A Multi-scale Nonparametric/Parametric Hybrid Recognition Strategy with Multi-category Posterior Probability Estimation

Zhao Lu[1], **Zheng Lu**[2], **and Haoda Fu**[3]
[1]Department of Electrical Engineering, Tuskegee University, Tuskegee, AL, USA
[2]Astella Pharma Global Development, Inc., Northbrook, IL, USA
[3]Eli Lilly and Company, Indianapolis, IN USA

Abstract — *The synthesis of an effective multi-category nonlinear classifier with the capability to output calibrated posterior probabilities to enable post-processing is of great significance in practical recognition situations in that the posterior probability reflects the assessment uncertainty. In this paper, a multi-scale nonparametric and parametric hybrid recognition strategy is developed for this purpose. Based on the binary tree representation for nested structure, a new nonlinear polychotomous classification algorithm with the capability of estimating posterior probability is developed on the strength of kernel learning and Bayesian decision theory. In particular, by capitalizing on the intrinsic conexus between hierarchical structure and multi-scale analysis, the polychotomous multi-scale Bayesian kernel Fisher discriminant is implemented for building the classifier at different scales for different levels. Finally, the performance of the proposed classification and posterior probability estimation algorithm is validated by designing a multi-category Bayesian kernel Fisher discriminant classifier for a satellite images dataset.*

Keywords: Kernel Fisher Discriminant; Binary Tree; Posterior Probability; Inter-class Separability; Class-conditional density function; Multi-scale.

1 Introduction

In the realm of pattern recognition and statistical learning, most of existing schemes can be categorized into parametric or nonparametric approaches. Parametric methods assume specific parametric models, while nonparametric methods usually do not require any postulations for the model and utilize the sampled data directly for model representation. Both parametric and nonparametric methods have their own strengths and limitations [1], and the complementarity between them has aroused considerable research endeavours in fusing non-parametric and parametric methods for targets tracking, nonlinear systems identification, classifier construction and modeling, etc [1−6]. In this paper, as a stride towards the fusion of kernel-based nonparametric computational learning methods and parametric density estimation methods, a multi-scale multi-class recognition strategy is developed, where the kernel Fisher discriminant (KFD) is employed for feature extraction and parametric class-conditional density estimation is used for Bayesian classification.

In real world, most of classification problems encountered comprise multiple categories, i.e., polychotomous classification problem, such as automatic target recognition, optical character recognition, face recognition, etc. In general, the issue of polychotomous classification is much more involved than dichotomic classification. With the burgeoning of various kernel learning algorithms since 1990s [7−9], such as support vector machine (SVM), kernel Fisher discriminant (KFD) and kernel principal component analysis (KPCA) and so on, the synthesis of multi-category nonlinear kernel classifier with superior generalization capability has become a focus of research in the past decade [10−16]. The conventional approaches for extending binary classifier to polychotomous classifier fall into two categories, i.e., the direct method and 'divide-and-combine' approach. The direct method is a straightforward generalization of the corresponding dichotomic algorithms, and all data are considered in one optimization formulation, which may result in prohibitively-expensive computing cost for solving a nonlinear optimization problem with a large number of variables.

In contrast to the direct method, the methodology of 'divide-and-combine' usually decomposes the multi-category problem into several subproblems that can be solved by using binary classifiers. Two widely used 'divide-and-combine' methods are pairwise and one-versus-rest. In the approach of pairwise, an n-class problem is converted into $n(n\text{-}1)/2$ dichotomic problems which cover all pairs of classes. Then, the binary classifiers are trained for each of pairs, and the classification decision for a test pattern is given on the aggregate of output magnitudes. Apparently, in pairwise methods, the number of binary classifiers built increases rapidly with the increasing of the number of classes, which easily leads to onerous computational task. This problem is alleviated in the one-versus-rest method, where only n binary classifiers are needed for n-class problem and each of them is trained to separate one class of samples from all others.

However, all training data have to be involved in constructing each binary classifier and one-versus-rest method is not capable to yield the optimal decision boundaries. In particular, both methods can result in the existence of unclassified regions.

Recently, as a new member in the family of 'divide-and-combine' methods, the multi-category classifier with hierarchical tree structure has aroused extensive interest in the community of pattern recognition and machine learning [16−19]. As a natural hierarchical representation for nested structure, the binary tree usually organizes information into different levels, which enables the multi-scale implementation so that the higher in the hierarchy a level is the finer scales the information is processed in.

Moreover, compared to the conventional approaches in constructing the multi-category classifiers, the polychotomous classifiers with hierarchical structure are advantageous in improving computational tractability and classification accuracy, diminishing the amount of data involved in training each binary classifier and eliminating unclassifiable regions. Also, the hierarchical structure invoked empowers the design and implementation of multi-scale polychotomous classification algorithms to take care of local as well as global complexity of the input-output map. For constructing the hierarchical tree structure, non-metric distance functions for measuring the inter-class separability was developed in Refs. [17−18, 20]. The significance of no-metric distance function in image classification and computer vision has been investigated in [21], and the raison d'etre of non-metric distance function is also corroborated by some research in psychology suggesting the ubiquity of non-metric distance in human similarity judgments [22].

On the other hand, the synthesis of a multi-category nonlinear classifier with the capability to produce a calibrated posterior probability $P(class|input)$ to enable post-processing is of great significance in practical recognition situations. For instance, a posterior probability allows decisions that can use a utility model. Posterior probabilities are also required when a classifier is making a small part of an overall decision, and the classification outputs must be combined with other sources of information for decision-making, such as example-dependent misclassification costs, the outputs of other classifiers or domain knowledge [23−25]. For the nonlinear kernel classification algorithms, albeit some endeavours have been devoted to convert the output of support vector classifier into the posterior probability by fitting some predefined mapping functions [23−27], such as logistic link function and sigmoid function, these schemes are empirical per se and the building of classifier is irrespective of the estimation of posterior probability.

Compared to the algorithm of support vector classification, which directly generates geometric decision boundary for dichotomy with an uncalibrated value, a crucial advantage of the KFD is that the produced outputs can easily be transformed into the posterior probabilities, i.e., the class membership. In other words, the output values imply not only whether a given test pattern belongs to a certain class, but also the probability of this event [7, 28]. Some recent researches have revealed the essence of KFD in nonlinear classification [29] and the equivalence between linear SVC and sparsified Fisher discriminant analysis [30]. Although the algorithm of Fisher discriminant can be generalized to $n-$class feature extraction and dimension reduction problem by directly projecting the data onto a $(n-1)$ dimensional space [31], this direct method is obviously unable to be used when the number of classes is greater than the dimensionality of the input space. While, for the algorithm of polychotomous KFD developed in Ref. [17], it can be used for multi-category problem regardless of the dimensionality of the input space, and in particular the hierarchical tree structure synthesized provides a natural framework for evaluating the multi-class posterior probabilities. Herein, in the line of our previous arguments [17−18], the problem of evaluating multi-class posterior probability is approached by an innovative multi-scale polychotomous Bayesian kernel Fisher discriminant algorithm developed in this paper. The proposed algorithm primarily rests on two pillars: class-conditional density function estimation and binary tree representation for nested structure. The former enables the evaluation of posterior probability for the dichotomic subproblems, and the latter empower us to convert the multi-category classification problem into $(n-1)$ dichotomic subproblems and thereby implement the multi-scale classification.

The rest of this paper is organized as follows. In the next section, the kernelized group clustering algorithm used in [17] for binary tree induction is briefly reviewed. Following that, the polychotomous Bayesian KFD on the strength of Lindeberg-Feller central limit theorem is discussed in Section 3. In Section 4, the algorithms for estimating class conditional probability densities and multi-class posterior probability are presented. The simulation study on satellite image data classification is conducted in Section 5, with concluding remarks in Section 6.

The following generic notations will be used throughout this paper: non-boldface symbols such as $y, k, P, \cdots$ refer to scalar valued objects, lower case boldface symbols such as $\boldsymbol{x}, \boldsymbol{\varphi}, \boldsymbol{\beta}, \cdots$ refer to vector valued objects, and capital boldface symbols such as $\boldsymbol{N}, \boldsymbol{K}, \boldsymbol{A}, \cdots$ will be used for matrices and sets.

2 Macro-class partition algorithm for binary tree synthesis

The strategy of determining the topology of binary tree by dividing the multiple classes to be recognized into two smaller macro-classes at each non-leaf node has been developed in Refs. [17−18]. Apparently, there exist many possibilities to split the multiple classes into two smaller macro-classes; hence the macro-class partitioning algorithm plays a vital role in the success of this strategy. Albeit the hierarchical divisive clustering method may be a natural choice for macro-class partitioning [32], the challenge is posed for defining the appropriate distance function capable of measuring the inter-

class separability in feature space for clustering classes in the scenario of nonlinear classification.

In Ref. [20], the sum of minimum distances function d_{md} was proposed for measuring the inter-class separability

$$d_{md}(\boldsymbol{A},\boldsymbol{B})=\frac{1}{2}\left(\sum_{\boldsymbol{a}_i\in A}\min_{\boldsymbol{b}_j\in B}\left\|\boldsymbol{a}_i-\boldsymbol{b}_j\right\|+\sum_{\boldsymbol{b}_j\in B}\min_{\boldsymbol{a}_i\in A}\left\|\boldsymbol{a}_i-\boldsymbol{b}_j\right\|\right) \quad (1)$$

where $\boldsymbol{A}=\{\boldsymbol{a}_i \mid i=1,2,\cdots,p\}$ and $\boldsymbol{B}=\{\boldsymbol{b}_i \mid i=1,2,\cdots,q\}$ are training datasets of two different classes. Compared to the well-known Hausdorff metric, which is defined as the maximum distance between any point in one shape and the point that is closest to it in the other, the distances function d_{md} defined by (1) is non-metric and advantageous due to its capability of taking into account the overall structure of the points set. Further, for measuring inter-class separability in the feature space induced by nonlinear mapping $\boldsymbol{\varphi}(\cdot)$, the sum of minimum distance function was kernelized to the following form in Ref. [17−18]

$$\tilde{d}_{md}(\boldsymbol{A},\boldsymbol{B})=\frac{1}{2}\left(\sum_{\boldsymbol{a}_i\in\boldsymbol{A}}\min_{\boldsymbol{b}_j\in\boldsymbol{B}}\sqrt{2-2k(\boldsymbol{a}_i,\boldsymbol{b}_j)}+\sum_{\boldsymbol{b}_j\in\boldsymbol{B}}\min_{\boldsymbol{a}_i\in\boldsymbol{A}}\sqrt{2-2k(\boldsymbol{a}_i,\boldsymbol{b}_j)}\right) \quad (2)$$

where $k(\boldsymbol{a}_i,\boldsymbol{b}_i)=\boldsymbol{\varphi}(\boldsymbol{a}_i)^T\boldsymbol{\varphi}(\boldsymbol{b}_i)$ is the kernel function such that $k(\boldsymbol{x},\boldsymbol{x})=1$, and obviously the kernelized distance function $\tilde{d}_{md}$ in (2) can be evaluated without explicitly knowing the nonlinear mapping $\boldsymbol{\varphi}(\cdot)$.

Before training the dichotomic classifiers at non-leaf nodes, the topology of the binary tree needs to be determined firstly by partitioning the classes to be recognized into two smaller macro-classes at each non-leaf node from top to down. This procedure specifies the training datasets used for training each binary classifier and therefore is critical to the recognition performance of the hierarchical classification algorithm.

In the hierarchical classification algorithm, it is obvious that the degeneration of classification performance at higher level has greater impact on the overall classification performance than that occurred at lower levels. Therefore, the upper level the more separable classes should be partitioned, i.e., maximizes the degree of separability while partitioning the multiple classes into two macro-classes from top to down.

With the kernelized distance function $\tilde{d}_{md}$ for measuring the inter-class separability, the macro-class partition algorithm implemented by invoking the hierarchical divisive clustering can be applied for each non-leaf node from top to down, where the classes in one macro-class are recursively divided into two macro-classes belonging to left-node and right-node respectively. Initially, the macro-class partition algorithm starts from the root node, where the macro-class includes all classes to be recognized. Firstly, the kernelized sum of minimum distance function $\tilde{d}_{md}$ between all pairs of the classes in one macro-class are evaluated, and then partition the pair of classes between which the distance is maximal into the left-node and right-node as the prototype classes of the child nodes, respectively. Subsequently, assign the remaining classes in the non-leaf node into the child node whose prototype class is the closest to it in the sense of kernelized distance function $\tilde{d}_{md}$. Thus, two smaller macro-classes, either of which may also consist of multiple classes, are formed in the left child node and right child node, respectively. Iterating this procedure from top to down for every non-leaf node until only one individual class is left in each leaf node produces a hierarchy of nested macro-classes, and thereby determines the topology of the binary tree. Apparently the number of leaf nodes equals to the number of classes.

3 Estimation of class-conditional PDF of projected data in kernel feature space

The binary tree synthesized via macro-class partition algorithm offers a skeleton where the dichotomic classifier can be trained at each non-leaf node for implementing a decision rule that separates the macro-class into its left child node and its right child node. Thus, the n-class polychotomous classifier can be constructed by training $(n-1)$ binary classifier at non-leaf nodes, which is less than the number of dichotomic classifiers trained in pairwise and one-versus-rest methods. Also, as learning proceeds from top to down, the amount of data involved in the subsequent training processes decrease rapidly. These substantially improve the computational tractability. In this section, following a briefly review for KFD algorithm, the estimation of the underlying class-conditional PDF for the projections generated via KFD in feature space will be discussed.

Given a set of $m-$dimensional input vectors $\boldsymbol{x}_j$, $j=1,\cdots,\ell$, ℓ_1 input vectors in the subset $\boldsymbol{D}_1$ labeled ω_1 and ℓ_2 input vectors in the subset $\boldsymbol{D}_2$ labeled ω_2. In the algorithm of KFD, the generalized Rayleigh quotient is maximized in the feature space in order to find the projection direction $\boldsymbol{w}$ which maximizes the between-class variance and minimizes the within-class variance for the projections on it. In feature space, the generalized Rayleigh quotient becomes

$$J(\boldsymbol{w})=\frac{\boldsymbol{w}^T\boldsymbol{S}_B\boldsymbol{w}}{\boldsymbol{w}^T\boldsymbol{S}_W\boldsymbol{w}}, \quad (3)$$

where

$$\boldsymbol{S}_B=(\boldsymbol{m}_2-\boldsymbol{m}_1)(\boldsymbol{m}_2-\boldsymbol{m}_1)^T,$$

$$\boldsymbol{S}_W=\sum_{i=1}^{2}\sum_{\boldsymbol{x}_j\in D_i}(\boldsymbol{\varphi}(\boldsymbol{x}_j)-\boldsymbol{m}_i)(\boldsymbol{\varphi}(\boldsymbol{x}_j)-\boldsymbol{m}_i)^T,$$

$$\boldsymbol{m}_i=\frac{1}{\ell_i}\sum_{\boldsymbol{x}_j\in\boldsymbol{D}_i}\boldsymbol{\varphi}(\boldsymbol{x}_j).$$

Define the matrices $\boldsymbol{N}$ and $\boldsymbol{M}$ as follows

$$\boldsymbol{N}=(\boldsymbol{\vartheta}_2-\boldsymbol{\vartheta}_1)(\boldsymbol{\vartheta}_2-\boldsymbol{\vartheta}_1)^T,$$

$$\boldsymbol{M} = \sum_{i=1}^{2} \left[\boldsymbol{K}_i \boldsymbol{K}_i^T - \ell_i \boldsymbol{\vartheta}_i \boldsymbol{\vartheta}_i^T \right]$$

where $\boldsymbol{\vartheta}_i$ is the ℓ – dimensional column vector with components

$$(\boldsymbol{\vartheta}_i)_r = \sum_{\boldsymbol{x}_j \in \boldsymbol{D}_i} \frac{k(\boldsymbol{x}_r, \boldsymbol{x}_j)}{\ell_i}$$

and $\boldsymbol{K}_i$ are the kernel matrices with entries $(\boldsymbol{K}_i)_{rj} = k(\boldsymbol{x}_r, \boldsymbol{x}_j)$. With the vital ansatz that $\boldsymbol{w} = \sum_{j=1}^{\ell} \beta_j \boldsymbol{\varphi}(\boldsymbol{x}_j)$, the generalized Rayleigh quotient (3) can be reformulated in terms of kernel function in the feature space as [33]

$$J(\boldsymbol{\beta}) = \frac{\boldsymbol{\beta}^T \boldsymbol{N} \boldsymbol{\beta}}{\boldsymbol{\beta}^T \boldsymbol{M} \boldsymbol{\beta}}. \tag{4}$$

The expansion coefficients vector $\boldsymbol{\beta}$ can be obtained by maximizing the $J(\boldsymbol{\beta})$ in (4), and several effective algorithms for that have been available and discussed in [7]. Thereby, the projections of the mapped data points $\boldsymbol{\varphi}(\boldsymbol{x}_j)$ onto the discriminant $\boldsymbol{w}$ in feature space can be calculated as

$$y = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}) = \sum_{j=1}^{\ell} \beta_j \boldsymbol{\varphi}^T(\boldsymbol{x}_j) \boldsymbol{\varphi}(\boldsymbol{x}) = \sum_{j=1}^{\ell} \beta_j k(\boldsymbol{x}_j, \boldsymbol{x}). \tag{5}$$

From equation (5), it is reasonable to treat the projection $y = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x})$ as a scalar random variable, which is the weighted summation of all components of the data points $\boldsymbol{\varphi}(\boldsymbol{x})$ mapped into the high-dimensional feature space. It is noteworthy that the feature spaces induced by kernel functions are usually very high-dimensional, and for instance, the dimension of the feature space induced by Gaussian RBF kernel is infinite. Hence, according to the celebrated Lindeberg-Feller Central Limit Theorem, this fact implies that the set of projections y of the mapped data in each class tends to be distributed normally, i.e.

$$p(y \mid \omega_i) \sim N(\mu_i, \sigma_i) \tag{6}$$

where

$$N(\mu_i, \sigma_i) = \frac{1}{(2\pi\sigma_i^2)^{1/2}} exp\left\{ -\frac{(y-\mu_i)^2}{2\sigma_i^2} \right\},$$

is the univariate Gaussian probability density function. Thus, the estimation of the class-conditional density function $p(y \mid \omega_i)$ for the projections y_j is boiled down to the issue of estimating the parameters μ_i, σ_i of Gaussian PDFs, which can be readily solved by the methods of maximum likelihood or Bayesian inference. In this paper, the method of maximal likelihood estimation is exerted for calculating the parameters of class-conditional Gaussian PDF, and the details of maximal likelihood estimation algorithm can be referred to [31−32]. The availability of class-conditional density functions makes it possible to build the classifier upon the Bayesian decision theory, which is a fundamental statistical approach, whose power, coherence, and analytical nature when applied in pattern recognition make it among the elegant formulations in science.

4 Multi-category posterior probability estimation & multi-scale discriminant

With the estimated class-conditional Gaussian density functions $p(y \mid \omega_i)$, $i = 1, 2$ the two-class posterior probability can be evaluated at each non-leaf node

$$P(\omega_i \mid y) = \frac{p(y \mid \omega_i) P(\omega_i)}{\sum_{i=1}^{2} p(y \mid \omega_i) P(\omega_i)}, \quad i = 1, 2 \tag{7}$$

where $P(\omega_i)$ is the priori probability, which can be estimated from the training dataset empirically, and the denominator is the unconditional probability density function. Thereby, the dichotomic Bayesian classifier can be constructed at each non-leaf node by selecting the class ω_i having the largest posterior probability, so that $\boldsymbol{x}$ is assigned to class ω_i if

$$P(\omega_i \mid y) > P(\omega_k \mid y) \quad for\ all\ \ i \neq k \tag{8}$$

where y is the projections of $\boldsymbol{x}$ onto the discriminant $\boldsymbol{w}$ in the feature space. A Bayesian approach achieves the exact minimum probability of error based entirely on evaluating the posterior probability.

For classifying an unlabeled pattern, the evaluation starts from the root node of the binary tree, and then from top to down the synthesized dichotomic classifiers on the non-leaf nodes is used to assign the input pattern into one of child nodes. This procedure is iterated until the unlabeled pattern is finally classified into the class associated with one of leaf nodes, which determine a path from the root to one of leaf-nodes for each unlabeled pattern. Contrary to the conventional 'divide-and-combine' methods where all the dichotomic decision functions need to be calculated in evaluating an unlabeled pattern, only those dichotomic decision functions on the specified path need to be calculated in the proposed method.

In the realm of pattern recognition, there is general consensus that one of important technical challenges is how to estimate the multi-class posterior probability, which is more unwieldy than that for dichotomic classifier. However, in the algorithm developed in this paper, the multi-class posterior probabilistic outputs can be readily evaluated by capitalizing on the posterior probability estimated in (7) at each non-leaf node of the synthesized binary-tree. For the path along which an unlabeled pattern was classified from the root to one of the leaf nodes, each trained dichotomous Bayesian KFD on the path outputs the posterior probability, which is used to determine which child node the unlabeled pattern should be

assigned to. Given that the path is determined by a sequence of dichotomous KFD successively, the posterior probability of classifying the unlabeled pattern into one of the multiple classes can be calculated by multiplying the posterior probabilistic outputs produced by each dichotomous KFD on the path. Contrary to the conventional methods, in which the values for all the decision functions need to be calculated in the phase of classification, it is not necessary to calculate the values of all the decision functions in the proposed method.

On the other hand, by taking advantage of the monotonicity of natural logarithm, the discriminant function induced by rule (8) on each non-leaf node can be expressed as

$$\begin{aligned} f(y) &= \ell n\, P(\omega_1 \mid y) - \ell n\, P(\omega_2 \mid y) \\ &= \ell n\, \frac{p(y \mid \omega_1)}{p(y \mid \omega_2)} + \ell n\, \frac{P(\omega_1)}{P(\omega_2)} \end{aligned} \tag{9}$$

Substituting the Gaussian density functions into $p(y \mid \omega_1)$ and $p(y \mid \omega_2)$ yields

$$f(y) = \frac{1}{2}\left[\frac{(y-\mu_2)^2}{\sigma_2^2} - \frac{(y-\mu_1)^2}{\sigma_1^2} \right] + ln\, \frac{\sigma_2}{\sigma_1} + ln\, \frac{P(\omega_1)}{P(\omega_2)} \tag{10}$$

If the variances for the macro-classes on the non-leaf node are equal, viz. $\sigma_1 = \sigma_2 = \sigma$, the equations (10) becomes

$$f(y) = \frac{\mu_1 - \mu_2}{\sigma^2} y + \frac{\mu_2^2 - \mu_1^2}{2\sigma^2} + ln\, \frac{P(\omega_1)}{P(\omega_2)} \tag{11}$$

The decision function $f(\boldsymbol{x})$ for data point $\boldsymbol{x}$ on each non-leaf node can be obtained by plugging equation (5) into the equation above as follows

$$f(\boldsymbol{x}) = \frac{\mu_1 - \mu_2}{\sigma^2} \sum_{r=1}^{\ell} \beta_r k(\boldsymbol{x}_r, \boldsymbol{x}) + C \tag{12}$$

where the constant

$$C = \frac{\mu_2^2 - \mu_1^2}{2\sigma^2} + ln\, \frac{P(\omega_1)}{P(\omega_2)} \tag{13}$$

Hence, in this case the discriminant function can be represented in the form of kernel expansion (12), which is same as that in support vector learning. Whereas, for the case that the variances for the macro-classes on the non-leaf node are not same, viz. $\sigma_1 \neq \sigma_2$, the expression of discriminant function becomes more involved than (12), and it is no longer as simple as the linear combination of kernel functions.

The hierarchical structure of binary tree together with the kernel expansion (12) also shed light on the avenue to fulfill the polychotomous multi-scale Bayesian kernel Fisher discriminant. Hierarchical structures organize information into different levels and usually arrange it so that the higher in the hierarchy a level is, the smaller scale the information is analyzed. In the algorithm developed in this paper, the degree of separability between macro-classes on the non-leaf nodes of the binary tree decrease from top to down, and the synthesis of polychotomous classifier can be viewed as a mathematical process of hierarchically building classifier such that finer details are added to the coarser description at each level. This intrinsic conexus between hierarchical structure and multi-scale analysis sheds lights on the way to implement the polychotomous multi-scale Bayesian KFD via setting different kernel parameters on different levels of the tree. For the Gaussian RBF kernel used in this research

$$k(\boldsymbol{x}, \boldsymbol{y}) = exp\left(\frac{-\|\boldsymbol{x} - \boldsymbol{y}\|^2}{2\rho_n^2} \right) \tag{14}$$

The values of parameter ρ_n can be set as $\rho_1 > \rho_2 > \cdots > \rho_m$ at different levels n from top to down for controlling the scales. With the declining of the degree of separability between macro-classes from top to down, the scale parameter also decrease gradually. The larger scale parameters are adopted for the lower levels to prevent memorizing data, and the smaller scale parameters are employed for the higher levels for irregular localized features. In Ref. [34], two schemes, which use geometric sequence and arithmetic sequence respectively, have been invoked to adjust the scale parameters ρ_n for non-flat function regression.

5 Landsat satellite image data classification

The goal of image classification is to separate images according to their visual content into two or more disjoint classes [35]. In this section, the developed multi-scale parametric/nonparametric hybrid recognition strategy and multi-class posterior probability estimation algorithm are applied on the recognition of satellite image data [36], which is a benchmark problem from real-world and has been intensively studied. The experimental result is compared with those acquired from other popular multi-class pattern classification methods in terms of the generalization capability. The implementation of algorithms is on the strength of the *Statistical Pattern Recognition Toolbox* [37]. For the sake of fair comparison, the same training and validation datasets as those in Ref. [36] are used.

The satellite image database was generated by taking a small section from the original Landsat Multi-Spectral Scanner (MSS) image data from a part of Western Australia. In this database, each sample was featured by 36 attributes, which are numerical in the range 0 to 255. Namely, the input space is of 36 dimensions. Totally, 4435 samples are included in the training dataset and 2000 samples in the validation dataset. There are six categories of different soil conditions to be classified, and their distributions in the training and validation dataset are listed in Table 1.

For synthesizing the proposed polychotomous multi-scale Bayesian KFD classifier, the value and tuning scheme of scale parameter of the adopted kernel function need to be specified beforehand. In our experiment, the Gaussian radial basis function kernel with scale parameter $\rho_1 = 33$ is used at the root node, and subsequently the scale parameter is tuned

as $\rho_{n+1} = \rho_n - \delta$, where δ is the common difference of the arithmetic sequence and n is the level of the hierarchical binary tree (root node is at the lowest level, i.e. level 1). The first step towards building the multi-class classifier is to induce the topology of the binary tree by taking advantage of macro-class partition algorithm described in section 2. For satellite image training database used herein, the topological structure of binary tree obtained via top-to-down induction is visualized in Fig. 1.

Upon determining the structure of the binary and the macro-classes on each non-leaf node, the algorithms developed in sections 3&4 can be brought to bear for training the dichotomic classifier at each non-leaf node and estimating the posterior probability.

To confirm the superiority of the proposed polychotomous multi-scale Bayesian KFD algorithm in terms of generalization capability, the testing error rate is calculated on the validation datasets, and then compared with those obtained from other popular classification strategies [36], such as Logistic regression, RBF neural networks, *K*-nearest-neighbor and multi-category SVM direct method [10], and so on. The results are listed in Table 2 and the details about the parameters setting and algorithmic implementation can be referred to the references [18,36]. From the test error rates in Table 2, it is salient that the polychotomous multi-scale Bayesian KFD excels other commonly-used pattern classification methods, including multi-class SVMs, in generalization capability. Also, for the Bayesian classification algorithms, the superiority in classification accuracy also implies the triumph in estimating the posterior probability. The uniqueness of path from root node to one leaf node enables us to calculate the multi-category posterior probability by multiplying the posterior probabilistic outputs produced by each dichotomous KFD on the path.

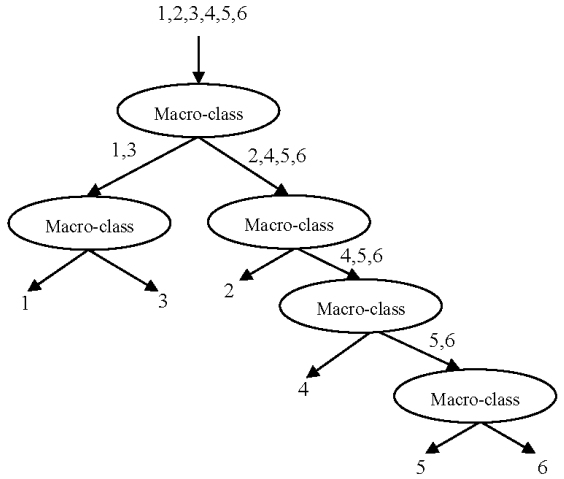

Fig. 1. Binary tree induced for Landsat satellite image datasets.

TABLE I
DISTRIBUTION OF TRAINING AND VALIDATION SAMPLES IN DATASET

Description	Training	Validation
1 red soil	1072(24.17%)	461 (23.05%)
2 cotton crop	479 (10.8%)	224 (11.20 %)
3 grey soil	961 (21.67%)	397 (19.85%)
4 damp grey soil	415 (9.36%)	211 (10.55%)
5 soil with vegetation stubble	470 (10.6%)	237 (11.85%)
6 very damp grey soil	1038 (23.4%)	470 (23.50%)

TABLE II
COMPARISON ON TESTING ERROR RATES OF VARIOUS ALGORITHMS

Pattern classification algorithms	Testing error rate (%)
Logistic discrimination	16.9
Quadratic discrimination	15.5
RBF neural networks	12.1
K-nearest-neighbor	9.4
Pairwise multi-class SVM	9.2
One-versus-rest multi-class SVM	9.65
Direct multi-class SVM	9.15
Method proposed in this article	8.55

6 Conclusions

The fact that the outputs produced by KFD can be interpreted as probabilities makes it possible to assign a confidence to the final classification. Based on this fact, in the polychotomous multi-scale classification algorithm developed in this paper, several key components are elegantly synergized together for synthesizing the multi-category Bayesian classifier in an nonparametric/parametric hybrid way: non-metric distance function for measuring inter-class separability; binary tree representation for nested macro-classes; Bayesian classification via class-conditional PDF estimation; multi-scale classification implemented in hierarchy.

The computations for constructing and evaluating the binary classifiers on non-leaf nodes are propagated from the root downwards through the binary tree. In the experiment on satellite image dataset, the excellent generalization capability and learnability are confirmed in terms of the testing error rate on validation dataset, which also corroborated the reliability of posterior probability estimation for multiple classes.

7 References

[1] P. Chaudhui, A. K. Ghosh, and H. Oja. “Classification based on Hybridization of Parametric and Nonparametric Classifiers,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 31, pp. 1153−1164, 2009.

[2] G. Zhai, X. Yang. “Image reconstruction from random samples with Multiscale hybrid parametric and nonparametric modeling,” IEEE Trans. Circuits and Systems for Video Technology, vol. 22, pp. 1554−1563, 2012.

[3] S. F. Masri. “A hybrid parametric/nonparametric approach for the identification of nonlinear systems,” Probabilistic Engineering Mechanics, vol. 9, pp. 47−57, 1994.

[4] J. Peres, R. Oliveira, and S. Feyo de Azevedo. "Bioprocess hybrid parametric/nonparametric modeling based on the concept of mixtures of experts," Biochemical Engineering Journal, vol. 39, pp. 190−206, 2008.
[5] L. Bruzzone, L., and R. Cossu. "A multiple-cascade-classifier system for a robust and partially unsupervised updating of land-cover maps," IEEE Trans. Geoscience and Remote Sensing, vol. 40, pp. 1984−1996, 2002.
[6] J. V. Black, and C. M. Reed. "A Hybrid Parametric, Nonparametric to Bayesian Target Tracking," in 1996 IEE Colloquium on Target Tracking and Data Fusion, pp. 178−183.
[7] B. Schölkopf, A. J. Smola. Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, MIT Press, Cambridge, MA, 2002.
[8] J. Shawe-Taylor, N. Cristianini. Kernel Methods for Pattern Analysis, Cambridge University Press, 2004.
[9] N. Cristianini, J. Shawe-Taylor. Support Vector Machines and other Kernel-based Learning Methods, Cambridge University Press, 2000.
[10] J. Weston, C. Watkins. "Support vector machines for multi-class pattern recognition," in Proc. 7th European Symposium on Artificial Neural Networks, Belgium, 1999.
[11] C. W. Hsu, C. J. Lin, "A comparison of methods for multiclass support vector machines," IEEE Transactions on Neural Networks, vol. 13, pp. 415−425, 2002.
[12] K. Crammer, Y. Singer. "On the algorithmic implementation of multiclass kernel-based vector machines," Journal of Machine Learning Research, vol. 2, pp. 265−292, 2001
[13] Y. Lee, Y. Lin, G. Wahba. "Multicategory support vector machines: Theory and applications to the classification of microarray data and satellite radiance data," Journal of the American Statistical Association, vol. 99, pp. 67−81, 2004.
[14] E. L. Allwein, R. E. Schapire, and Y. Singer. "Reducing multiclass to binary: A unifying approach for margin classifiers," Journal of Machine Learning Research, vol. 1, pp. 113−141, 2000.
[15] J. Chen, C. Wang. "Combining support vector machines with a pairwise decision tree," IEEE Geoscience and Remote Sensing Letters, vol. 5, pp. 409−413, 2008.
[16] D. Casasent, Y.C. Wang. "A hierarchical classifier using new support vector machines for automatic target recognition," Neural Networks, vol. 18, pp. 541−548, 2005.
[17] Z. Lu, L. Liang, G. Song, S. Wang. "Polychotomous kernel Fisher discriminant via top-down induction of binary tree," Computers & Mathematics with Applications, vol. 60, pp. 511−519, 2010.
[18] Z. Lu, F. Lin, H. Ying. "Design of decision tree via kernelized hierarchical clustering for multiclass support vector machines," Cybernetics and Systems, vol. 38, pp. 187−202, 2007.
[19] S. Cheong, S. H. Oh, S.-Y. Lee. "Support vector machines with binary tree architecture for multi-class classification," Neural Information Processing − Letters and Reviews, vol. 2, pp. 47−51, 2004.
[20] T. Eiter. "Distance measures for point sets and their computation," Acta Informatica, vol. 34, pp. 109–133, 1997.
[21] D. W. Jacobs, D. Weinshall, and Y. Gdalyahu. "Classification with nonmetric distances: Image retrieval and class representation," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 22, pp. 583−600, 2000.
[22] I. Niiniluoto. Truthlikeness, D. Reidel Publishing Company, 1987.
[23] J.C. Platt. "Probabilities for SV machines," In Advances in Large Margin Classifiers, A.J. Smola, P. Bartlett, B. Schölkopt, and D. Schuurmans, Ed. Cambridge, MA: MIT Press, 1999, pp. 61–73.
[24] H.T. Lin, C.J. Lin, and R.C. Weng. "A note on Platt's probabilistic outputs for support vector machines," Machine Learning, vol. 68, pp. 267−276, 2007.
[25] B. Zadrozny, C. Elkan. "Transforming classifier scores into accurate multiclass probability estimates," in Proc. 8th Int. Conf. Knowledge Discovery and Data Mining, 2002, pp. 694−699.
[26] B. Fei, J. Liu. "Binary tree of SVM: A new fast multiclass training and classification algorithm," IEEE Trans. Neural Networks, vol. 17, pp. 696−704, 2006.
[27] J. Milgram, M. Cheriet, and R. Sabourin. "Estimating accurate multi-class probabilities with support vector machines," in Proc. Int. Joint Conf. Neural Networks, 2005, pp. 1906−1911.
[28] S. Mika. "Kernel Fisher discriminant," Ph.D. dissertation, Univ. of Technology, Berlin, 2002.
[29] J. Yang, Z. Jin, J. Yang, and D. Zhang, A. F. Frangi. "Essence of kernel Fisher discriminant: KPCA and LDA," Pattern Recognition, vol. 37, pp. 2097−2100, 2004.
[30] A. Shashua. "On the relationship between the support vector machine for classification and sparsified Fisher's linear discriminant," Neural Processing Letters, vol. 9, pp. 129−139, 1999.
[31] C. M. Bishop. Neural Networks for Pattern Recognition, Oxford University Press, 1996.
[32] S. Theodoridis, K. Koutroumbas. Pattern Recognition, Academic Press, 4th Ed., 2009.
[33] S. Mika, G. Rätsch, J. Weston, B. Schölkopf, K. Müller. "Fisher discriminant analysis with kernel," in 1999 Proc. IEEE Int'l Workshop Neural Networks for Signal Processing IX, pp. 41−48.
[34] D. Zhang, J. Wang, and Y. Zhao. "Non-flat function estimation with a multi-scale support vector regression," Neurocomputing, vol. 70, pp. 420−429, 2006.
[35] P. V. Gehler. "Kernel learning approaches for image classification," Ph.D. dissertation, Saarland University, Germany, 2009.
[36] R. King, C. Feng, and A. Shutherland. "Statlog: comparison of classification algorithms on large real-world problems," Applied Artificial Intelligence, vol. 9, pp. 289–333, 1995.
[37] V. Franc, V. Hlavac. Statistical Pattern Recognition Toolbox for MATLAB. [Software]. Czech Technical University, Czech. Available: http://cmp.felk.cvut.cz/cmp/software/stprtool/