ee.princeton.edu).

Publisher Item Identifier S 1053-587X(98)08813-8.

and the class boundaries, model selection must be taken into consideration on the goodness of fit [4], [7]. Furthermore, once the correct model is determined, we may formulate parameter learning as problem of maximum likelihood (ML) estimation [1], [2], [10]. The most popular algorithm in this domain is expectation–maximization (EM) algorithm [3], [19]. However, the EM algorithm has the reputation of being a slow algorithm since its batch training has a first-order convergence in which new information acquired in the expectation step is not used immediately [19], [21], [22]. In order to balance the tradeoff between efficiency and accuracy, on-line algorithms are proposed for large-scale sequential learning [3], [11] and are extended to supervised learning [6], [17]. The price to be paid is then a greatly increased memory requirements [20]. In addition, since data quantification (inside each class) and data classification (between the classes) may be the two independent objectives in applications, the optimality criteria for them are indeed different. However, the relationship between these two objectives, as well as how the error interferes each other, have not been fully understood [23], [26]. Moreover, empirical results indicate that many neural network classifiers, whose structure and learning rule were designed to directly approximate the class posterior probabilities, may be unnecessarily complex since the coupled training scheme has to adapt and update simultaneously both the class likelihood and the class prior probabilities [6], [25], [39].

The objective of this work is to propose a unified learning strategy for mapping a database: The main idea is to find, in a first place, a set of local mixture models that efficiently represent the data, together with a model selection procedure in which the optimal number and shape of the local clusters are found by the information-theoretic criteria. A partition of the data set into classes that indicate the membership of each data point may then be realized in a second phase, where the decision boundaries will be determined according to a supervised error-correction training. The major differences between our work and the previous work [1], [9], [15], [17], [20], [22], [25] are as follows.

1) We impose a model selection procedure to determine both the number and kernel shape of local clusters inside each class using information-theoretic criteria. This allows us to analyze how the result in model selection affects the performances of both data quantification and classification.
2) We apply a fully adaptive incremental algorithm to the unsupervised learning of the class distribution functions. It involves a *soft* classification of the data under the principle of least relative entropy, thus leading to an efficient and unbiased estimation.
3) We add a fine-tuning phase for learning decision likelihood boundaries using a reinforce or antireinforce supervision approach in which the class prior is adjusted in a separate phase.

This decoupled training scheme permits the use of high-capacity classifiers while maintaining a reasonable computational complexity for the further classification of the data into the classes. In addition, we have analyzed the pairwise relationships between quantification and classification, between *soft* and *hard* classification, and between unsupervised and supervised learning. The insights provide the guidance for the correct use of various methods in real-world applications.

The remainder of the paper proceeds as follows. Section II presents the problem formulation regarding the statistical modeling, unsupervised data quantification, and supervised data classification. This is followed by detailed description of the methods and algorithms that, in practice, appears to be the most complete of the approaches that we have studied. In Section III, three application examples in different domains are presented that illustrate the performance of the proposed techniques in various aspects. Major conclusions and discussions are summarized in the final section.

II. Methods and Algorithms

A. Statistical Modeling

Recently, there has been considerable success in using finite mixture distributions and probabilistic modular networks for data quantification and classification [1], [3], [10], [17], [18], [34]. In order to validate the suitable stochastic models for data mapping with specified objectives, over the past few years, we have conducted an investigation into data statistics and derived several useful theorems [4], [12]. Assume that the data points x_i in a database come from M classes $\{\omega_1, \cdots, \omega_r, \cdots, \omega_M\}$, and each class contains K_r clusters $\{\theta_1, \cdots, \theta_k, \cdots, \theta_{K_r}\}$, where ω_r is the model parameter vector of class r, and θ_k is the kernel parameter vector of cluster k within class r. Further assume that in our training data set (which should be a representative subset of the whole database), each data point has a one-to-one correspondence to one of the classes denoted by its class label l_{ir}^*, defining a supervised learning task, but the true memberships of the data to the local clusters are unknown, defining an unsupervised learning task.

For the model of local class distribution, since the true cluster membership for each data point is unknown, we can treat cluster labels of the data as random variables denoted by l_{ik} [23]. By introducing a probability measure of a multinomial distribution with an unknown parameter π_k to reflect the distribution of the number of data points in each cluster, the relevant (sufficient) statistics are the conditional statistics for each cluster and the number of data points in each cluster. The class conditional probability measure for any data point inside the class r, i.e., the standard finite mixture distribution (SFMD), can be obtained by writing down the joint probability density of the x_i and l_{ik} and then, summing it over all possible outcomes of l_{ik}, as a sum of the general form

$$f(u|\omega_r) = \sum_{k=1}^{K_r} \pi_k g(u|\theta_k) \tag{1}$$

where $\pi_k = P(\theta_k|\omega_r)$ with a summation equal to one, and $g(u|\theta_k)$ is the kernel function of the local cluster distribution. Several observations are worth reiteration.

1) All data points in a class are identically distributed from a mixture distribution.

2) The SFMD model uses the probability measure of data memberships to the clusters in the formulation instead of realizing the true cluster label for each data point.
3) Since the calculation of the data histogram $f_{\mathbf{x}_r}$ from a class relies on the same mechanism as in (1), its values can be considered to be a sampled version of the true class distribution f_r^*.

For the model of global class distributions, we denote the Bayesian prior for each class by $P(\omega_r)$. Then, the sufficient statistics for mapping a database, i.e., the conditional finite mixture distribution (CFMD), is the pair of $\{P(\omega_r),\ f(u|\omega_r)\}$. According to the Bayes' rule, the posterior probability $P(\omega_r|x_i)$ given a particular observation x_i can be obtained by

$$P(\omega_r|x_i) = \frac{P(\omega_r)f(x_i|\omega_r)}{p(x_i)} \tag{2}$$

where $p(x_i) = \sum_{r=1}^{M} P(\omega_r)f(x_i|\omega_r)$. Again, several observations are worth reiteration:

1) In order to classify the data points into classes, (2) is a candidate as a discriminant function.
2) Since defining a supervised learning requires information of l_{ir}^*, the Bayesian prior $P(\omega_r)$ is an intrinsically known parameter and can be easily estimated by $P(\omega_r) = \sum_{i=1}^{N} l_{ir}^*/N$.
3) The only uncertainty comes from class likelihood function $f(u|\omega_r)$ that should be the key issue in the follow-on learning process.

For simplicity, in the following context we will omit class index r in our discussion when only single class distribution model is concerned and use θ to denote the parameter vector of regional parameter set $\{(\pi_k, \theta_k)\}$.

B. Data Quantification via Unsupervised Learning

The problem of data quantification addresses the combined estimation of regional parameters (π_k, θ_k) and detection of the structural parameter K_r and the kernel shape of $g(\cdot)$ in (1) based on the observations $\mathbf{x}_r$. One natural criterion used for learning the optimal parameter values is to minimize the distance between the SFMD, which is denoted by $f_r(u)$, and the class data histogram, which is denoted by $f_{\mathbf{x}_r}(u)$ [3]. In this work, we use relative entropy (Kullback–Leibler distance), which was suggested by information theory [37], as the distance measure [for simplicity, we use $f_r(u)$ to denote $f(u|\omega_r)$ in our formulation] given by

$$D(f_{\mathbf{x}_r}||f_r) = \sum_{u} f_{\mathbf{x}_r}(u)\ \log\ \frac{f_{\mathbf{x}_r}(u)}{f(u|\omega_r)}. \tag{3}$$

Note that the new cost function overcomes the problems of using squared error by weighting errors more heavily when probabilities are near zero and one and diverging in the case of convergence at the wrong extreme [2], [11]. Furthermore, we have previously shown that when relative entropy is used as a distance measure, the distance minimization method is equivalent to the soft-split classification-based method under the criterion of maximum likelihood (ML) [12], [32]. The conclusion is summarized by the following theorem (see the proof in the Appendix):

Theorem 1: Consider a sequence of random variables $x_1, \cdots, x_{N_r}$ in $\mathcal{R}^{N_r}$. Assume that the sequence $\{x_i\}$ is independent and identically distributed (i.i.d.) by the distribution f_r. Then, the joint likelihood function $\mathcal{L}_r(\theta)$ is determined only by the histogram of data $f_{\mathbf{x}_r}$ and is given by

$$\mathcal{L}_r(\theta) = \exp\left(-N_r[H(f_{\mathbf{x}_r}) + D(f_{\mathbf{x}_r}||f_r)]\right) \tag{4}$$

where H denotes the entropy with base e, and the maximization of joint likelihood function $\mathcal{L}_r(\theta)$ is equivalent to the minimization of relative entropy $D(f_{\mathbf{x}_r}||f_r)$.

Thus, data quantification is formulated as a distribution learning problem, and the actual optimality is achieved when this cost function reaches its minimum. However, statistical dependence between data points is one of some fundamental concerns in the problem formulation since the calculation of the data histogram assumes that all the data points are independent random variables. In order to validate the correct use of the (3) in data quantification, we prove the following theorem to show that the data histogram $f_{\mathbf{x}_r}(u)$ converges to the true distribution $f_r^*(u)$ for all u with probability one as $N_r \to \infty$. Thus, when N_r is sufficiently large, minimization of the relative entropy between f_r and f_r^* can be well approximated by the minimization of the relative entropy between $f_{\mathbf{x}_r}$ and f_r. This fitting procedure can be practically implemented by maximizing the joint likelihood function under the independence approximation of the data (see proof in Appendix) [4].

Theorem 2: Consider a sequence of random variables $x_1, \cdots, x_{N_r}$ in $\mathcal{R}^{N_r}$. Assume that the sequence $\{x_i\}$ is asymptotically independent [40] and identically distributed by the finite normal mixture distribution f_r^*. For a closed convex set $\mathrm{E} \subset \mathcal{F}_r$ and distribution $f_{\mathbf{x}_r} \notin \mathrm{E}$, let $f_r \in \mathrm{E}$ be the distribution that achieves the minimum distance to $f_{\mathbf{x}_r}$, i.e.,

$$f_r = \arg\ \min_{\mathcal{F}_r \in \mathrm{E}} D(f_{\mathbf{x}_r}||\mathcal{F}_r). \tag{5}$$

Then, when N_r approaches infinity, we have

$$\lim_{N_r \to \infty} D(f_r||f_r^*) = 0 \tag{6}$$

with probability one, i.e., the estimated distribution of $\mathbf{x}_r$, given that f_r achieves the minimum of $D(f_{\mathbf{x}_r}||f_r)$ is close to f_r^* for large N_r.

Another important issue concerning unsupervised distribution learning is the detection of the structural parameters of the class distribution known as model selection [1]. The objective here is to propose a systematic strategy for determining the optimal number and kernel shape of local clusters when the prior knowledge is not available. The motivations are driven by various objectives and requirements in the real applications. For example, the prior knowledge on the true structure of a database is generally unknown, i.e., the number and the kernel shape of the local clusters are not available beforehand, and model selection is required in the data mapping procedure. This is indeed the case that is particularly critical in real clinical applications, where the structure of the disease patterns for a particular patient or for a particular type of cancer may be arbitrarily complex; therefore, correct identification and quantification of the information is very important [4],

[7]. Thus, it will be desirable to have a neural network structure that is adaptive in the sense that the number and kernel shape of local clusters are not fixed beforehand. One conventional approach for doing this is to use a sequence of hypothesis tests [3], [36]. The problem in this approach, however, is the subjective judgment in the selection of the threshold for different tests. Recently, there has been a great deal of interest in using information theoretic criteria, such as Akaike information criterion (AIC) [27], [34] and minimum description length (MDL) [28], [30], to solve this problem. The major thrust of this approach has been the formulation of a model fitting procedure in which an optimal model is selected from the several competing candidates such that the selected model best fits the observed data. For example, AIC will select the model that gives the minimum defined by

$$\mathrm{AIC}(K_a) = -2\,\log\left(\mathcal{L}(\hat{\theta}_{\mathrm{ML}})\right) + 2K_a \tag{7}$$

where $\mathcal{L}(\hat{\theta}_{\mathrm{ML}})$ is the likelihood of $\hat{\theta}_{\mathrm{ML}}$, and K_a is the number of free adjustable parameters in the model. From a quite different point of view, MDL reformulates the problem explicitly as an information coding problem in which the best model fit is measured such that it assigns high probabilities to the observed data, while at the same time, the model itself is not too complex to describe [28]. A model is selected by minimizing the total description length defined by

$$\mathrm{MDL}(K_a) = -\log\left(\mathcal{L}(\hat{\theta}_{\mathrm{ML}})\right) + 0.5K_a\,\log N_r. \tag{8}$$

Note that, different from AIC, the penalty term in MDL takes into account the number of observations. However, the justifications for the optimality of these two criteria with respect to data quantification or classification are somewhat indirect and remain unresolved [3], [27], [32], and none of these approaches have directly addressed the problem of kernel shape learning [7].

In this work, we derive a new formulation of the information theoretic criterion [the minimum conditional bias/variance (MCBV) criterion] to solve model selection problem. Nevertheless, it was Akaike/Rissanen's work that was the inspirational source to this work, but some new interpretations are presented and justified with the information-theoretic means [32]. Our approach has a simple optimal appeal in that it selects a minimum conditional bias and variance model, i.e., if two models are about equally likely, MCBV selects the one whose parameters can be estimated with the smallest variance.

The new formulation is based on the fundamental argument that the value of the structural parameter can not be arbitrary or infinite because such an estimate might be said to have low "bias," but the price to be paid is high "variance" [31]. From Jaynes' principle, which is stated as "*the parameters in a model which determine the value of the maximum entropy should be assigned values which minimize the maximum entropy*" [29], let joint entropy of $\mathbf{x}$ and $\hat{\theta}$ be $H(\mathbf{x}, \hat{\theta}) = H(\mathbf{x}|\hat{\theta}) + H(\hat{\theta})$, following the Bayes' law, a very neat interpretation states that the maximum of conditional entropy $H(\mathbf{x}|\hat{\theta})$ is precisely the negative of the logarithm of the likelihood function $\mathcal{L}(\mathbf{x}|\hat{\theta})$ corresponding to the entropy-maximizing distribution of $\mathbf{x}$ [28], [30]. Thus, we have

$$\max_{P_{\mathbf{x}}} H(\mathbf{x}|\hat{\theta}) = -\log\left(\mathcal{L}(\mathbf{x}|\hat{\theta})\right)\Big|_{P_{\mathbf{x}}=\prod_{i=1}^{N_r} f_r(x_i)}. \tag{9}$$

Note that the uniformly randomization in the SFMD modeling corresponds to the maximum uncertainty [23], [37]. Furthermore, maximizing the entropy of the parameter estimates $H(\hat{\theta})$ results in

$$\max_{P_{\hat{\theta}}} H(\hat{\theta}) = \sum_{k=1}^{K_a} H(\hat{\theta}_k) \tag{10}$$

where when the variance of the parameter estimate is determined by the corresponding sample estimate, normal and independent distribution $P_{\hat{\theta}}$ gives the maximum entropy [37], [38].

Since the joint maximum entropy is a function of K_a and $\hat{\theta}$, by taking the advantage of the fact that model estimation is separable in components and structure, we define the MCBV criterion as

$$\mathrm{MCBV}(K) = -\log\left(\mathcal{L}(\mathbf{x}|\hat{\theta}_{\mathrm{ML}})\right) + \sum_{k=1}^{K_a} H(\hat{\theta}_{k\mathrm{ML}}) \tag{11}$$

where $-\log\left(\mathcal{L}(\mathbf{x}|\hat{\theta})\right)$ is the conditional bias, and $\sum_{k=1}^{K_a} H(\hat{\theta}_k)$ is the conditional variance of the model. As both terms represent natural estimation errors about their true models and should be treated on an equal basis, a minimization leads to the characterization of the optimum estimation as

$$K_0 = \arg\left\{\min_{1\le K\le K_{\mathrm{MAX}}} \mathrm{MCBV}(K)\right\}. \tag{12}$$

That is, if the cost of model variance is defined as the entropy of parameter estimates, the cost of adding new parameters to the model must be balanced by the reduction they permit in the ideal code length for the reconstruction error. A practical MCBV formulation with code-length expression is further given by

$$\begin{aligned}\mathrm{MCBV}(K) = &-\log\left(\mathcal{L}(\mathbf{x}|\hat{\theta}_{\mathrm{ML}})\right)\\ &+\sum_{k=1}^{K_a} \tfrac{1}{2}\,\log\,2\pi e\mathrm{Var}(\hat{\theta}_{k\mathrm{ML}}).\end{aligned} \tag{13}$$

However, the calculation of $H(\hat{\theta}_{k\mathrm{ML}})$ requires the true values of the model parameters that are to be estimated. It has been shown that if the number of observations exceeds the minimal value, the accuracy of the ML estimation tends quickly to the best possible accuracy determined by the Cramér–Rao lower bounds (CRLB's), as has been well studied theoretically in [1] and [38]. Thus, the CRLB's of the parameter estimates are used in the actual calculation representing the "conditional" bias and variance [33]. We have found that the new formulation for determining the value of K_0 exhibits a very good experimental performance that is consistent with both

AIC and MDL. It should be noted, however, that it is not the only plausible one; other criteria, such as cross validation techniques, may also be useful in this case.

The performance of model selection for two frequently used methods, i.e., the AIC and MDL, and the proposed criterion (MCBV) were first tested and compared in the simulation study. The computer-generated data was made up of four overlapping normal components. Each component represents one local cluster. The value for each component were set to a constant value, and the noise of normal distribution was then added to this simulation digital phantom. Three noise levels with different variance were set to keep the same signal-to-noise ratio (SNR), where SNR is defined as $10\log_{10}(\Delta\mu)^2/\sigma^2$, with $\Delta\mu$ being the mean difference between clusters, and σ^2 is the noise power. The original data for the simulation study are given in Fig. 1(a). The AIC, MDL, and MCBV curves, as functions of the number of local clusters K, are plotted in the same figure. According to the information-theoretic criteria, the minima of these curves indicate the correct number of the local cluster. From this experimental figure, it is clear that the number of local clusters suggested by these criteria are all correct. For larger noise level, the model selection based on the MCBV criterion provides a more differentiable result than the other two criteria. More application of the MCBV to the identification of real data structures will be presented in the next section.

As the counterpart for adaptive model selection, there are many numerical techniques to perform ML estimation of cluster parameters [3]. For example, EM algorithm first calculates the posterior Bayesian probabilities of the data through the observations and the current parameter estimates (E-step) and then updates parameter estimates using generalized mean ergodic theorems (M-step). The procedure cycles back and forth between these two steps. The successive iterations increase the likelihood of the model parameters. In order to obviate the need to store all the incoming observations and change the parameters immediately after each data point allowing for high data rates, we developed a probabilistic self-organizing mixture (PSOM) algorithm to solve the problem. This is a fully incremental and stochastic learning algorithm and is a generalized adaptive version of the similar algorithm we presented in [12]. The scheme provides winner-takes-in probability (Bayesian "soft") splits of the data, hence, allowing the data to contribute simultaneously to multiple clusters. For the sake of simplicity, we assume the kernel shape of local cluster to be a Gaussian with mean μ_k and variance σ_k^2 in the following derivation. By differentiating $D(f_{\mathbf{x}_r}||f_r)$ given in (3) (here, the index of cluster r is omitted) with respect to the unconstrained parameters μ_k and σ_k^2, we obtain the standard gradient descent learning rule for the mean and variance parameter vectors ($k=1,\cdots,K$)

$$\mu_k^{(t+1)} = \mu_k^{(t)} + \frac{\lambda}{N}\sum_{i=1}^{N}(x_i-\mu_k^{(t)})\frac{z_{ik}^{(t)}}{\sigma_k^{2(t)}} \tag{14}$$

$$\sigma_k^{2(t+1)} = \sigma_k^{2(t)} + \frac{\lambda}{N}\sum_{i=1}^{N}[(x_i-\mu_k^{(t)})^2-\sigma_k^{2(t)}]\frac{z_{ik}^{(t)}}{2\sigma_k^{4(t)}} \tag{15}$$

where λ is the learning rate, and $z_{ik}^{(t)}$ is the posterior Bayesian probability defined by

$$z_{ik}^{(t)} = \frac{\pi_k^{(t)} g(x_i|\mu_k^{(t)},\sigma_k^{2(t)})}{f(x_i|\theta)}. \tag{16}$$

By adopting a stochastic gradient descent scheme for minimizing $D(f_{\mathbf{x}_r}||f_r)$ [22], the corresponding on-line formulation is obtained by simply dropping the summation sign and updating the parameters after each stimulus presentation; this is equivalent to approximating, at each step, the sum on the right side of (14) and (15) with just one term randomly drawn from the N terms. Furthermore, we employ a learning rate adaptation to increase the rate of convergence through the adaptive stochastic gradient descent algorithm ($k=1,\cdots,K$) [35] as in

$$\mu_k^{(t+1)} = \mu_k^{(t)} + a(t)(x_{t+1}-\mu_k^{(t)})z_{(t+1)k}^{(t)} \tag{17}$$

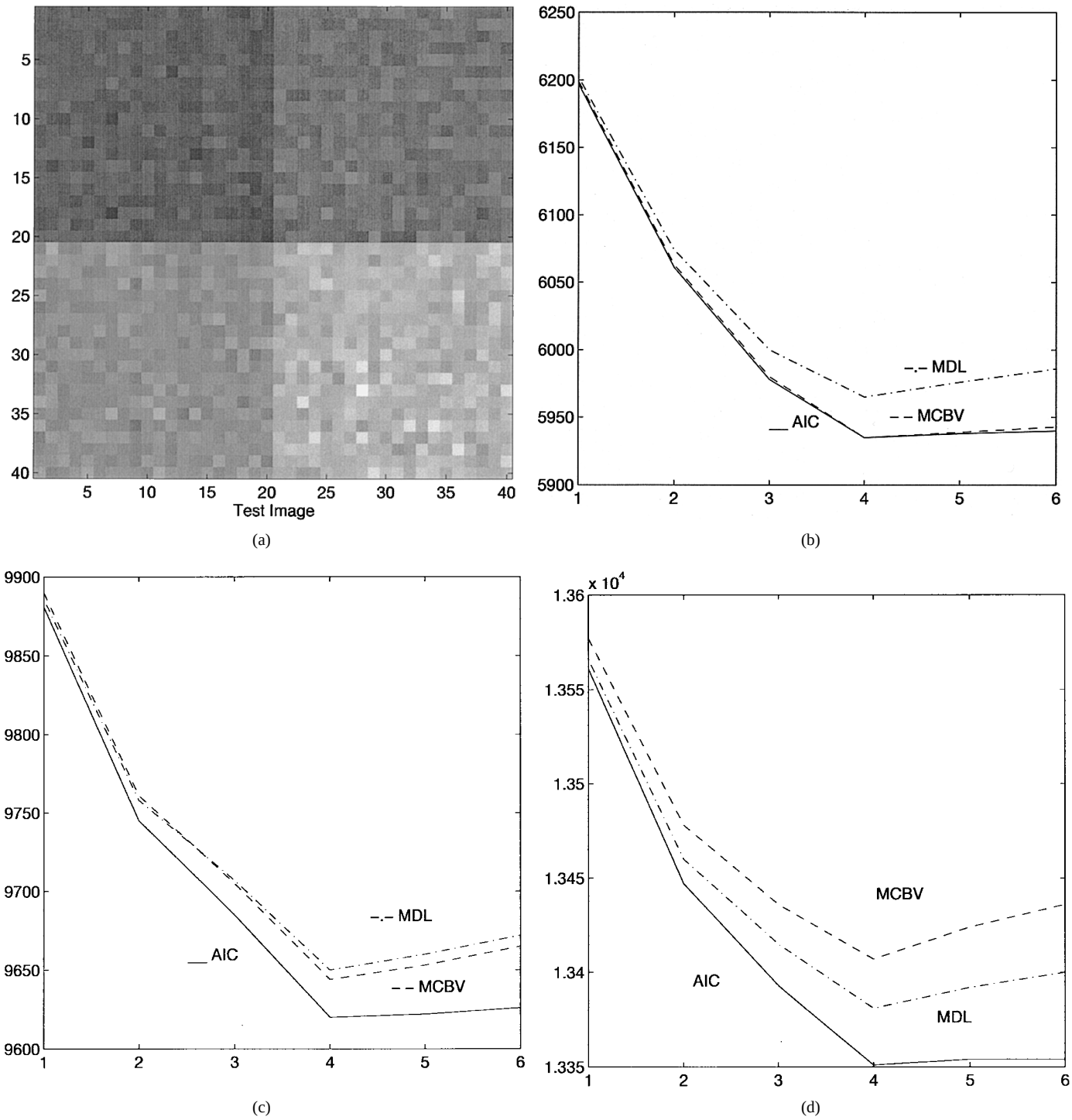
$$\sigma_k^{2(t+1)} = \sigma_k^{2(t)} + b(t)[(x_{t+1}-\mu_k^{(t)})^2-\sigma_k^{2(t)}]z_{(t+1)k}^{(t)} \tag{18}$$

where the variance factors are incorporated into the learning rates, while the posterior Bayesian probabilities are kept, and $a(t)$ and $b(t)$ are introduced as the learning rates, two sequences converging to zero, ensuring unbiased estimates after convergence. The idea behind this update rule is motivated by the principle that every weight of a network should be given its own learning rate and that these learning rates should be allowed to vary over time [35]. Based on generalized mean ergodic theorem [37], updates can also be obtained for the constrained regularization parameters π_k in the SFMD model. For simplicity, given an asymptotically convergent sequence, the corresponding mean ergodic theorem, i.e., the recursive version of the sample mean calculation, should hold asymptotically [3]. From the M-step of EM algorithm, we define the interim estimate of π_k by

$$\pi_k^{(t+1)} = \frac{t}{t+1}\pi_k^{(t)} + \frac{1}{t+1}z_{(t+1)k}^{(t)}. \tag{19}$$

Hence, the updates given by (17)–(19) provide the incremental procedure for computing the SFMD component parameters. Their practical use, however, requires a strong mixing condition (data randomization) and a decaying annealing procedure (learning rate decay) [40]. These two steps are currently controlled by user-defined parameters that may not be optimized for a specific case. Therefore, algorithm initialization must be chosen carefully and appropriately [12], [32]. An overall convergence dynamics of the PSOM is similar to the competitive learning (CL) algorithm in that a solution is obtained by "resonating" between input data and an internal representation [36]. Such a mechanism can be considered to be a more realistic learning tool than the EM algorithm. In addition, the data distribution for each class can also be modeled by a finite generalized Gaussian mixture (FGGM) given by [34], where $g(x_i|\theta_k)$ is the generalized Gaussian kernel representing the kth local cluster's pdf, which is defined by

$$g(x_i|\theta_k) = \frac{\alpha\beta_k}{2\Gamma(1/\alpha)}\exp[-|\beta_k(x_i-\mu_k)|^\alpha],\quad \alpha>0 \tag{20}$$



(a) (b) (c) (d)

Fig. 1. Original test image ($K_0 = 4$, SNR $= 10$ dB) and the AIC/MDL/MCBV curves in model selection (left to right: $\sigma = 3, 30, 300$).

where

μ_k mean;
$\Gamma(\cdot)$ Gamma function;
β_k parameter related to the variance σ_k by

$$\beta_k = \frac{1}{\sigma_k}\left[\frac{\Gamma(3/\alpha)}{\Gamma(1/\alpha)}\right]^{1/2}. \tag{21}$$

It has been shown that when $\alpha = 2.0$, we have the Gaussian pdf; when $\alpha = 1.0$, we have the Laplacian pdf. When $\alpha \gg 1$, the distribution tends to a uniform pdf; when $\alpha < 1$, the pdf becomes sharp. Therefore, the generalized Gaussian model is a suitable model for those data in which statistical properties are unknown, and the kernel shape can be controlled by selecting different α values.

C. Data Classification via Supervised Learning

The objective of data classification is to realize the class membership l_{ir} for all data points based on the observation x_i and the class statistics $\{P(\omega_r),\ f(u|\omega_r)\}$. It is well known that the optimal data classifier is the Bayes classifier since it can

achieve the minimum rate of classification error [38]. Measuring the average classification error by the mean squared error E, many previous researchers have shown that minimizing E by adjusting the parameters of class statistics is equivalent to directly approximating the posterior class probabilities when dealing with the two-class problem [2], [38]. In general, for the multiple class problem, the optimal Bayes classifier (minimum average error) classifies input patterns based on their posterior probabilities: Input x_i is classified to class ω_r if

$$P(\omega_r|x_i) > P(\omega_j|x_i) \tag{22}$$

for all $j \neq r$. It should be noted that in the formulation of classifier design, the optimal criterion used for the future data classification has been intuitively and directly applied to the learning of class statistics from the training data set.

Following this philosophy, great effort has been made in designing the network as an estimator of the posterior class probability [36]. By closely investigating the global class distribution modeling discussed in the previous section, we found that the classifier design for data classification can be dramatically simplified at the learning stage. Revisiting (2), since the class prior probability $P(\omega_r)$ is a known parameter when a supervised learning is applied, the posterior class probability $P(\omega_r|x_i)$ can be obtained without any further effort. Thus, by conditioning $P(\omega_r)$, the problem is formulated as a supervised classification learning of the class conditional likelihood density $f(u|\omega_r)$. It is very important that the learning process has been treated in a different way from the testing process while maintaining a consistency between the objective and the criterion. Moreover, when the ultimate goal of the learning is data classification, the question that may be asked is the following: Learning class likelihoods or decision boundaries? Since, in fact, only the decision boundaries are of the interests, the problem can be reformulated as the learning of the class boundaries (much more efficient) rather than the class likelihoods (generally time consuming). Thus, an efficient supervised algorithm to learn the class conditional likelihood densities called the "decision-based learning" [5] is adopted in this paper. The decision-based learning algorithm uses the *misclassified* data to adjust the density functions $f(u|\omega_r)$, which are initially obtained using the unsupervised learning scheme described previously so that the minimum classification error can be achieved. The algorithm is summarized as follows.

Define the rth-class discriminant function $\phi_r(x_i, \mathbf{w})$ to be $P(\omega_r)f(x_i|\omega_r)$. Given a set of training patterns $\mathbf{X} = \{x_i;\, i = 1, 2, \cdots, M\}$. The set $\mathbf{X}$ is further divided into the "positive training set" $\mathbf{X}^+ = \{x_i;\, x_i \in \omega_r,\, i = 1, 2, \cdots, N\}$ and the "negative training set" $\mathbf{X}^- = \{x_i;\, x_i \notin \omega_r,\, i = N+1, N+2, \cdots, M\}$. Define an energy function

$$E = \sum_{i=1}^{M} l(d(i)) \tag{23}$$

where

$$d(i) = \begin{cases} T - \phi_r(x_i, \mathbf{w}), & \text{if } x_i \in \mathbf{X}^+ \\ \phi_r(x_i, \mathbf{w}) - T, & \text{if } x_i \in \mathbf{X}^- \end{cases} \tag{24}$$

and where $T = \max_{\forall j \neq r}(\phi_j(x_i, \mathbf{W}))$. The *penalty function* l can be either a piecewise linear function

$$l(d) = \begin{cases} \zeta d, & \text{if } d \geq 0 \\ 0, & \text{if } d < 0 \end{cases} \tag{25}$$

where ζ is a positive constant or a sigmoidal function

$$l(d) = \frac{1}{1 + \exp^{-d\xi}}. \tag{26}$$

Notice that 1) energy function E is always large or equal to zero and 2) only misclassified training patterns contribute to the energy function. Therefore, the misclassification is minimized if E goes to the minimum.

The reinforced and antireinforced learning rules are used to update the network

$$\begin{aligned} &\text{Reinforced} \\ &\text{Learning:} \quad \mathbf{w}^{(j+1)} = \mathbf{w}^{(j)} + \eta l'(d(t))\nabla\phi(\mathbf{x}(t), \mathbf{w}) \\ &\text{Antireinforced} \\ &\text{Learning:} \quad \mathbf{w}^{(j+1)} = \mathbf{w}^{(j)} - \eta l'(d(t))\nabla\phi(\mathbf{x}(t), \mathbf{w}). \end{aligned} \tag{27}$$

If the misclassified training pattern is from a positive training set, reinforced learning will be applied. If the training pattern belongs to the negative training set, we antireinforce the learning, i.e., pull the kernels away from the problematic regions.

A probabilistic decision-based neural network (PDBNN) [6] is a probabilistic modular network designed especially for data classification where a Bayesian decomposition of the learning process provides a unique opportunity to optimize the structure of training scheme [4], [6], [25]. Since the information about class population is, in general, physically uncorrelated with the conditional features about the individual class, a decoupled two-step training, in terms of both network structure and learning rule, makes much more sense than that in the conventional posterior-typed neural networks, i.e., the conditional likelihood of each class and the class Bayesian prior should be adjusted separately in the classification spaces. In theory, when the cost function in future classification is defined as the average Bayes' risk (with a discrete version of squared or mean squared classification error) [2], a sufficient measure field, which is determined by the average likelihood risk, can be applied in the supervised learning [6].

Thus, PDBNN divides its network resources into M different pieces, and each piece is designated to one data class only, i.e., the subnet outputs of the PDBNN are designed to model the likelihood functions (likelihood-typed network). As illustrated in Fig. 2, the structure of the PDBNN consists of several disjoint subnets and a winner-take-all network, where the class likelihood functions are first estimated from equally presented class samples, and the final decision boundaries are determined simply weighting the likelihood by the class populations. Clearly, by taking the advantage of availability of class prior in supervised training, the cost function can be redefined, the sample set can be reorganized, and both the network structure and learning process can be dramatically simplified [4]. For a M-classification problem, PDBNN contains M different class subnets, each of which represents one

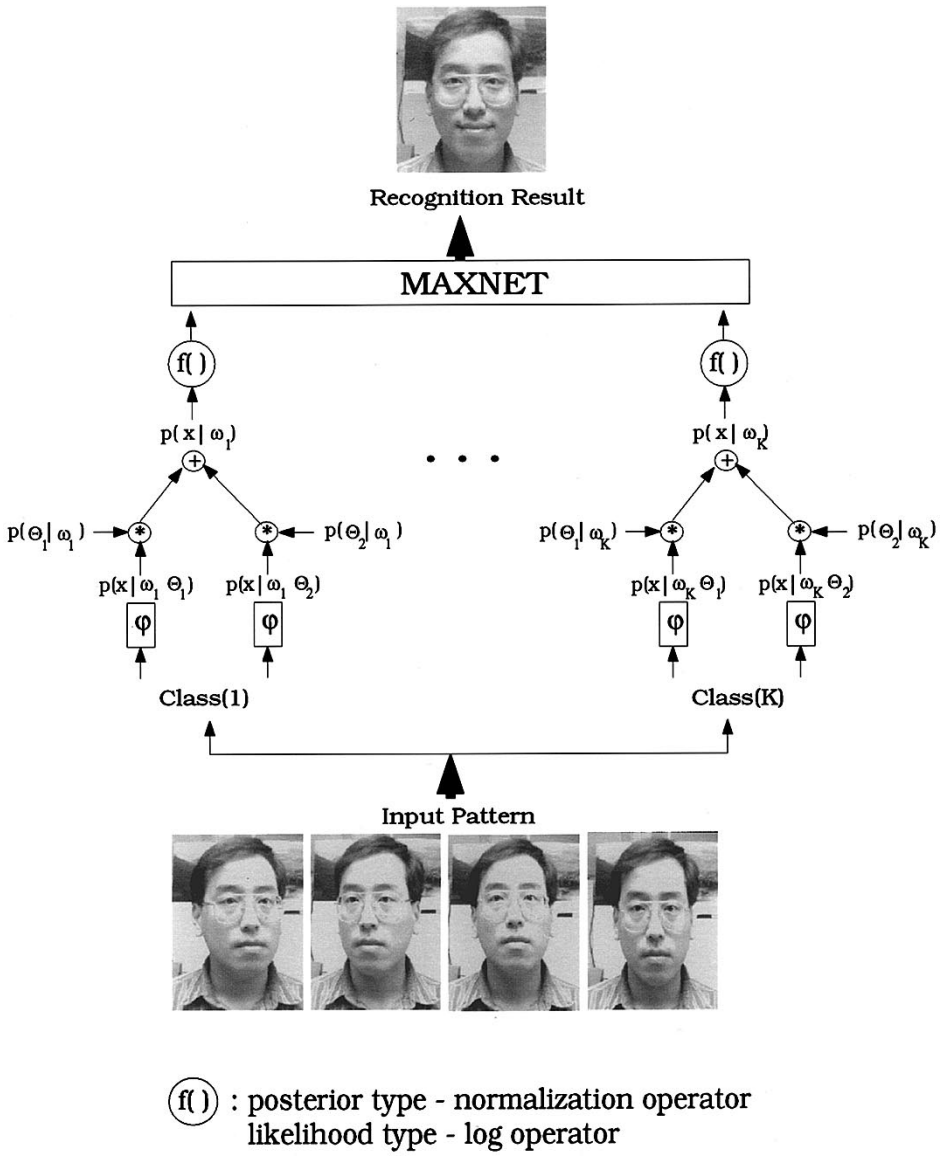


Fig. 2. PDBNN network structure. Each class subnet is designated to recognize one class. All the network weightings are in probabilistic format.

data class in the database. Within each subnet, several neurons (or clusters) are applied in order to handle problems that have complicated decision boundaries. The outputs of class subnets are fed into a winner-take-all network. The winner-take-all network categorizes the input pattern to the data class whose subnet produces the highest output value. Recall our problem formulation in Section II-C; it becomes clear that each piece of the PDBNN is exactly a PSOM subnet. Thus, when the ultimate goal is data classification, all of the network parameters can now be initialized by the quantification (unsupervised learning) step before supervised training. This initialization, together with the fact that the number of hidden units in each PSOM is relatively small compared with that of the PDBNN, makes PDBNN achieve a faster convergence rate and, often, better classification accuracy.

The training scheme of the PDBNN is based on the so-called locally unsupervised globally supervised (LUGS) learning. There are two phases in this scheme: During the locally unsupervised (LU) phase, each subnet is trained individually, and no mutual information across the classes may be utilized. Unsupervised algorithms such as the PSOM described in the previous section can be applied in this phase.

After the LU phase is completed, the training enters the globally supervised (GS) phase. In the GS phase, teacher information is introduced to reinforce or antireinforce the decision boundaries obtained during LU phase. There are three main aspects of this training phase.

1) *When to update:* A selective training scheme can be adopted, e.g., weight updating only when misclassification.
2) *What to update:* The learning rule is distributive and localized. It applies *reinforced learning* to the subnet corresponding to the correct class and *antireinforced learning* to the (unduly) winning subnet.
3) *How to update:* Adjust the boundary by updating the weight vector **w** either in the direction of the gradient of the discriminant function (i.e., reinforced learning) or the opposite of that direction (i.e., antireinforced learning).

Since only misclassified data points will be used for fine tuning of the decision boundaries, possible bias in the estimation of class distributions should be addressed. However, the key point we want to make is that this approach is very efficient, and although the global class description may be biased because of selective training, the decision boundaries will be more accurate. In fact, our intensive experiments indicate that only the data closed to the decision boundaries provide useful information in the boundary estimation. In particular, when the class distribution is formulated by a SFMD, the data far from the decision boundaries make little impact on the final classification results [6].

The discriminant functions in all clusters will be trained by the two-phase learning. A common model for the PDBNN to approximate the likelihood function is the mixture of Gaussians. The PDBNN designer can choose either hyperbasis function (HyperBF) or elliptical basis function (EBF) for the neurons to approximate full-rank or diagonal covariance matrices, respectively [6]. For the sake of simplicity, in this paper, we demonstrate the GS learning algorithm by using EBF only.

Suppose input pattern x_i is a D-dimensional vector $x_i = [x_i^1, x_i^2, \cdots, x_i^D]^T$. Its EBF for cluster θ_k in class ω_r is

$$\psi(x_i, \omega_r, \theta_k) = -\tfrac{1}{2} \sum_{d=1}^{D} \beta_{rkd}(x_i^d - w_{rkd})^2 + C_{rk} \qquad (28)$$

where $C_{rk} = -(D/2)(\ln 2\pi - \sum_{d=1}^{D} \ln \beta_{rkd})$. The initial values of the cluster parameters, i.e., β and w, can be obtained by PSOM. The discriminant function $\phi_r(x_i, \mathbf{w})$ for class r (see Section II-C) becomes

$$\phi_r(x_i, \mathbf{w}) = P(\omega_r) \sum_{k=1}^{K_r} \pi_k \exp(\psi(x_i, \omega_r, \theta_k)). \qquad (29)$$

By applying reinforced and antireinforced learning rules in (29), β and w can further be updated. The gradient vectors for EBF at iteration j are computed as

$$\left.\frac{\partial \phi_r(x_i, \mathbf{w})}{\partial w_{rkd}}\right|_{\mathbf{w}=\mathbf{w}^{(j)}} = h_{irk}^{(j)} \cdot \beta_{rkd}^{(j)}(x_i^d - w_{rkd}^{(j)})$$

$$\left.\frac{\partial \phi(x_i, \mathbf{w})}{\partial \beta_{rkd}}\right|_{\mathbf{w}=\mathbf{w}^{(j)}} = \frac{h_{irk}^{(j)}}{2}\left(\frac{1}{\beta_{rkd}^{(j)}} - (x_i^d - w_{rkd}^{(j)})^2\right) \qquad (30)$$

$$h_{irk}^{(j)} = \frac{\pi_k^{(j)} \exp(\psi^{(j)}(x_i, \omega_r, \theta_k))}{\sum_{l} \pi_l^{(j)} \exp(\psi^{(j)}(x_i, \omega_r, \theta_l))}. \qquad (31)$$

The cluster prior probabilities π_k can also be updated by

$$\pi_k^{(j+1)} = (1/N_r) \sum_{i=1}^{N_r} h_{irk}^{(j)}. \qquad (32)$$

III. Application Examples and Discussions

A. Medical Image Quantification

In this section, we present the results using the information theoretic criteria to determine the appropriate number and/or kernel shape of tissue types (with a correspondence to the local clusters) in the real MR brain images and digital mammograms as well as the results using the proposed quantification technique (e.g., the PSOM) to estimate the tissue quantities from these images. A fully automatic thresholding method, adaptive Lloyd–Max histogram quantization (ALMHQ) that we introduced recently in [12] is used to initialize the quantification, and the tissue parameters are then finalized by the PSOM. For the validation of the tissue quantification using the proposed algorithms, the global relative entropy (GRE) value is used as an objective measure to evaluate the accuracy of the data quantification, which is consistent with our problem formulation in Section II-B. The objective of the experiment is to illustrate the algorithm performance on real-world applications.

Fig. 3(a) and (b) show the original data consisting of two adjacent, T1-weighted images parallel to the anterior commissural-posterior commisural (AC-PC) line and the corresponding image histograms (c) and (d). This data were acquired with a General Electric (GE) Sigma 1.5 Tesla system. The imaging parameters are TR 35, TE 5, flip angle 45°, 1.5-mm effective slice thickness, 0 gap, 124 slices with in plane 192×256 matrix, and a 24-cm field of view. Since the skull, scalp, and fat in the original brain images do not contribute to the brain tissue, we edit the MR images to exclude nonbrain structures prior to tissue quantification [24]. Experience indicates that this procedure helps to achieve better quantification of brain tissues by delineation of the other tissue types that are not clinically interesting [9]. It can be clearly seen that the histograms have different shapes from slice to slice and that the tissue types are highly overlapped. This situation presents a great challenge to any computerized technique, even though it has been successful in the simulation study. In this study, in addition to the "gold standard" evaluation performed by neuroradiologists [8], we use the GRE value to reflect the quality of tissue quantification.

Based on pre-edited MR brain image, the procedure for quantifying the tissue types in one slice is summarized as follows.

1) For each value of K (number of tissue types), ML tissue quantification is performed by the PSOM algorithm.
2) Scan the values of $K = K_{\min}, \cdots, K_{\max}$, and use the information-theoretic criteria to determine the suitable number of tissue types.
3) Select the result of tissue quantification corresponding to the value of K_0 determined in Step 2).
4) Evaluate the performance of tissue quantification in terms of the GRE value, convergence rate, and computational complexity.

In our experiment, since the number of tissue types is unknown, we first show that the number of tissue types varies from slice to slice. Let $K_{\min} = 2$ and $K_{\max} = 9$, and calculate AIC(K), MDL(K), and MCBV(K) ($K = K_{\min}, \cdots, K_{\max}$). We obtained the results shown in Fig. 4, which suggested that the two brain images contain six and eight tissue types, respectively. According to the model fitting procedure in designing the optimal structure of the modular networks we discussed before, the minima of these criteria also determines

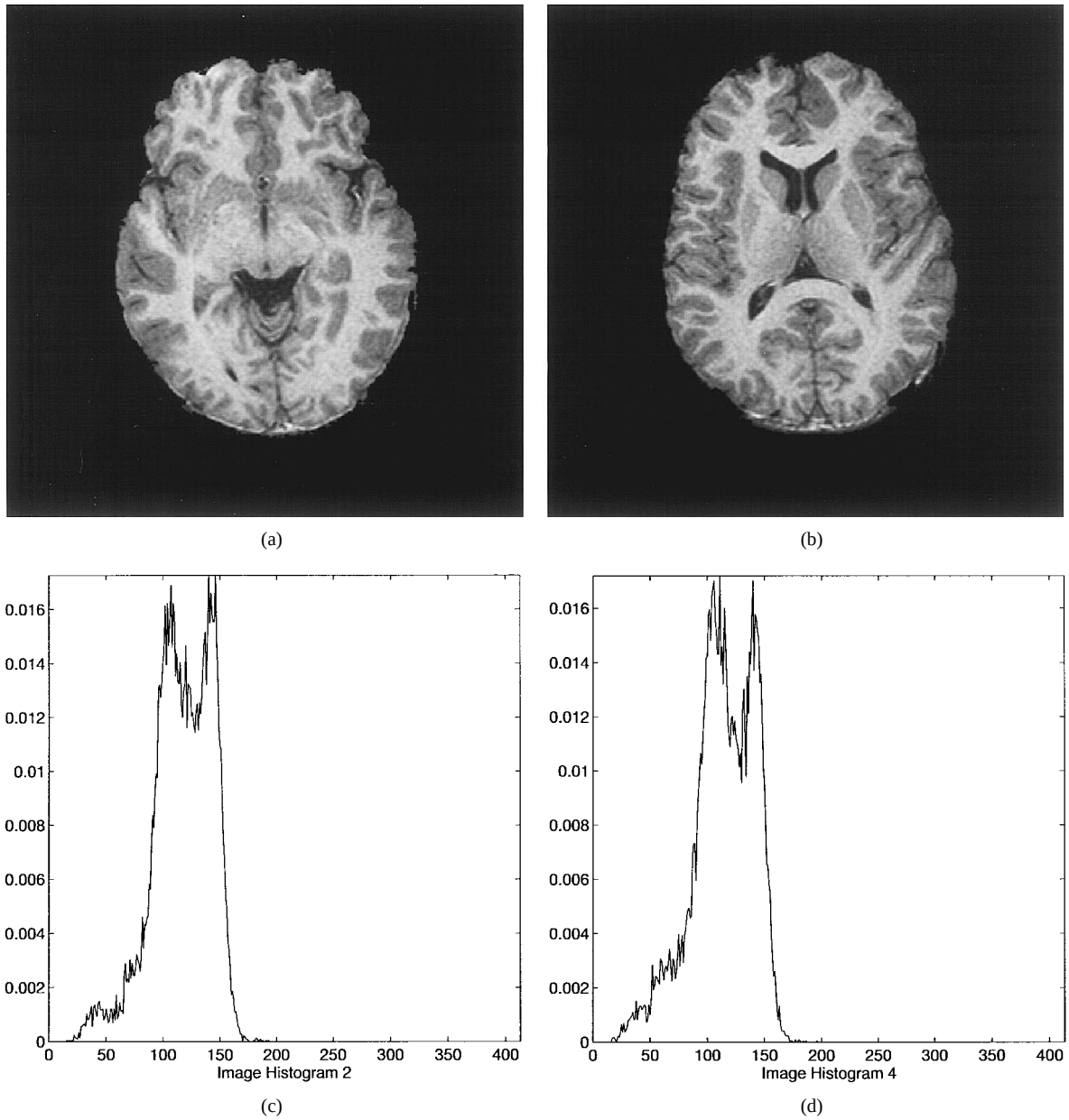


Fig. 3. Pure brain tissues extracted from (a) and (b) original MR images and (c) and (d) the corresponding histograms.

the most appropriate number of mixture components in the corresponding PSOM. These figures show that the overall performance of the three information-theoretic criteria is fairly consistent when applied to the real MR brain images. Our experience indicates, however, that AIC tends to overestimate while MDL tends to underestimate the number of tissue types, and MCBV provides the solution between those of AIC and MDL, which is believed to be more reasonable especially in terms of providing a balance between the bias and variance of the parameter estimates. As discussed in the literature, brain material is generally composed of three principal tissue types, i.e., WM, GM, CSF, and their pair-wise combinations known as the partial volume effect. Previous studies have proposed a six-tissue model representing the primary tissue types, and the mixture tissue types were defined as CSF-white (CW), CSF-gray (CG), and gray-white (GW). In this work, we also consider the triple mixture tissue, which is defined by CSF-white-gray (CWG). More importantly, since the MRI scans clearly show the distinctive intensities at the local brain areas, the functional tissue types need to be considered. In particular, caudate nucleus and putamen are the two important local brain functional areas.

For each fixed K, the PSOM algorithm is iteratively used to quantify the different tissue types, where the learning is fully data-driven [12]. For slice 2, the results of final tissue quantification with $K_0 = 7, 8, 9$ are shown in Fig. 5. Corresponding to $K_0 = 8$, a GRE value of 0.02–0.04 nats in quantification is achieved. It was found that most of the variance parameters are different, which suggests that assuming the same variance for each tissue type with distinct image-intensity distribution may not be realistic. These quantified tissue types agreed with that of a physician's qualitative analysis results.

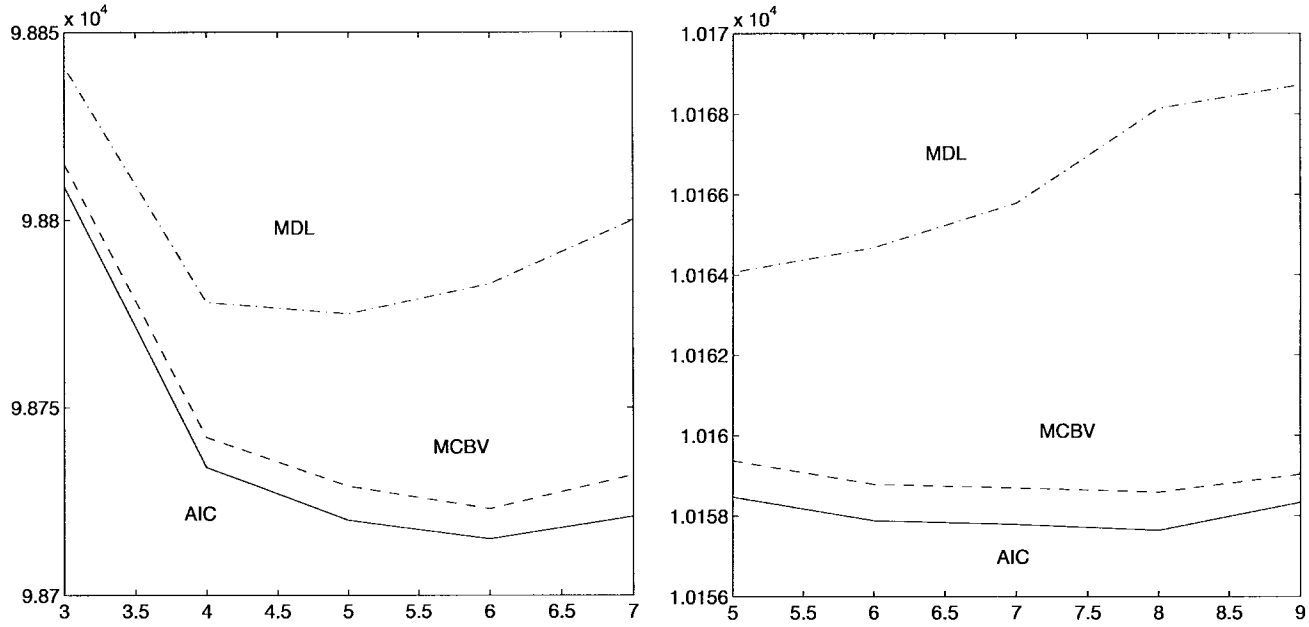


Fig. 4. Results of model selection for slice 1-2 ($K_0 = 6$ and 8, left to right).

We then present a comparison of the performance of PSOM with that of the EM [3], [19], [21] and the CL [6], [22] algorithms on MR brain tissue quantification. The task is to evaluate the computational accuracy and efficiency of the algorithm in the standard finite normal mixture distribution learning. To be able to make fair comparisons with the other two methods, we applied all the methods to the same example and used the GRE value between the image histogram and the estimated SFNM distribution as the goodness criterion to evaluate the quantification error. The left side of Fig. 6 shows learning curves of the PSOM and competitive learning (CL) averaged over five independent runs. As observed in the figure, PSOM outperforms CL learning by faster convergence rate and lower quantification error, where the final GRE value is about 0.04 nats. The right side of Fig. 6 presents the comparison of PSOM with that of the EM algorithm for 25 epochs. From the learning curves, again note that the PSOM algorithm shows superior estimation performance. The final quantification error is about 0.02 nats while preserving the faster convergence rate.

We have also applied the same procedure to the digital mammograms given in Fig. 7, where we show that if the number of cluster K is known, the kernel shape of local clusters will affect the accuracy of the histogram quantification for real mammographic images. Since, in this case, we do not assume a fixed kernel shape, FGGM is used, and three information criteria (AIC, MDL, and MBVC) were used to determine both the number and kernel shape of the regions in the digital mammograms. Twenty real mammograms with masses were chosen as testing images. The selected mammograms were digitized with an image resolution of $100\ \mu\text{m} \times 100\ \mu\text{m}$ per pixel by the laser film digitizer (Model Lumiscan 150). The image sizes are $1792 \times 2560 \times 12$ b/pixel. We found that, although with different α, all three criteria achieved minimum when $K = 8$. It indicates that these information criteria are relatively insensitive to the change of α, as also claimed in [34]. With this observation, we can further decouple the relation between K and α and choose the appropriate value of one while fixing the value of another. It is interesting to note that the result of model selection here is very consistent with the conclusion in some previous studies: according to the work in [41], the most appropriate region number (K) is eight for most digital mammograms. We fixed $K = 8$, and changed the values of α for estimating the FGGM model parameters using the PSOM/EM algorithm. The GRE value between the histogram and the estimated FGGM distribution is used as a measure of the estimation bias. We found that GRE achieved a minimum value when $\alpha = 3.0$ as shown in Fig. 8. Compared with the conventional finite normal mixture model ($\alpha = 2.0$), which has been mostly chosen by many previous researchers, this experiment indicates that the FGGM model provides more freedom, thus allowing its correct uses to the situation when the true statistical properties of the digital mammograms are not available.

B. Face Recognition Experiment

A PDBNN-based face recognition system [6] is being developed under a collaboration between Siemens Corporate Research, Princeton, NJ, and Princeton University, Princeton, NJ. The total system diagram is depicted in Fig. 9. All four main modules—face detector, eye localizer, feature extractor, and face recognizer—are implemented on a SUN Sparc10 workstation. An RS-170 format camera with 16 mm, F1.6 lens is used to acquire image sequences. The S1V digitizer board digitizes the incoming image stream into 640×480 8-bit gray-scale images and stores them into the frame buffer. The image acquisition rate is on the order of 4–6 frames/s. The acquired images are then down sized to 320×240 for the following processing.

As shown in Fig. 9, the processing modules are executed sequentially. A module will be activated only when the in-

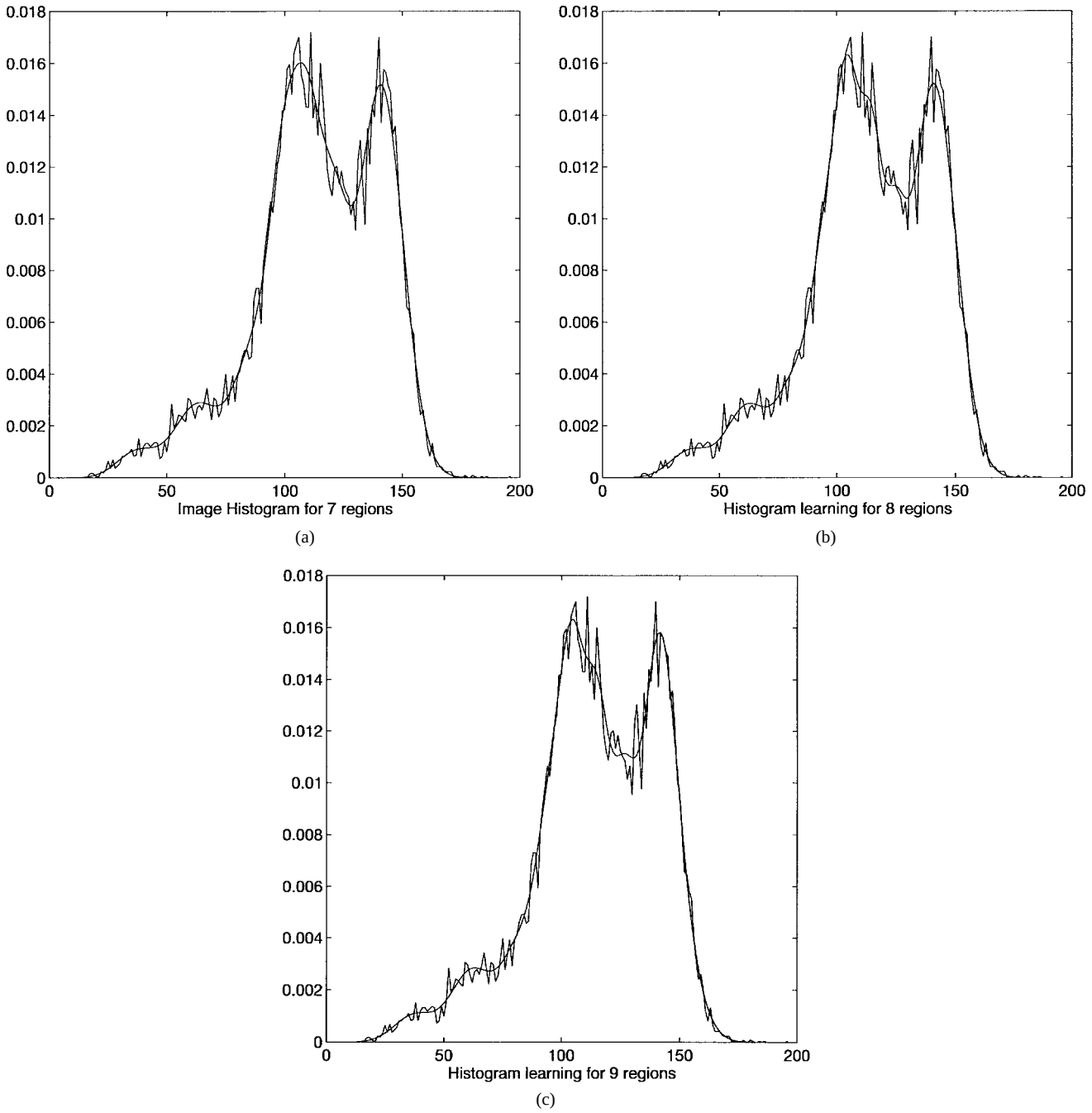


Fig. 5. Histogram learning for slice 2. (a) $K_0 = 7$. (b) $K_0 = 8$. (c) $K_0 = 9$.

coming pattern passes the preceding module (with an agreeable confidence). After a scene is obtained by the image acquisition system, a quick detection algorithm based on binary template matching is applied to detect the presence of a proper sized moving object. A PDBNN face detector is then activated to determine whether there is a human face. If positive, a PDBNN eye localizer is activated to locate both eyes. A subimage ($\approx 140 \times 100$) corresponding to the face region will then be extracted. Finally, the feature vector is fed into a PDBNN face recognizer for recognition and subsequent verification.

The system built on the proposed one has been demonstrated to be applicable under reasonable variations of orientation and/or lighting and with the possibility of eyeglasses. This method has been shown to be very robust against large variation of face features, eye shapes, and cluttered background [6]. The algorithm takes only 200 ms to find human faces in an image with 320×240 pixels on a SUN Sparc10 workstation. For a facial image with 320×240 pixels, the algorithm takes 500 ms to locate two eyes. In the face recognition stage, the computation time is linearly proportional to the number of persons in the database. For a 200-person database, it takes less than 100 ms to recognize a face. Furthermore, because of the inherent parallel and distributed processing nature of PDBNN, the technique can be easily implemented via specialized hardware for real-time performance.

We conduct an experiment on the face database from the Olivetti Research Laboratory, Cambridge, U.K. (the ORL database). There are ten different images of 40 different

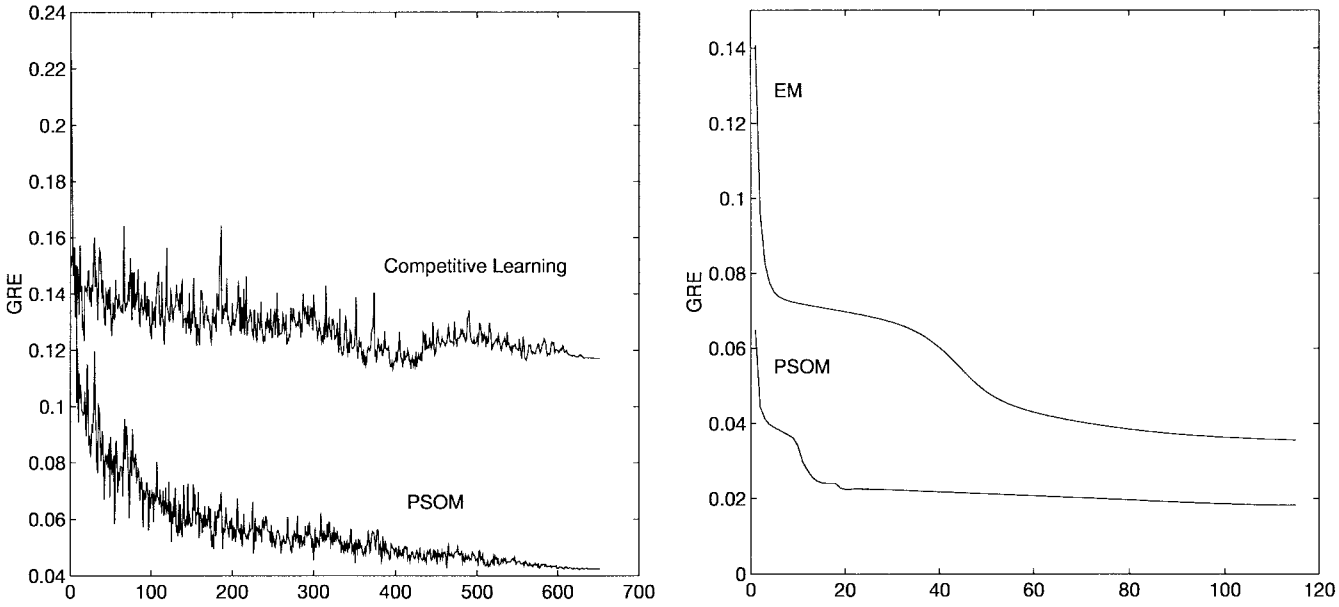


Fig. 6. Comparison of the learning curves of (left) PSOM and CL and (right) EM.

Fig. 7. Typical image of an original digital mammogram.

people. There are variations in facial expression (open/close eyes, smiling/nonsmiling), facial details (glasses/no glasses), scale (up to 10%), and orientation (up to 20°). A HMM-based approach is applied to this database and achieves 13% error rate [13]. The popular eigenface algorithm [16] reports the error rate around 10% [13], [14]. In [15], a pseudo 2-D HMM method is used and achieves 5% at the expense of long computation time (4 m/pattern on Sun Sparc II). In [14], Lawrence *et al.* use the same training and test set size as Samaria did as well as a combined neural network (self organizing map and convolutional neural network) to do the recognition. This scheme spent 4 hr to train the network and less than 1 s to recognize one facial image. The error rate for the ORL database is 3.8%. Our PDBNN-based system reaches similar performance (4%) but has much faster training and recognition speed (20 m for training and less than 0.1 s for recognition). Both approaches run on SGI Indy. Table I summarizes the performance numbers on the ORL database.

TABLE I
PERFORMANCE OF DIFFERENT FACE RECOGNIZERS ON THE ORL DATABASE. PART OF THIS TABLE IS ADAPTED FROM S. LAWRENCE *et al.*, "FACE RECOGNITION: A CONVOLUTIONAL NEURAL NETWORK APPROACH," TECHNICAL REPORT, NEC RESEARCH INSTITUTE, 1995

System	Error rate	Classification time	Training Time
PDBNN	4%	< 0.1 seconds	20 minutes
SOM + CN	3.8%	< 0.5 seconds	4 hours
Pseudo 2D-HMM	5%	240 seconds	n/a
Eigenface	10%	n/a	n/a
HMM	13%	n/a	n/a

We have also applied the PDBNN method to the so-called "M + 1 classes" problem in which the pattern under testing could be either from one of the M classes or from some other unknown class (the "unknown" class or the "intruder" class). Note that the unknown class probability is often very hard to estimate, and for some applications, it is almost impossible to obtain enough training samples for the unknown class (for example, in the face recognition problem, the unknown class includes the faces all over the world). In our experiment, PDBNN uses a different decision rule from that of the "M class" problem: Pattern x_i belongs to class r if both of the following conditions are true: a) $\phi(\omega_r, x_i) > \phi(\omega_j, x_i), \forall\, j \neq r$, and b) $\phi(\omega_r, x_i) > T$, T is a threshold obtained by decision-based learning. Otherwise, pattern x_i belongs to the unknown

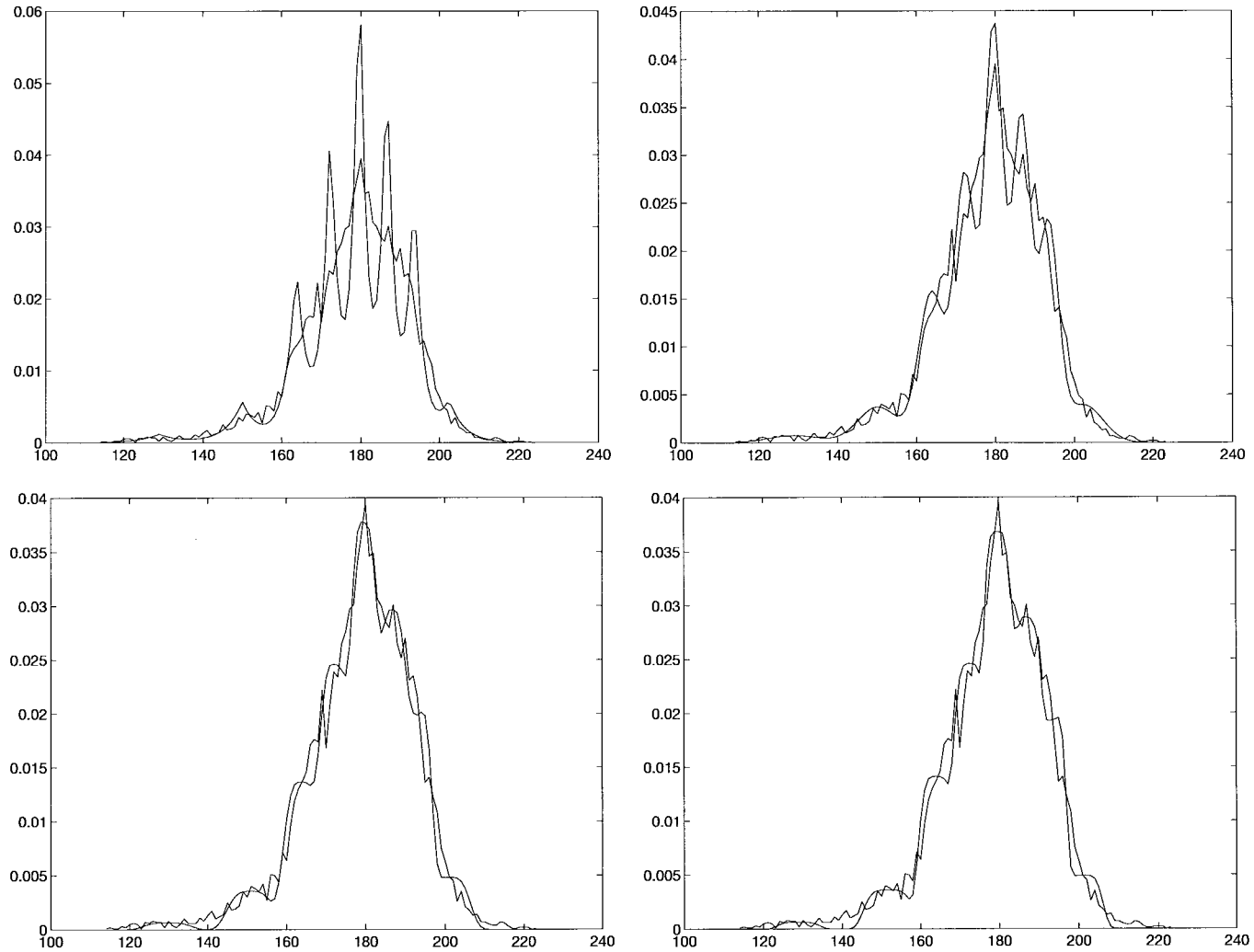


Fig. 8. Comparison of mammogram histogram learning with different kernel shapes ($K_0 = 8$).

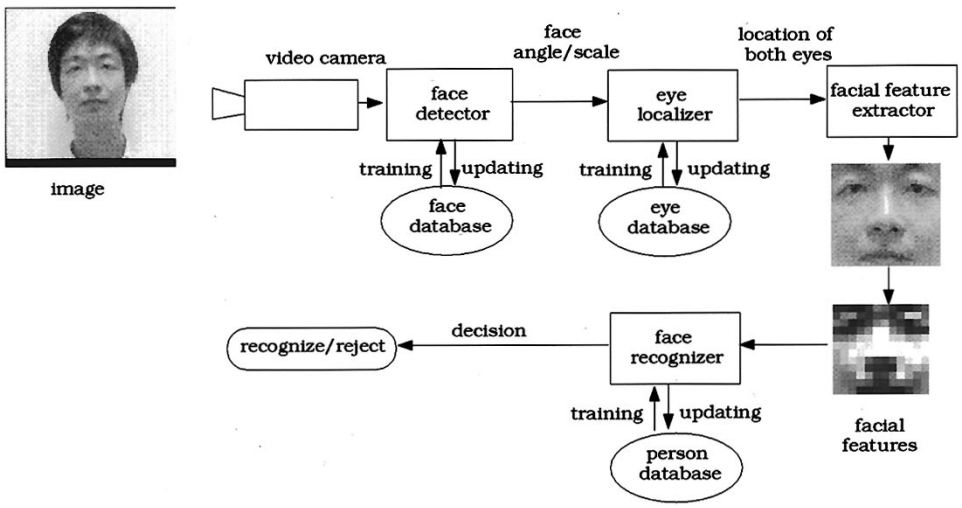


Fig. 9. System configuration of the face recognition system. Face recognition system acquires images from video camera. Face detector determines if there are faces inside images. Eye localizer indicates the exact positions of both eyes. It then passes their coordinates to facial feature extractor to extract low-resolution facial features as input of face recognizer.

class. We observed consistent and significant improvement in classification results, comparing pure Bayesian decision and the PDBNN approach (e.g., recognition rate from 70–90%) contributed by the fine-tuning process [6]. The following example further shows the effect of the fine-tuning process: For 100-person face recognition, we have 500 training patterns/person and 20 test patterns/person. After the LU phase, we obtained a training accuracy of 89.2% (44 608/50 000) and

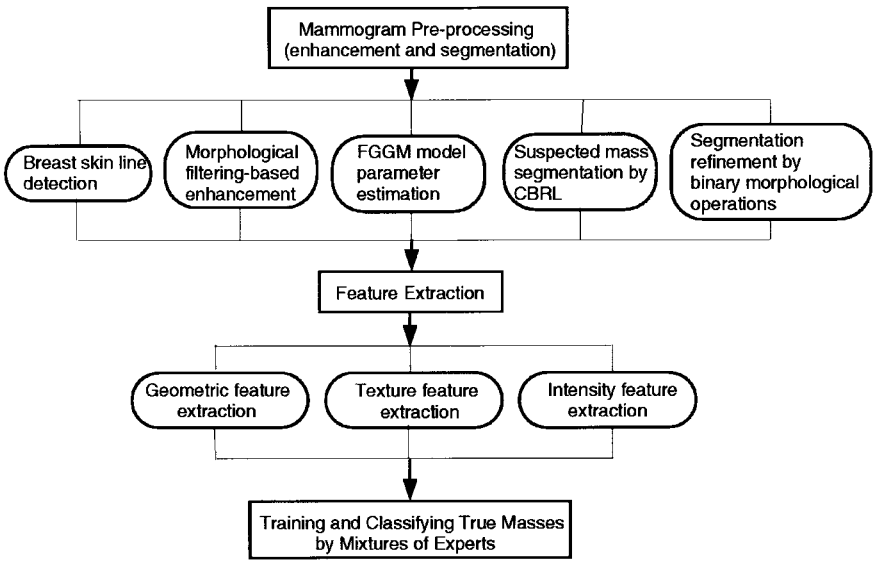
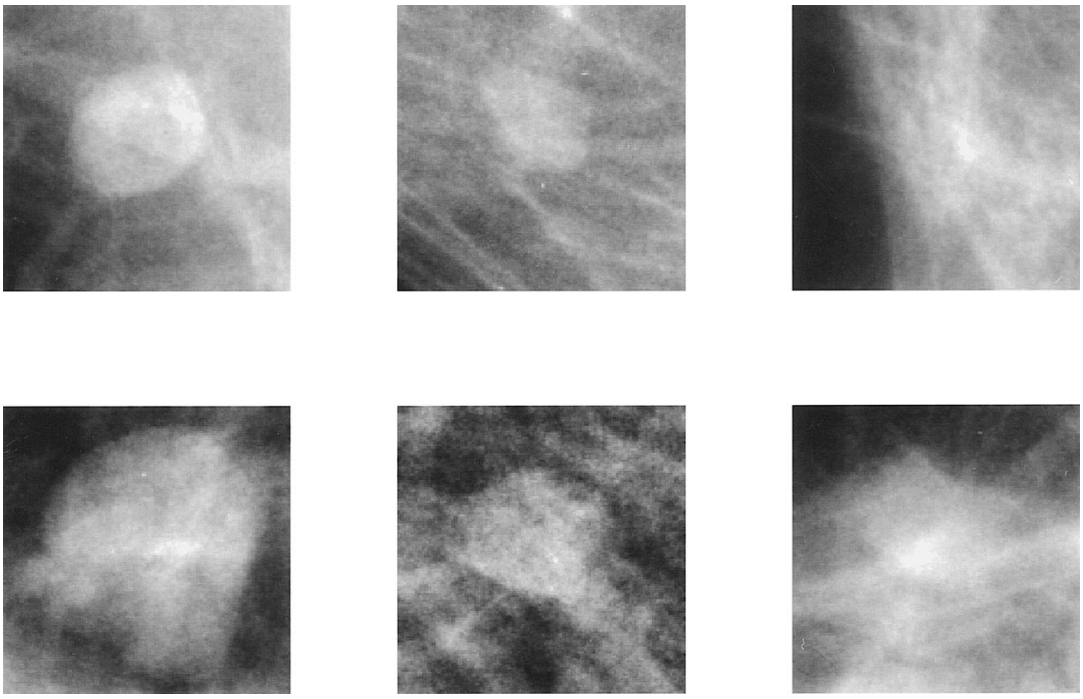


Fig. 10. Flow diagram of mass detection in digital mammograms.



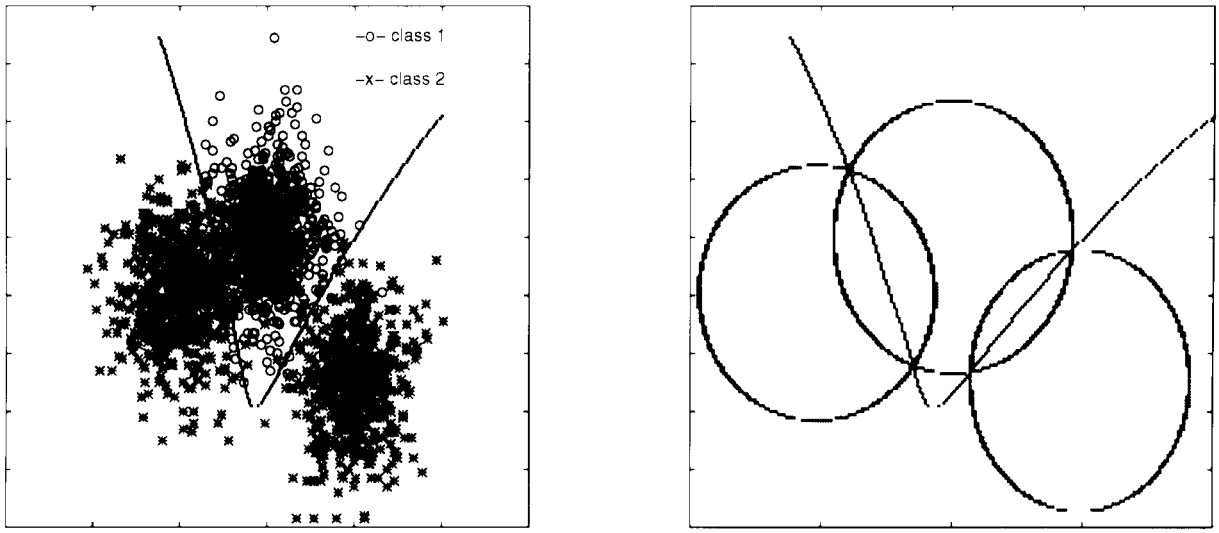
(a) (b) (c)

Fig. 11. Typical mass appearances in mammograms. (a) Well-defined masses. (b) Ill-defined masses. (c) Spiculated masses.

a test accuracy of 71.5% (1430/2000). After the GS phase, we improved the performance to a training accuracy of 98.9% (49 495/50 000) and a test accuracy of 96.2% (1924/2000). Nevertheless, when we have the luxury of knowing the object probability model in advance, the fine-tuning process may not be necessary. It is reasonable to acknowledge that the face recognition result from our experiment is valid since the ORL database is a widely used public database like the FERET database. With a comparison with the recognition rate of the eigenface method on an early FERET database (smaller size), we found that the performance of the proposed method is comparable and/or superior to the eigenface approach.

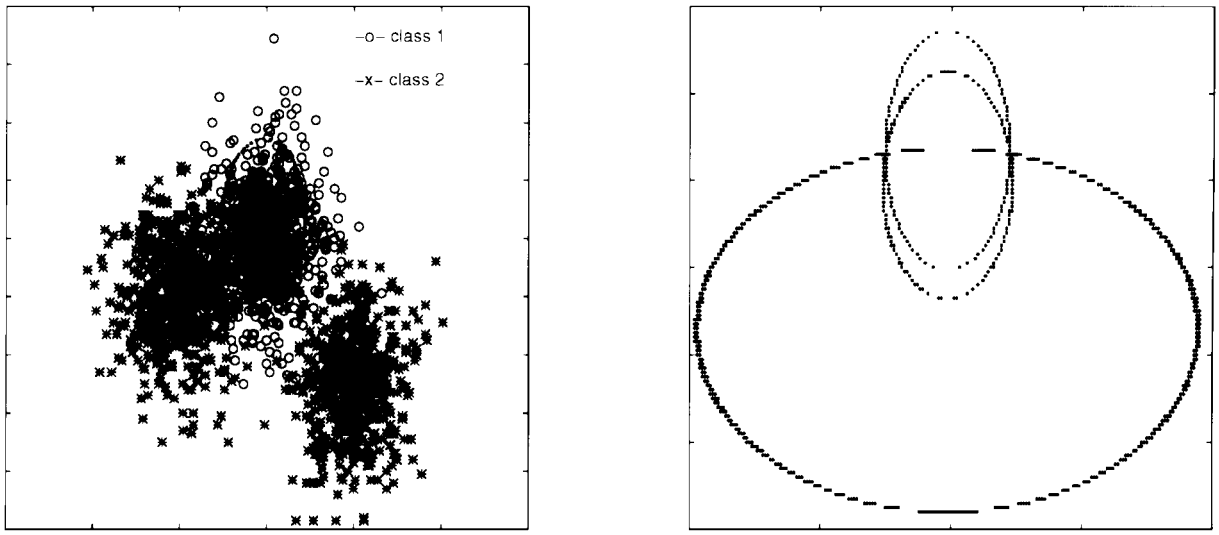
C. Featured Database Analysis

As we have discussed in Sections I and II, model selection is the first and a very important learning task in mapping a database, and the objective of the procedure is to determine both the number and kernel shape of local clusters in each class. The inaccuracy in model selection will affect the performances of both data quantification and classification. Using the proposed learning scheme, the structure of the probabilistic modular networks will be optimized following the model selection and PSOM [7], [32]. When all the class distributions are learned accurately, further data classification



(a) (b)

Fig. 12. Two-dimensional feature space in classification example 1 where "*o*" denotes true mass cases; "*" denotes false mass cases. (a) Class 2 contains two clusters. (b) Decision boundary learning with four cross points.



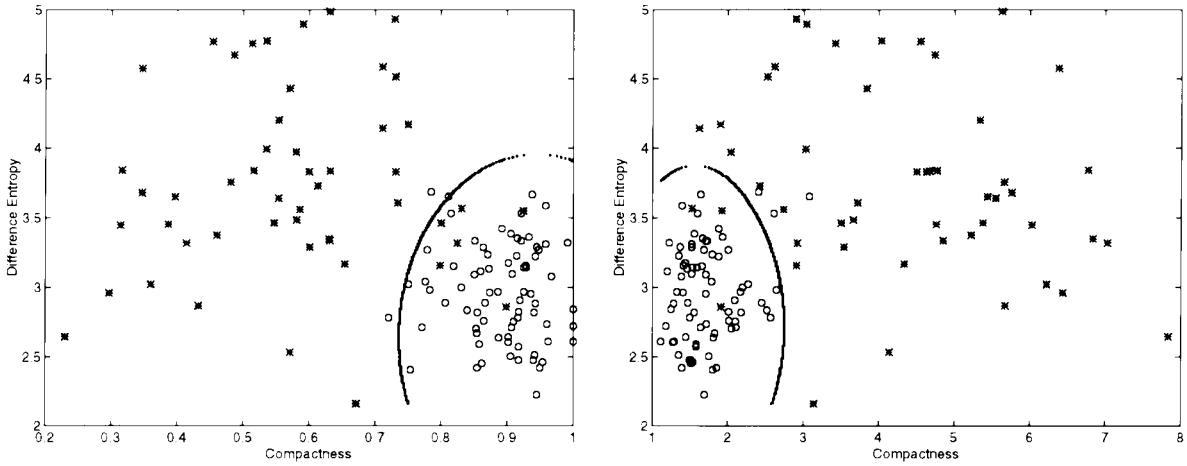
(a) (b)

Fig. 13. Two-dimensional feature space in classification example 2, where "*o*" denotes true mass cases, and "*" denotes false mass cases. (a) Class 2 contains one cluster. (b) Decision boundary learning with two cross points.

will be achieved simply following Bayesian rule [38]. In this subsection, these objectives and the related conclusions are further illustrated by two examples in the computed-aided diagnosis (CAD) for breast cancer detection [7]. The objective is to detect masses in digital mammography since masses are the important signs leading to early breast cancer [7]. For the purpose of improving the performance of CAD for detection of early breast cancer in mammography, a crucial step in any strategic solution is to quantitatively analyze the featured database (with the cases of normal and cancer tissues), i.e., to create a map of the feature distributions regarding the disease patterns [4], [7]. Since the featured database in CAD is constructed from the preprocessed suspected regions, model selection is very important in providing useful diagnostic suggestions. Furthermore, based on the feedback after all possible lesions are detected and their features are quantified, database quality and learning capability of the CAD system design can also be analyzed by the model selection, comparing different feature extraction and database construction schemes [4]. The framework of the proposed method for mass detection is illustrated in Fig. 10.

Some typical mass cluster appearances on mammograms are displayed in Fig. 11. With a preprocessing step, all suspected mass regions, as well as some normal dense tissues with brighter intensities, are located. The latter should be eliminated from the true masses through feature discrimination. On the clinical site, masses are evaluated based on the location, density, size, shape, margins, and the presence of associated calcifications.

In the first example, we show that the inappropriate determination of the number of clusters inside each class will affect the performance of data classification. Since a classi-



(a) (b)

Fig. 14. Classification results. –o– denotes true mass cases; –*– denotes false mass cases. (a) Classification using compactness definition 1. (b) Classification using compactness definition 2.

fication based on feature space is commonly used in many pattern analysis applications, including mammographic mass detection, typical intensity, geometric, and texture features are extracted and investigated from the segmented regions. These features usually possess clinical significance and are widely used in most CAD systems. A detailed description of feature extraction can be found in [7]. Suppose we extract two major features that characterize the two targeted classes (mass and nonmass), as it shown in Fig. 12. In this example, class 1 contains one cluster, and class 2 contains two clusters. The 2-D histogram pairs of these features extracted from true and false mass regions are investigated, and the features that can better separate the true and false mass regions are selected for further study. In this study, area, compactness (circularity), and difference entropy were found to have better discrimination and reliability properties. Therefore, we chose them to perform the classification.

Two PDBNN-like modular networks are trained to classify these two classes. The classification results are shown in Figs. 12 and 13. The result in Fig. 12 is with the right cluster number in Class 2. The result in Fig. 13 is with the wrong cluster number in Class 2. In this simple experiment, it is clearly shown that comparing the results in Fig. 12 with those in Fig. 13, the classification boundary with the right cluster number may be much more accurate than that with the heuristically determined cluster number since the decision boundary between classes 1 and 2 will be determined by four cross points in the first case, whereas in the second case, the decision boundary will be determined by only two cross points. From this example, we can show that the error of data classification is controlled by the accuracy in estimating the decision boundaries between classes, and the quality of the boundary estimates is indeed dependent on both the bias and variance of the class likelihood estimates. It can be seen that the bias may be lower in case 1 than in case 2, but the variance will be higher in case 1 than case 2. A similar example is the curve fitting from noisy data [31].

In the second example, we use the proposed classifier to distinguish true masses from false masses based on the features extracted from the suspected regions. The objective is to reduce the number of suspicious regions and identify the true masses. We selected 150 mammograms from the mammographic database. Each mammogram contained at least one mass case of varying size and location. The areas of suspicious masses were identified by an expert radiologist based on visual criteria and biopsy-proven results. We selected 50 mammograms with biopsy-proven masses from the data set for training. The mammogram set used for testing contained 46 single-view mammograms: 23 normal cases and 23 with biopsy-proven masses. The feature vector contained two features: compactness and difference entropy. According to our investigation, these two features have the better separation (discrimination) between the true and false mass classes. These features are also not correlated with each other. According to our experience, the values of compactness with definition 1 are more reliable than those of compactness with [7, Def. 2]. A training feature vector set was constructed from 50 true-mass ROI's and 50 false-mass ROI's. The training set was used to train two modular probabilistic decision-based neural networks separately. Fig. 14(a) shows the classification of two classes with compactness definition 1. Fig. 14(b) shows the classification of two classes with compactness definition 2.

In our evaluation study, six to 15 suspected masses per mammogram were detected and required further evaluation. The receiver operating characteristic (ROC) method is used to evaluate the detection performance of our method [38]. In the ROC analysis, the distribution of the positive and negative cases can be represented by certain probability distributions. When the two distributions overlap on the decision axis, a cutoff point can be made at an arbitrary decision threshold. The corresponding true-positive fraction (TPF) versus false-positive fraction (FPF) for each threshold can be drawn on a plane. By indicating several points on the plot, curve fitting can be employed to construct an ROC curve. The area under the curve, which is referred to as Az, can be used as a performance index of the system. In general, the higher the Az, the better the performance. In addition, two other indexes [sensitivity (TPF) and specificity (1-FPF)] are usually used to

evaluate the system performance on the specified point of the ROC curve. In this study, a computer program (LABROC) is employed for the evaluation analysis. We found that the proposed classifier can reduce the number of suspicious masses with a sensitivity of 84% at a specificity of 82% (1.6 false positive findings per mammogram) based on the database containing 46 mammograms (23 of them have biopsy-proven masses). In conclusion, when compared with the conventional neural networks, the probabilistic modular networks can lead to more efficient learning and provide better understanding in the analysis of the distribution patterns of multiple features extracted from the suspicious masses.

IV. Conclusions and Discussions

We have presented a strategy for mapping a database by probabilistic modular networks and information-theoretic criteria. Local class distribution is modeled by a standard finite mixture. Information-theoretic criteria are applied to detect the number and shape of local clusters, thus allowing the corresponding neural network to adaptively evolve its structure to the best representation of the local data. The PSOM algorithm is used to quantify the parameters of the local clusters, leading to an ML estimation. The decision boundaries in the data classification are then fine tuned by a global supervised learning. The results obtained by using the simulated data and the real databases demonstrate the promise and effectiveness of the proposed technique.

Our main contribution is the complete proposal of a detripled learning strategy for the determination of both modular and components of the network. In this approach, the network structures (in terms of which statistical model is more suitable) are justified in a first step and followed by a soft classification of the data (in terms of each data point supports all local clusters simultaneously). The associated probabilistic class labels are then realized in a third step as the competitive learning of this induced hard classification task. To summarize, the results of the experiments we have performed indicate the plausibility of this approach for database mapping and show that it can be applied to practical and clinical problems such as those encountered in face recognition and computer-aided diagnosis.

Model selection for the first time explicitly incorporates the bias/variance dilemma in finite data training, and when tested with synthetic and actual data, the results show that the number of hidden nodes should be adjusted for both data quantification and data classification, thus leading to a unified framework. At issue is how the model selection affects the estimation error and how the error in the estimation of class likelihoods further affects the classification error when the estimates are used in a classification rule. However, none of previously developed methods has directly addressed a goal of minimizing classification errors, which is a central objective of data classification. It is necessary, therefore, to develop methods that are more directly related to the minimization of classification errors. On the other hand, many previous researchers have shown that one of the most fundamental problems in detection and estimation is the bias/variance dilemma [25], [26], [30], [31]. It has been reported that the bias and variance components of the estimation error combine to influence classification in a very different way than with squared error on the likelihoods themselves [1], [25], [26]. Their results also suggested that the bias and variance components may not be treated in an equal base for further improving the classifier's performance [26], and a minimum entropy approach was proposed for model selection aiming at maximizing the class separability [1]. However, these methods may be found to be problematic when the accuracy of both data quantification and classification is required.

Further comparison of the data quantification to the data classification calls for the following pair-wise relationships in the learning paradigm (supervised and unsupervised) and in the implementing scheme (*soft* and *hard*). In fact, when data quantification is the objective, unsupervised learning is preferred where only a *soft* classification of the data is required [23]. More precisely, since maximum likelihood is the criterion, local cluster parameters can be learned without *hard* data classification [1], [12], [22], [24]. If this unsupervised process involves a *hard* classification of a sample into the cluster for which the posterior probability is maximum, such as in the k-means algorithm [22], the quantities obtained by the sample averages after data classification may not be consistent with the previous quantification result since a perfect classification may not be possible when the distributions of local clusters are highly overlapping [23]. The quantification result, in general, will be biased. On the other hand, in order to perform data classification for the testing set where the objective is to minimize the average Bayes' risk, supervision is needed at a first place and can be realized by simply dividing the training set (e.g., a subset of the testing set) into the groups for the estimation of each local class likelihood (e.g., unsupervised learning of local clusters), whereas the global class Bayesian prior can be picked up immediately as the by-product of the dividing process. In this research, we deal with data quantification for local clusters and data classification between classes as two separate problems and use different optimality criteria. However, it is worth reiteration that in order to efficiently determine the decision boundaries between classes in data classification, supervised and unsupervised training may be jointly performed.

Appendix
Collected Proofs of the Theorems

Proof of Theorem 1: Since the multiplication over i in joint likelihood is not affected by the data order, we regroup them in an increasing order of the gray levels u_l such that $u_1 < u_2, \cdots, < u_L$. Hence, we write

$$\mathcal{L}_r(\theta) = \prod_{i=1}^{N_r} f_r(x_i) = \prod_{l=1}^{L} \left(\prod_{x_i = u_l} f_r(x_i) \right). \tag{33}$$

By the definition of data histogram (i.e., the type) in [37], the number of data with gray level u_l equals $N_r f_{\mathbf{x}_r}(u_l)$; thus, we

have

$$\begin{aligned}\mathcal{L}_r(\theta) &= \prod_{l=1}^{L} f_r(u_l)^{N_r f_{\mathbf{x}_r}(u_l)} \\ &= \prod_{l=1}^{L} \exp\left(N_r f_{\mathbf{x}_r}(u_l) \log f_r(u_l)\right) \\ &= \prod_{l=1}^{L} \exp\left(N_r[f_{\mathbf{x}_r}(u_l) \log f_r(u_l) \right. \\ &\quad - f_{\mathbf{x}_r}(u_l) \log f_{\mathbf{x}_r}(u_l) \\ &\quad \left. + f_{\mathbf{x}_r}(u_l) \log f_{\mathbf{x}_r}(u_l)]\right) \\ &= \exp\left(-N_r \sum_{l=1}^{L} \left[f_{\mathbf{x}_r}(u_l) \log \frac{1}{f_{\mathbf{x}_r}(u_l)} \right.\right. \\ &\quad \left.\left. + f_{\mathbf{x}_r}(u_l) \log \frac{f_{\mathbf{x}_r}(u_l)}{f_r(u_l)}\right]\right) \\ &= \exp\left(-N_r[H(f_{\mathbf{x}_r}) + D(f_{\mathbf{x}_r}\|f_r)]\right). \end{aligned}$$ □

Proof of Theorem 2: For each data value u_l, we apply indicator function $I(\cdot, u_l)$ to data sequence $\mathbf{x}_r$. By the definition of histogram, we have the relationship between the histogram $f_{\mathbf{x}_r}(u_l)$ and the sample average of the indicator functions $I(x_i, u_l)$. Since sequence $\mathbf{x}$ is asymptotically independent and identically distributed by the finite normal mixture distribution, they are ergodic processes. In addition, since the indicator function is a deterministic measurable function, by the Birkhoff–Khinchin theorem [40]

$$\Pr\left(\lim_{N_r\to\infty} \frac{1}{N_r} \sum_{i=1}^{N_r} I(x_i, u_l) = E[I(x_i, u_l)]\right) = 1. \quad (34)$$

Since, by the fundamental theorem of expectation, we have

$$E[I(x_i, u_l)] = \sum_{u} I(x_i = u, u_l) f_r^*(u) = f_r^*(u_l) \quad (35)$$

we can substitute (3) and (9) into (8) to obtain

$$\Pr\left(\lim_{N_r\to\infty} f_{\mathbf{x}_r}(u_l) = f_r^*(u_l)\right) = 1$$

which implies that the distance of $D(f_{\mathbf{x}_r}\|f_r^*)$ goes to 0 as $N_r \to \infty$.

We now show that the estimated distribution f_r is close to f_r^* for large N_r in relative entropy. By the "Pythagorean" theorem ([37, Th. 12.6.1])

$$D(f_{\mathbf{x}_r}\|f_r) + D(f_r\|f_r^*) \le D(f_{\mathbf{x}_r}\|f_r^*) \quad (36)$$

which in turn implies that

$$D(f_r\|f_r^*) \le D(f_{\mathbf{x}_r}\|f_r^*) \quad (37)$$

since $D(f_{\mathbf{x}_r}\|f_r) \ge 0$. Note that the relative entropy $D(f_{\mathbf{x}_r}\|f_r^*)$ behaves like the square of the Euclidean distance [37]. From the conditions given by the theorem, the angle between the distances $D(f_{\mathbf{x}_r}\|f_r)$ and $D(f_r\|f_r^*)$ must be obtuse, which implies (36). Consequently, since $D(f_{\mathbf{x}_r}\|f_r^*) \to 0$, it follows that

$$\lim_{N_r\to\infty} D(f_r\|f_r^*) = 0 \quad (38)$$

as $N_r \to \infty$ with probability one. □

REFERENCES

[1] L. Perlovsky and M. McManus, "Maximum likelihood neural networks for sensor fusion and adaptive classification," *Neural Networks*, vol. 4, pp. 89–102, 1991.

[2] H. Gish, "A probabilistic approach to the understanding and training of neural network classifiers," in *Proc. IEEE Intl. Conf. Acoust., Speech, Signal Process.*, 1990, pp. 1361–1364.

[3] D. M. Titterington, A. F. M. Smith, and U. E. Markov, *Statistical Analysis of Finite Mixture Distributions.* New York: Wiley, 1985.

[4] Y. Wang, "Database mapping by mixture of experts in computer-aided diagnosis," Tech. Rep., Georgetown Univ. Med. Cent., Washington, DC, July 1996.

[5] S. Y. Kung and J. S. Taur, "Decision-based neural networks with signal/image classification applications," *IEEE Trans. Neural Networks*, vol. 1, pp. 170–181, Jan. 1995.

[6] S. H. Lin, S. Y. Kung, and L. J. Lin, "Face recognition/detection by probabilistic decision-based neural network," *IEEE Trans. Neural Networks, Special Issue on Artificial Neural Networks and Pattern Recognition*, vol. 8, Jan. 1997.

[7] H. Li *et al.*, "Detection of masses on mammograms using advanced segmentation techniques and an HMOE classifier," in *Proc. 3rd Int. Workshop Digital Mammography*, Chicago, IL, June 1996, pp. 397–400.

[8] P. Santago and H. D. Gage, "Quantification of MR brain images by mixture density and partial volume modeling," *IEEE Trans. Med. Imag.*, vol. 12, pp. 566–574, Sept. 1993.

[9] A. J. Worth and D. N. Kennedy, "Segmentation of magnetic resonance brain images using analog constraint satisfaction neural networks," *Inform. Process. Med. Imag.*, pp. 225–243, 1993.

[10] D. P. Helmbold, R. E. Schapire, Y. Singer, and M. K. Warmuth, "A comparison of new and old algorithms for a mixture estimation problem," Tech. Rep., Univ. Calif., Santa Cruz and AT&T Lab., 1996.

[11] E. Weinstein, M. Feder, and A. V. Oppenheim, "Sequential algorithms for parameter estimation based on the Kullback–Leibler information measure," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 1652–1654, Sept. 1990.

[12] Y. Wang and T. Adali, "Efficient learning of finite normal mixtures for image quantification," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Atlanta, GA, 1996, pp. 3422–3425.

[13] F. S. Samaria and A. C. Harter, "Parameterization of a stochastic model for human face identification," in *Proc. IEEE Workshop Appl. Comput. Vision*, Sarasota, FL, 1994.

[14] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back, "Face recognition: A convolutional neural network approach," Tech. Rep., NEC Res. Inst., 1995.

[15] F. S. Saramia, "Face recognition using hidden markov model," Ph.D. dissertation, Univ. Cambridge, Cambridge, U.K., 1994.

[16] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cogn. Neurosci.*, vol. 3, pp. 71–86, 1991.

[17] M. I. Jordan and R. A. Jacobs, "Hierarchical mixture of experts and the EM algorithm," *Neural Comput.*, vol. 6, pp. 181–214, 1994.

[18] C. E. Priebe, "Adaptive mixtures," *J. Amer. Stat. Assoc.*, vol. 89, no. 427, pp. 910–912, 1994.

[19] R. A. Redner and N. M. Walker, "Mixture densities, maximum likelihood and the EM algorithm," *SIAM Rev.*, vol. 26, pp. 195–239, 1984.

[20] R. M. Neal and G. E. Hinton, "A new view of the EM algorithm that justifies incremental and other variants," *Biometrika*, 1993.

[21] L. Xu and M. I. Jordan, "On convergence properties of the EM algorithm for Gaussian mixture," Tech. Rep., Artif. Intell. Lab., Mass. Inst. Technol, Cambridge, Jan. 1995.

[22] J. L. Marroquin and F. Girosi, "Some extensions of the K-means algorithm for image segmentation and pattern classification," Tech. Rep., Artif. Intell. Lab., Mass. Inst. Technol, Cambridge, Jan. 1993.

[23] D. M. Titterington, "Comments on 'application of the conditional population-mixture model to image segmentation,'" *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, pp. 656–658, Sept. 1984.

[24] Y. Wang and T. Adali, "Probabilistic neural networks for parameter quantification in medical image analysis," *Biomed. Eng. Recent Development*, 1994.

[25] J. L. Marroquin, "Measure fields for function approximation," *IEEE Trans. Neural Networks*, vol. 6, pp. 1081–1090, May 1995.
[26] J. H. Friedman, "On bias, variance, 0/1–loss, and the curse-of-dimensionality," Tech. Rep., Stanford Univ., Stanford, CA, 1996.
[27] H. Akaike, "A new look at the statistical model identification," *IEEE Trans. Automat. Contr.*, vol. AC-19, Dec. 1974.
[28] J. Rissanen, "A universal prior for integers and estimation by minimum description length," *Ann. Stat.*, vol. 11, no. 2, 1983.
[29] E. T. Jaynes, "Information theory and statistical mechanics," *Phys. Rev.*, vol. 108, no. 2, pp. 620–630/171–190, May 1957.
[30] J. Rissanen, "Minimax entropy estimation of models for vector processes," *Syst. Identification: Advances Case Studies*, pp. 97–119, 1987.
[31] S. Geman, E. Bienenstock, and R. Doursat, "Neural networks and the bias/variance dilemma," *Neural Comput.*, vol. 4, pp. 1–52, 1992.
[32] Y. Wang, "Image quantification and the minimum conditional bias/variance criterion," in *Proc. 30th Conf. Inform. Sci. Syst.*, Princeton, NJ, Mar. 20–22, 1996, pp. 1061–1064.
[33] L. I. Perlovsky, "Cramer–Rao bounds for the estimation of normal mixtures," *Pattern Recognit. Lett.*, vol. 10, pp. 141–148, 1989.
[34] J. Zhang and J. M. Modestino, "A model-fitting approach to cluster validation with application to stochastic model-based image segmentation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, pp. 1009–1017, Oct. 1990.
[35] R. A. Jacobs, "Increased rates of convergence through learning rate adaptation," *Neural Networks*, vol. 1, pp. 295–307, 1988.
[36] S. Haykin, *Neural Networks: A Comprehensive Foundation.* New York: MacMillan, 1994.
[37] T. M. Cover and J. A. Thomas, *Elements of Information Theory.* New York: Wiley, 1991.
[38] H. V. Poor, *An Introduction to Signal Detection and Estimation.* New York: Springer-Verlag, 1988.
[39] A. S. Pandya and R. B. Macy, *Pattern Recognition with Neural Networks in C++.* Boca Raton, FL: CRC, 1996.
[40] R. Gray and L. Davisson, *Random Processes—A Mathematical Approach for Engineers.* Englewood Cliffs, NJ: Prentice-Hall, 1986.
[41] M. J. Bianchi, A. Rios, and M. Kabuka, "An algorithm for detection of masses, skin contours, and enhancement of microcalcifications in mammograms," in *Proc. Comput.-Assisted Radiol.*, Winston-Salem, NC, June 1994, pp. 57–64.

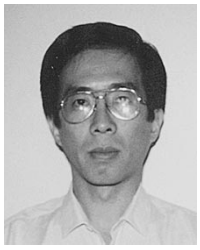
Yue Wang received the B.S. and M.S. degrees from Shanghai Jiao Tong University, Shanghai, China, in 1984 and 1987, respectively, and the Ph.D. degree from the University of Maryland, Baltimore County, Baltimore, in 1995, all in electrical engineering.

He is currently with the Department of Electrical Engineering and Computer Science, Catholic University of America, Washington, DC, as an Assistant Professor. He is also affiliated with the Department of Radiology, Georgetown University School of Medicine, Washington, DC, as an Adjunct Assistant Professor. His research interests include image analysis, medical imaging, information visualization, database mapping, volumetric display, visual explanation, and their applications in biomedicine and multimedia informatics.

Dr. Wang is the recipient of a 1998 U.S. Army Medical Research Command Career Development Award.

Shang-Hung Lin received the B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, R.O.C., in 1991. He received the M.S. and Ph.D. degrees in electrical engineering from Princeton University, Princeton, NJ, in 1994 and 1996, respectively.

He is currently with Epson Palo Alto Laboratory, Palo Alto, CA. His primary research interests include neural networks, pattern recognition, computer vision, and image processing.



Huai Li received the B.S. degree in engineering physics from Tsinghua University, Beijing, China, in 1985, the M.Med. degree in biomedical engineering from Beijing Medical University in 1988, and the M.S. and Ph.D. degrees in electrical engineering from University of Maryland, College Park, in 1995 and 1997, respectively.

Since 1997, he has been with the multimedia team at Odyssey Technologies, Inc., Jessup, MD, and is currently a Member of the Technical Staff, working on image/video processing and telecommunication. His research interests include medical image analysis, image processing, video coding, and telecommunication.

Sun-Yuan Kung (F'88) received the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA.

In 1974, he was an Associate Engineer of Amdahl Corporation, Sunnyvale, CA. From 1977 to 1987, he was a Professor of Electrical Engineering Systems, University of Southern California, Los Angeles. Since 1987, he has been a Professor of Electrical Engineering at Princeton University, Princeton, NJ. He has authored more than 300 technical publications, including three books: *VLSI Array Processors* (Englewood Cliffs, NJ: Prentice-Hall, 1988) (with Russian and Chinese translations), *Digital Neural Networks* (Englewood Cliffs, NJ: Prentice-Hall, 1993), and *Principal Component Neural Networks* (New York: Wiley, 1996).

Dr. Kung received the 1992 IEEE Signal Processing Society's Technical Achievement Award for his contributions on parallel processing and neural network algorithms for signal processing. Since 1990, he has served as Editor-in-Chief of the *Journal of VLSI Signal Processing*. Recently, he served as a General Chair of the 1997 IEEE Workshop on Multimedia Signal Processing at Princeton University. He was appointed an IEEE-SP Distinguished Lecturer in 1994. He received the 1996 IEEE Signal Processing Society's Best Paper Award.