Spatially Adaptive Classification of Land Cover With Remote Sensing Data

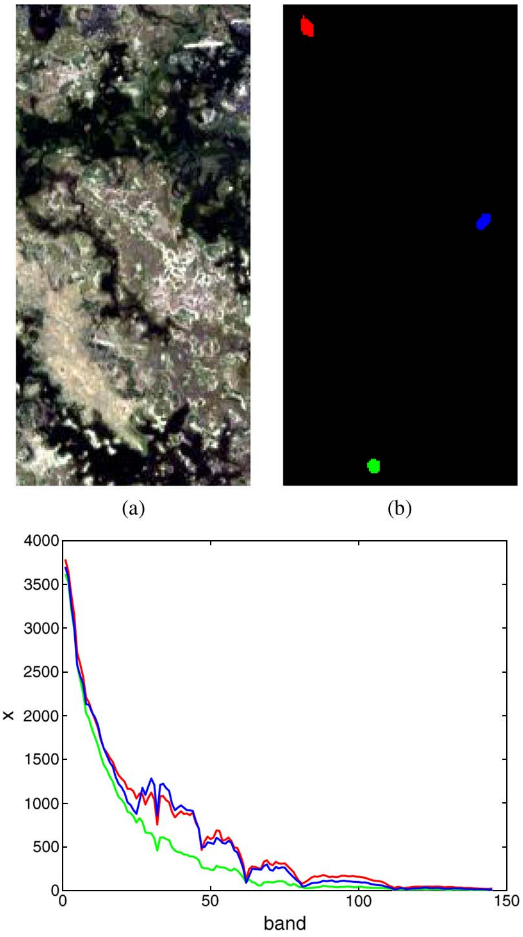
Goo Jun, *Member, IEEE*, and Joydeep Ghosh, *Fellow, IEEE*

***Abstract*—This paper proposes a novel framework called Gaussian process maximum likelihood for spatially adaptive classification of hyperspectral data. In hyperspectral images, spectral responses of land covers vary over space, and conventional classification algorithms that result in spatially invariant solutions are fundamentally limited. In the proposed framework, each band of a given class is modeled by a Gaussian random process indexed by spatial coordinates. These models are then used to characterize each land cover class at a given location by a multivariate Gaussian distribution with parameters adapted for that location. Experimental results show that the proposed method effectively captures the spatial variations of hyperspectral data, significantly outperforming a variety of other classification algorithms on three different hyperspectral data sets.**

***Index Terms*—Classification, Gaussian processes, hyperspectral imaging (HSI), kriging, spatial statistics.**

I. Introduction

REMOTE sensing data provide synoptic and timely information for identifying and monitoring large geographical areas that are less accessible by other means. In particular, hyperspectral imaging provides rich spectral information about remotely sensed objects and is one of the most useful and popular techniques for land use and land cover (LULC) classification [1]. Each pixel in a hyperspectral image consists of hundreds of spectral bands ranging from infrared to visible spectrum. Different land-cover classes show different spectral responses. For example, spectral responses of forest are quite different from spectral responses of corn fields, but there are only subtle discrepancies between spectral responses of various types of corn fields. On the other hand, there also exist within-class variations in the spectral responses of the same land-cover class. Identification of a land-cover class from other classes with similar spectral responses becomes a challenging task when the within-class variation is comparable to the between-class differences.



(a) (b) (c)

Fig. 1. Averaged spectral responses of water class at different locations. (a) Botswana image. (b) Locations of water. (c) Spectral signatures.

Variations of spectral features can be contributed to many factors such as soil composition, weather, terrain, hydrologic conditions, and/or measurement noise. Many of these are spatially varying factors. Geographically closer regions usually have similar geological and environmental conditions; hence, smaller variations of spectral responses are expected in smaller spatial footprints. Fig. 1 shows how the spectral signature of a single land-cover class changes over space. Fig. 1(a) shows the red–green–blue version of a hyperspectral image acquired by Hyperion over the Okavango Delta, Botswana. This 30-m resolution data cover a spatial extent of approximately 44 km by 7.5 km and is used in the experiments described later. Fig. 1(b) shows three different locations of water samples in this image, and Fig. 1(c) shows the average spectral responses at these locations in the corresponding color. As can be seen from the figure, there are nonnegligible changes of spectral responses within the same land-cover class.

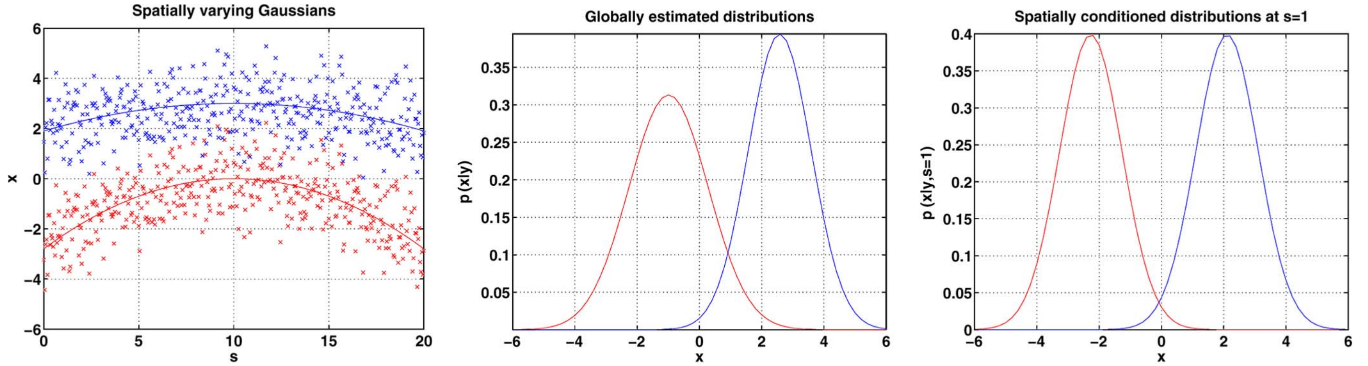
Manuscript received June 4, 2010; revised October 21, 2010; accepted November 28, 2010. Date of publication March 17, 2011; date of current version June 24, 2011. This work was supported by the National Science Foundation under Grant IIS-0705815.

G. Jun is with the Department of Biostatics, University of Michigan at Ann Arbor, MI 48109 USA (email: gjun@umich.edu).

J. Ghosh is with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712 USA (email: ghosh@ece.utexas.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TGRS.2011.2105490



(a) (b) (c)

Fig. 2. Toy example showing how spatial variations affect a global model. (a) Spatial variation. (b) $p(x|y)$ from global estimates. (c) Local model $p(x|y, s = 1)$.

Conventional classification algorithms assume a global (single) model that applies to all pixels, i.e., across the entire image. Although this assumption may hold in small spatial footprints, spectral signatures do not remain constant across the image as shown in the example. In the presence of spatial variations, a classifier with a global model trained on a small region may not generalize well to other areas. On the other hand, training such a classifier using the samples taken across the entire image may lead to large within-class variations in the training data, and the resulting classifier would have difficulties in distinguishing similar land-cover classes from one another. Fig. 2 is a toy example that illustrates the problems of using a global model under spatially varying distributions. Red and blue points in Fig. 2(a) are randomly generated from two different Gaussian distributions with spatially varying means. The x-axis represents the 1-D spatial coordinate s, and the y-axis represents a 1-D feature x. At a given location, data points of each class are randomly generated from a unimodal Gaussian distribution. Means of class-conditional distributions are generated by smooth quadratic functions of s, and the same (constant) standard deviations (SD) are used for both classes. The red and blue curves indicate the true means used to generate random samples. Fig. 2(b) shows the class-conditional Gaussian distributions $p(x|y)$ modeled by maximum likelihood (ML) estimators without utilizing any spatial information. Fig. 2(c) shows true class-conditional distributions of both classes at a specific location $p(x|y, s = 1)$. As can be seen in the example, it is possible to have better separation of classes by proper modeling of spatially varying parameters. *The main aim of this paper is to statistically model such variations using Gaussian processes (GPs), in order to spatially detrend spectral features. Classification methods acting on these detrended features result in substantially improved labeling of land covers.*

Statistical modeling of spatially varying data has long been studied as an important field of statistics, called spatial statistics or geostatistics [2]. Geostatistical techniques such as kriging have been used to model spatial dependencies in data and used for a variety of environmental problem applications [3], [4]. In kriging, each instance is modeled as an outcome of a random process, and the prediction for a new location is made by a linear combination of values at previously known locations, weighted differently according to the pairwise distances. Nearby instances usually get higher weights than distant instances, and the underlying assumption for such a weighting scheme is embodied in the first law of geography by Waldo Tobler: "Everything is related to everything else, but near things are more related than distant things [5]." This underscores the importance of neighborhood information as well as global (non-myopic [6]) relationships between spatially remote instances. The kriging approach has recently been adopted by the machine learning community, where it is referred to as a GP model [7]. In the GP model, instances in the feature space are modeled as realizations of Gaussian random processes.

We now propose a novel framework for the spatially adaptive classification of hyperspectral data and name it the Gaussian process maximum likelihood (GP-ML) model [8]. In GP-ML, spectral features of a given class are decomposed as a sum of a constant (global) component and a spatially varying component, which is modeled by Gaussian process regressions (GPRs). Once the spatially varying component is identified, it is subtracted from the original features for spatial detrending. The residual information is assumed to be spatially invariant and is modeled as conventional multivariate Gaussians to facilitate ML estimation.

II. Related Work and Background

A. Land-Cover Classification With Hyperspectral Data

In recent years, the LULC classification by hyperspectral image analysis has become an important part of remote sensing research [1], [9]–[11]. Compared to multispectral images where each pixel usually contains a few bands, pixels in hyperspectral image consist of more than a hundred spectral bands, providing fine-resolution spectral information. Classification techniques used for this application should be able to handle high-dimensional high-resolution data and a fairly high number of classes.

There have been a number of studies that utilize spatial information for hyperspectral data analyses or attempt to use features that are relatively invariant across space [12], [13]. A geostatistical analysis of hyperspectral data has been studied by Griffith [14], but no classification method was provided. One way to incorporate spatial information into a classifier is stacking feature vectors from neighboring pixels [15]. A vector stacking approach for the classification of hyperspectral data, i.e., max-cut support vector machine (MC-SVM), has been proposed by Chen *et al.* [16], where features from the homogeneous neighborhood are stacked using a MC algorithm. We compare the proposed framework to the MC-SVM algorithm in the experiment section. Another way to incorporate spatial information is via image segmentation algorithms [17], [18]. The results from these approaches largely depend on the initial segmentation results. Some algorithms exploit spatial distributions of land-cover classes directly. The simplest direct method is majority filtering [19], where the classified map is smoothed by 2-D low-pass filters. Another popular method that incorporates spatial dependencies into the probabilistic model is the Markov random field model [20]–[23]. Bazi and Melgani [24] used a GP classifier for classification of hyperspectral images but not for spatial adaptation. GPs have been also used recently for detecting change [25] and for estimating biophysical parameters [26]. Han and Goodenough [27] proposed using surrogate data for analyzing the nonlinearities in hyperspectral data. The closest approach to this paper is by Goovaerts [28], where the existence of each land-cover class is modeled by indicator kriging to be combined with the spectral classification results. However, the spatial information was not used to model the variations of spectral features. In fact, none of the aforementioned algorithms measure and model spatial variations of spectral features directly.

Generative models of hyperspectral data often assume a multivariate Gaussian distribution for each class, and both the ML classification and the expectation-maximization algorithm have been widely used in hyperspectral data analyses [29]. When applied to large spatially extended regions, a classifier is often trained at one location and applied to other locations, and there have been several studies to adapt for dynamically changing properties of hyperspectral data in such settings. Chen *et al.* applied manifold techniques to analyze the nonlinear variations of hyperspectral data [30], [31]. Kim *et al.* extended this manifold-based approach with multiresolutional analyses [32] and proposed a spatially adaptive manifold learning algorithm for hyperspectral data analysis in the absence of sufficient labeled examples [33]. Some studies have proposed classification algorithms that can transfer the knowledge learned from one region to spatially or temporally separated regions. For example, Rajan *et al.* [34] provided a framework to transfer knowledge between spatially and temporally separated hyperspectral data, but this approach does not utilize spatial relations between locations. There have also been studies on the active learning of hyperspectral data to minimize the required number of labeled instances to achieve the same or better classification accuracies [35]–[37], but these active learning algorithms do not utilize any spatial information either.

In our earlier works [8], [38], a spatially adaptive classification algorithm was proposed, where spatial variations of spectral features are characterized by GPs. In this paper, the previously proposed framework is enhanced by decomposing the spectral features of a given class as a sum of a constant (global) component and a spatially varying component and by processing each dimension separately before dimensionality reduction. These enhancements lead to a substantial improvement in performance, as evidenced across a more extensive set of experiments that cover a wider range of hyperspectral images.

B. Spatial Data Analysis

Many real-world data sets have spatial components. Spatial information is critical in certain social science data such as census, survey, and public health data taken from different cities. Other classical examples of spatial data include geological data such as the distribution of mineral ores, soil composition, and weather data. Spatial information also plays an important role in the study of ecological data such as the distribution of endangered species, vegetation data, and agricultural data. Statistical analysis of spatial data is referred to as spatial statistics or geostatistics [39]. In spatial statistics, each data point is considered to be an outcome of a random process $x(\mathbf{s})$. $\mathbf{s} \in S$ is the spatial coordinate of the data point, where $S \subset R^2$ is the set of spatial locations in a 2-D Euclidean space. In this setup, we can observe that nonspatial statistical models can be thought as a special case of the spatial model. Spatial data can be modeled as point patterns, lattice data, or geospatial data according to the characteristics of S [2]. We will focus on the geospatial data model, where S is a fixed set of points in R^2. The most common task of geospatial data analysis is predicting the value of $x(\mathbf{s}_*)$ for a new location $\mathbf{s}_*$ from the fixed set of existing data $x(\mathbf{s})$, $\mathbf{s} \in S$. The process of finding the optimal linear predictor is called kriging, named after a South African mining engineer, D. G. Krige [40], [41]. When the underlying stochastic process is a Gaussian random process, the linear predictor obtained by kriging is optimal in the least-square sense. In kriging, $x(\mathbf{s})$ is modeled as a sum of a trend (mean) function $\mu(\mathbf{s})$ and an additive noise term $\epsilon(\mathbf{s})$

$$x(\mathbf{s}) = \mu(\mathbf{s}) + \epsilon(\mathbf{s}).$$

There are several different types of kriging. The simplest one is called simple kriging, where it is assumed that the process has a zero mean $\mu(\mathbf{s}) = 0$, and the covariance function is known *a priori*. There are more general techniques such as ordinary kriging and universal kriging, where $\mu(\mathbf{s})$ is assumed to be some unknown constant and unknown function, respectively [2], [39]. The GPR model in machine learning is a simple kriging model, since a zero-mean prior is typically assumed [7]. Kriging has been widely used to model various kinds of spatially varying quantities but has rarely been combined with classification algorithms to develop spatially adaptive classification schemes.

Recently, a technique called geographically weighted regression (GWR) [42] has been studied for regression problems where relationships between independent and dependent variables vary over space. GWR is different from kriging in a sense that its objective is to find spatially varying regression coefficients, while, in kriging, the objective is to find the spatial variation of variables. GWR and kriging both can be used for similar tasks, and a recent comparative study has shown that kriging is more suitable for prediction of spatially varying quantities, but a hybrid approach may be beneficial for description of complex spatially varying relationships [43].

C. ML Classification

The ML classifier is a popular technique for classification of hyperspectral data. Let $y \in \{y_1, \ldots, y_c\}$ be the class label associated with the spectral feature vector $\mathbf{x} \in R^d$. The posterior class probabilities are given by the Bayes' rule

$$p(y = y_i|\mathbf{x}, \Theta) = \frac{p(y = y_i|\Theta)p(\mathbf{x}|y = y_i, \Theta)}{\sum_{i=1}^{c} p(y = y_i|\Theta)p(\mathbf{x}|y = y_i, \Theta)} \quad (1)$$

where Θ is the set of model parameters. Let the class-conditional distributions be modeled by multivariate Gaussian distributions

$$p(\mathbf{x}|y = i, \Theta) \sim \mathcal{N}(\boldsymbol{\mu_i}, \Sigma_i) = \frac{1}{(2\pi)^{n/2}|\Sigma_i|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu_i})^T \Sigma_i^{-1}(\mathbf{x}-\boldsymbol{\mu_i})}. \quad (2)$$

$\Theta = \{(\boldsymbol{\mu_i}, \Sigma_i)|i = 1, \ldots, c\}$, where $\boldsymbol{\mu_i}$ and Σ_i are the mean vector and the covariance matrix of the ith class. The ML classifier makes an ML estimation of these parameters using the training data with known class labels. It then picks the class label of a test instance as the one that has the maximum posterior probability according to (1) and (2), i.e., it applies the Bayes' decision rule [44].

D. GPR

Over the last decade, the GP model for machine learning [7] has gained popularity. It has been applied to many domains including regression and classification problems. GP models are generally well suited for regression problems since they eliminate the model selection problem [45].

A random vector $\mathbf{x}$ is jointly Gaussian, denoted as $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, if and only if its joint probability density function has the form of (2). One useful property of a Gaussian random vector is that conditional and marginal distributions of Gaussian random vectors are also Gaussian. A GP is a random process such that all finite dimensional distributions of the process are jointly Gaussian random vectors [7]. Let x be a random process indexed by $\mathbf{s}$, then $x(\mathbf{s})$ is a GP, if and only if $\mathbf{x} = [x(\mathbf{s}_1), x(\mathbf{s}_2), \ldots, x(\mathbf{s}_n)]^T$ is a jointly Gaussian random vector for any finite set of $S = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n\}$. As a Gaussian distribution is defined by its mean and covariance, a GP is fully defined by a mean function $\mu(\mathbf{s})$ and a covariance function $k(\mathbf{s}_1, \mathbf{s}_2)$ and denoted as $x(\mathbf{s}) \sim \mathcal{GP}(\mu(\mathbf{s}), k(\mathbf{s}_1, \mathbf{s}_2))$. In GPR, the target variable is modeled by a Gaussian random process. Let us assume that the values of x are observed for some $S = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n\}$, and $x(\mathbf{s})$ is modeled as $x(\mathbf{s}) = f(\mathbf{s}) + \epsilon$, where ϵ is an additive white Gaussian noise term $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)$. We assume a (zero mean) GP prior for $f(\mathbf{s})$

$$f(\mathbf{s}) \sim \mathcal{GP}\left(\mu(\mathbf{s}) = 0,\ k(\mathbf{s}_1, \mathbf{s}_2)\right).$$

Then, given $f(\mathbf{s})$, the distribution of $x(\mathbf{s})$ is also Gaussian

$$p\left(x(\mathbf{s})|f(\mathbf{s})\right) = \mathcal{N}\left(f(\mathbf{s}), \sigma_\epsilon^2\right).$$

In regression problems, we are interested in making predictions based on the training data $\mathbf{x} = [x(\mathbf{s}_1), \ldots, x(\mathbf{s}_n)]^T$. The predictive distribution of an out-of-sample instance $x(\mathbf{s}_*)$ can be easily derived from the conditional distribution of jointly Gaussian random vectors as

$$p\left(x(\mathbf{s}_*)|x(\mathbf{s}_1), \ldots, x(\mathbf{s}_n)\right) = \mathcal{N}\Big(\mathbf{k}(\mathbf{s}_*, S)\left[K_{SS} + \sigma_\epsilon^2 I\right]^{-1}\mathbf{x}, \\ k(\mathbf{s}_*, \mathbf{s}_*) + \sigma_\epsilon^2 - \mathbf{k}(\mathbf{s}_*, S)\left[K_{SS} + \sigma_\epsilon^2 I\right]^{-1}\mathbf{k}(S, \mathbf{s}_*)\Big) \quad (3)$$

where $\mathbf{k}(\mathbf{s}_*, S) = [k(\mathbf{s}_*, \mathbf{s}_1), k(\mathbf{s}_*, \mathbf{s}_2), \ldots, k(\mathbf{s}_*, \mathbf{s}_n)]$, $\mathbf{k}(S, \mathbf{s}_*) = \mathbf{k}(\mathbf{s}_*, S)^T$, and K_{SS} is a matrix such that its (i, j)th element $K_{ij} = k(\mathbf{s}_i, \mathbf{s}_j)$. Given a set of known instances $\mathbf{x}$, the predictive distribution is Gaussian with parameters shown in (3). The predictive mean $\mathbf{k}(\mathbf{s}_*, S)[K_{SS} + \sigma_\epsilon^2 I]^{-1}\mathbf{x}$ is a linear combination of known instances $\mathbf{x}$, weighted according to the spatial correlation between $\mathbf{s}_*$ and $(\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n)$, which is represented by $\mathbf{k}(\mathbf{s}_*, S)$. In other words, we have a linear estimator that uses the kernel smoothing values $\mathbf{k}(\mathbf{s}_*, S)$ as the weights. The predictive variance of $x(\mathbf{s}_*)$ is the difference between the noninformative variance defined by the prior distribution and the information provided by spatial correlation. As can be seen in the formula, the covariance function $k(\mathbf{s}_1, \mathbf{s}_2)$ fully determines the characteristics of a GP. In many applications, the value of $k(\mathbf{s}_1, \mathbf{s}_2)$ decreases as $|\mathbf{s}_1 - \mathbf{s}_2|$ increases, which means that nearby known values have more influence in determining the unknown value $x(\mathbf{s}_*)$ as compared to known values further away.

III. Methods

A. GP-ML Framework

We propose a novel framework for the classification of hyperspectral data, namely, the GP-ML model, which characterizes the mean of each spectral band as a random process over space using the GP model. This framework provides a practical and effective way to model spatial variations in hyperspectral images. As discussed in earlier sections, the GP model has been long known in spatial statistics as *kriging* [2]. Traditionally, kriging has been considered to be only suitable for modeling of a single or small number of target variables. In this paper, we directly model spatially adaptive class-conditional distributions of high-dimensional data. For a given class, it is assumed that each band is spatially independent of other bands and well described by a single GP. Although values of different spectral bands in hyperspectral data are correlated, we simplify the problem by employing the naïve Bayes' assumption. Naïve Bayes' classifiers assume that features are independent given a class label, and studies have shown that the algorithm works well for many high-dimensional classification problems [46]. Modeling multiple correlated target variables has also been studied in spatial statistics, and it is called *cokriging* [2]. It is impractical and too demanding, however, to model hyperspectral data directly by cokriging [28], since cokriging requires solving $(n + 1) \cdot d$ linear equations for n data points with d dimensions, and the system becomes sensitive to noise due to the greatly increased number of parameters. There is also a broad literature on estimating the covariance matrix when faced with inadequate amounts of training data [47], [48] that can be applied if one desires to learn full covariance matrices. Estimating nonstationary covariance functions using local estimation methods as in [27] could be also considered, but it

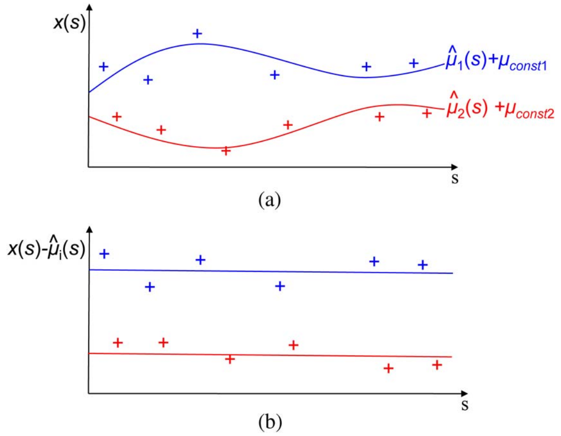


Fig. 3. Simplified illustration of GP-ML idea. The spatially varying mean of each component of the ith class $\hat{\boldsymbol{\mu}}_i(\mathbf{s})$ is modeled using GPR, and then, the variation is removed from each data point before fitting a stationary Gaussian distribution. (a) Data with spatial variation. (b) Spatial variation removed.

would significantly increase the complexity of the proposed framework.

Let $\mathbf{x}$ be a d-dimensional vector representing spectral bands of a pixel in a hyperspectral image, and $y \in \{y_1, y_2, \ldots, y_c\}$ be the class label that represents land-cover types, where c is the number of classes. The class-conditional probability distribution $p(\mathbf{x}|y_i)$ is usually assumed to be a multivariate Gaussian. For simple notation, let us focus on a single class and omit i where possible. Typically, both the mean $\boldsymbol{\mu}$ and the spectral covariance Σ are considered to be constant over the entire image. Instead, GP-ML models $\mathbf{x}(\mathbf{s})$ as a Gaussian random process are indexed by a spatial coordinate $\mathbf{s} \in R^2$. It is assumed that the spectral covariance matrix Σ is constant without spatial variation. The mean function of $\boldsymbol{\mu}(\mathbf{s})$ is modeled as a sum of a constant (global) mean $\boldsymbol{\mu}_{\text{const}}$ and a spatially varying zero-mean function $\hat{\boldsymbol{\mu}}(\mathbf{s})$, i.e., $\mathbf{x}(\mathbf{s}) \sim \mathcal{GP}(\hat{\boldsymbol{\mu}}(\mathbf{s}) + \boldsymbol{\mu}_{\text{const}}, k(\mathbf{s}_1, \mathbf{s}_2))$. Fig. 3 illustrates the concept using a simplified 1-D two-class example. Fig. 3(a) shows the original data with spatial variation. Each class is modeled as a sum of a constant mean $\boldsymbol{\mu}_{\text{const}}$ and the spatially varying zero-mean function $\boldsymbol{\mu}(\mathbf{s})$. Once we find $\hat{\boldsymbol{\mu}}(\mathbf{s})$ by GPRs and subtract it from the data, the residual information $\mathbf{x}(\mathbf{s}) - \hat{\boldsymbol{\mu}}(\mathbf{s})$ can be modeled with standard Gaussian distributions as shown in Fig. 3(b).

First, we subtract the constant (global) mean of the ith class from each instance of that class to make the data zero mean

$$\hat{\mathbf{x}}_k = \mathbf{x}_k - \boldsymbol{\mu}_{\text{const}}, \quad 1 \le k \le n_i,$$

$$\text{where} \qquad \boldsymbol{\mu}_{\text{const}} = \frac{1}{n_i}\sum_{k=1}^{n_i} \mathbf{x}_k.$$

$\mathbf{x}_k$, $k = 1, \ldots, n_i$ are instances that belong to the ith class, where n_i is the number of instances in the ith class. Now, let $\hat{\mathbf{x}}^j$ be a vector consisting of the jth bands of $\mathbf{x}_k$'s, $\hat{\mathbf{x}}^j = [\hat{x}_1^j, \hat{x}_2^j, \ldots \hat{x}_{n_i}^j]^T$. Then, we find the spatially varying means for jth dimension of the training data $\hat{\boldsymbol{\mu}}^j = [\hat{\mu}_1^j(\mathbf{s}_1), \ldots, \hat{\mu}_{n_i}^j(\mathbf{s}_{n_i})]^T$ and the predictive mean at the location $\mathbf{s}_*$ of the test instance $\hat{\boldsymbol{\mu}}_*(\mathbf{s}_*) = [\hat{\mu}_*^1(\mathbf{s}_*), \ldots, \hat{\mu}_*^d(\mathbf{s}_*)]^T$ according to (3)

$$\hat{\boldsymbol{\mu}}^j = \sigma_{f_j}^2 K_{SS}\left[\sigma_{f_j}^2 K_{SS} + \sigma_{\epsilon_j}^2 I\right]^{-1} \hat{\mathbf{x}}^j \tag{4}$$

$$\hat{\mu}_*^j(\mathbf{s}_*) = \sigma_{f_j}^2 \mathbf{k}(\mathbf{s}_*, S)\left[\sigma_{f_j}^2 K_{SS} + \sigma_{\epsilon_j}^2 I\right]^{-1} \hat{\mathbf{x}}^j \tag{5}$$

where $k = 1, \ldots, n_i$, $j = 1, \ldots, d$. $\sigma_{f_j}^2$ and $\sigma_{\epsilon_j}^2$ are hyperparameters for signal and noise powers of the jth band in the data. Then, we subtract $\hat{\boldsymbol{\mu}}_k = [\hat{\mu}_k^1(\mathbf{s}_k), \ldots, \hat{\mu}_k^d(\mathbf{s}_k)]^T$ from each $\mathbf{x}_k$ to remove spatially varying components from the original data

$$\mathbf{x}'_k = \mathbf{x}_k - \hat{\boldsymbol{\mu}}_k, \quad 1 \le k \le n_i.$$

$\mathbf{x}'$ can be thought as spatially detrended instances; hence, the distribution of $\mathbf{x}'$ is assumed to be multivariate Gaussian without spatial variation. Rather than estimating the parameters of high-dimensional Gaussian distributions, it is desirable to reduce the dimensionality of the data. For example, Fisher's multiclass linear discriminant analysis (LDA) finds a $R^d \rightarrow R^{(c-1)}$ dimensional projection Φ for a c-class problem [44]. The projection matrix Φ could be obtained from any linear other dimensionality reduction methods. Fisher's LDA is employed in most of our experiments since it finds the optimal linear subspace for separation of Gaussian distributed data; hence, it conforms to the GP-ML framework that assumes multivariate Gaussian distributions for spatially detrended data. There are other linear dimensionality reduction techniques developed for classification of hyperspectral data such as decision boundary feature extraction (DBFE) [49] and nonparametric weighted feature extraction (NWFE) [50]. The proposed framework is also evaluated using NWFE, which is more recent and less restrictive than DBFE, and the result will be presented in the experiment section. Let Φ be the $m \times d$ projection matrix $(m < d)$ obtained from a dimensionality reduction algorithm. The parameters for multivariate Gaussian distribution at s_* in the m dimensional linear subspace are

$$\boldsymbol{\mu}_{*\phi}(\mathbf{s}_*) = \Phi\left(\hat{\boldsymbol{\mu}}_*(\mathbf{s}_*) + \boldsymbol{\mu}_{\text{const}}\right), \tag{6}$$

$$\Sigma_\phi = \frac{1}{n_i - 1}\sum_{k=1}^{n_i}\left(\Phi\mathbf{x}'_k - \frac{1}{n_i}\sum_{l=1}^{n_i}\Phi\mathbf{x}'_l\right) \times \left(\Phi\mathbf{x}'_k - \frac{1}{n_i}\sum_{l=1}^{n_i}\Phi\mathbf{x}'_l\right)^T. \tag{7}$$

The subscript ϕ is used to denote the parameters and instances in the projected space. The class-conditional distribution of $\mathbf{x}_{*\phi}$ at location $\mathbf{s}_*$ in the projected space is assumed to be Gaussian

$$p(\mathbf{x}_{*\phi}|\mathbf{s}_*, y_i) \sim \mathcal{N}\left(\boldsymbol{\mu}_{*\phi}(\mathbf{s}_*), \Sigma_\phi\right).$$

By substituting (6) and (7) into (2) together with the value $\mathbf{x}_{*\phi} = \Phi\mathbf{x}_*$, we can calculate the probability that the test instance $\mathbf{x}_*$ at location $\mathbf{s}_*$ belongs to the ith class. By repeating the whole process for all the classes, one can predict the class label y_* that has the highest class-conditional probability.

Fig. 4 shows the results of applying the GP-ML model to the Indian Pine data. The x-axis represents the spatial coordinate

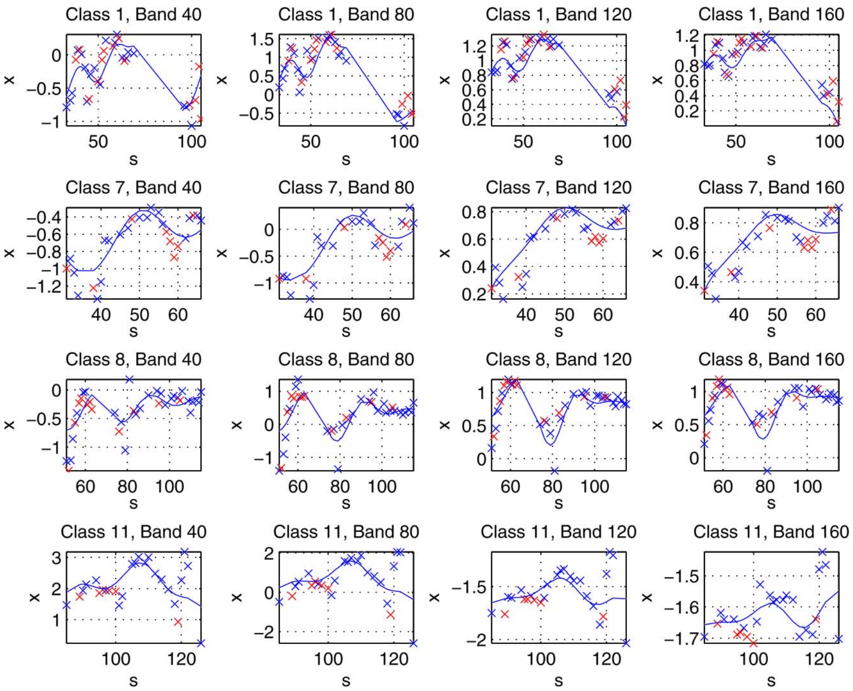


Fig. 4. Examples of spatial variations from Indian Pine data. Blue crosses are the training data (50%), red crosses are the test data at different locations, and blue curves are the predicted means obtained by GPRs. Points are selected by vertical and horizontal slices of the image containing the most samples of a given class.

s along horizontal or vertical slices, and the y-axis represents the spectral values of a selected band. Points in blue are the training instances, and points in red are the test instances. The blue curves are the mean functions of the jth spectral band $\hat{\mu}^j(\mathbf{s}) + \mu^j_{\text{const}}$, predicted by the GP-ML model. As can be seen from the figure, there are significant amounts of spatial variation in the spectral responses of a given class, and they are effectively captured by the GP-ML algorithm.

B. *Efficient Computation of GPs*

In GP-ML, we need d GPs per class to model d-dimensional data. In hyperspectral data analysis, this implies that we need to compute hundreds of GPRs, which is computationally very challenging. The most time-consuming part is the inversion of the covariance matrix as in (4) and (5). When we have n instances in a class, $(\sigma_f^2 K_{SS} + \sigma_\epsilon^2 I)$ is an $n \times n$ matrix; hence, inverting the matrix requires $O(n^3)$ computations. Using Cholesky decomposition as in [7] helps when we need rank-1 updates, but in our case, K_{SS} is fixed for a given class, and $(\sigma_f^2, \sigma_\epsilon^2)$ varies for each dimension. In this case, Cholesky decompositions cannot be updated efficiently, since $\sigma_\epsilon^2 I$ is a full-rank matrix. Instead, we exploit the eigen decomposition of the covariance matrix. K_{SS} is a positive semidefinite matrix; hence, we can diagonalize the matrix

$$\begin{aligned} K_{SS} &= V\Lambda V^T = V\,\mathrm{diag}(\lambda_k)V^T, \quad k = 1,\ldots,n \\ K_{SS}^{-1} &= V\Lambda^{-1}V^T = V\,\mathrm{diag}\left(\lambda_k^{-1}\right)V^T. \end{aligned}$$

Columns of V are eigenvectors of K_{SS}, and Λ is a diagonal matrix such that the kth diagonal element λ_k is the corresponding eigenvalue of the kth column of V. Since V is an orthonormal matrix, $VV^T = I$ and $VIV^T = I$; hence, we can derive simple analytical solutions for the inverses in (4) and (5)

$$\begin{aligned} \left(\sigma_f^2 K_{SS} + \sigma_\epsilon^2 I\right)^{-1} &= \left(\sigma_f^2 V\Lambda V^T + \sigma_\epsilon^2 VIV^T\right)^{-1} \\ &= V\left(\sigma_f^2\Lambda + \sigma_\epsilon^2 I\right)^{-1}V^T \\ &= V\,\mathrm{diag}\left(\frac{1}{\sigma_f^2\lambda_k + \sigma_\epsilon^2}\right)V^T \end{aligned}$$

where $k = 1,\ldots,n$. In the same manner, (4) can be further simplified as

$$\begin{aligned} &\sigma_f^2 K_{SS}\left(\sigma_f^2 K_{SS} + \sigma_\epsilon^2 I\right)^{-1} \\ &\quad = V\left(\sigma_f^2\Lambda\right)\ \cancel{V^T V}\left(\sigma_f^2\Lambda + \sigma_\epsilon^2 I\right)^{-1}V^T \\ &\quad = V\,\mathrm{diag}\left(\frac{\sigma_f^2\lambda_k}{\sigma_f^2\lambda_k + \sigma_\epsilon^2}\right)V^T. \end{aligned}$$

It is important to note that the matrix multiplications in (4) and (5) should be calculated from right to left because it will always leave a column vector in the right end of the equation, and we do not need to multiply two $n \times n$ matrices. This method has the time complexity of $O(n^2)$ instead of $O(n^3)$ for the entire calculation once we have the eigen decomposition beforehand. Because K_{SS} is common across all dimensions for a given class, we need only one eigen decomposition per class.

C. *Covariance Function*

In the GP model, a covariance function determines the nature of the process, and the covariance function is characterized by hyperparameters. Most covariance functions have a hyperparameter called the length parameter, which determines how fast the correlation between two points changes as the distance between the points increases. We employed the popular squared exponential covariance function [7]

$$k(\mathbf{s}_1, \mathbf{s}_2) = \exp\left(-\frac{\|\mathbf{s}_1 - \mathbf{s}_2\|^2}{2L^2}\right) \tag{8}$$

where L is the length parameter. In GP-ML, the length parameter L is assumed to be identical over all classes and over all dimensions. The signal power σ_f^2 and the noise power σ_ϵ^2 are also hyperparameters.

There are two different approaches for hyperparameter estimation [7]: greedy search and cross validation. The greedy search method uses the partial derivatives of the likelihood of the GP and finds a locally optimal set of hyperparameters. In this paper, we used cross validation to find the length parameter L that maximizes the overall classification accuracies. We did not use the likelihood-based method because of two reasons. First, we have too many GPs that are not independent of one another; hence, it is difficult to model the overall likelihood. Second, finding hyperparameters based on individual likelihood is not appropriate, since our objective function is the overall classification accuracies rather than the fitness of individual processes; hence, it is better to select parameters based on the classification results. A predefined set of L values were tried using four-fold cross validation on the training data to find the one that yields the highest classification accuracy. The range of L was determined according to the spatial resolution of the hyperspectral image.

Hyperparameters for the signal power and the noise power are explicitly measured from the training data. The variance of the jth band σ_j^2 is measured and assigned to the signal power $\sigma_{f_j}^2$ and the noise power $\sigma_{\epsilon_j}^2$ by assuming a fixed signal-to-noise ratio, $R = \sigma_f^2/\sigma_\epsilon^2$

$$\begin{aligned}\sigma_j^2 &= \sigma_f^2 + \sigma_\epsilon^2 = (R+1)\sigma_\epsilon^2\\ \sigma_\epsilon^2 &= \frac{1}{R+1}\sigma_j^2 \quad \sigma_f^2 = \frac{R}{R+1}\sigma_j^2.\end{aligned}$$

Initially, we tried selection of the best value for R also by cross validation, but values ranging from 5 to 100 did not make noticeable changes in overall accuracies; hence, a fixed value of ten is used in the following experiments.

IV. Experiments

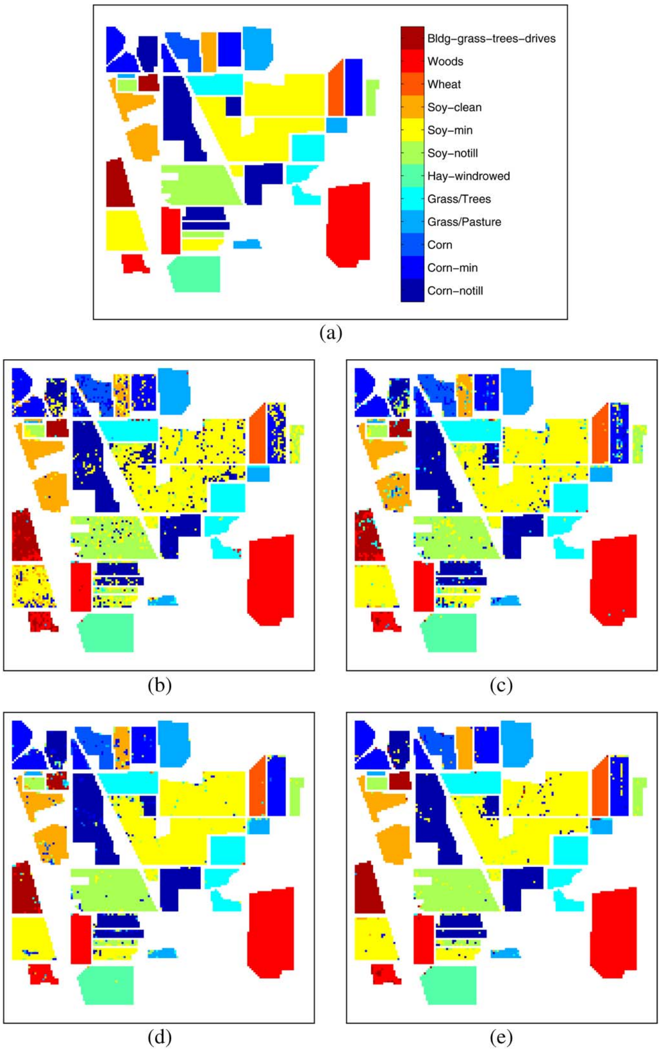
Three different hyperspectral data sets are used for empirical evaluations of the proposed framework: Purdue University's Indian Pine data from Indiana [16], [51], National Aeronautics and Space Administration's (NASA) Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) data from the John F. Kennedy Space Center (KSC) area in Florida [52], and the Earth Observing-1 (EO-1) hyperspectral data from the Okavango Delta area in Botswana, Africa [53]. Detailed description of each data set will be given in the following sections.

We compared the performance of the GP-ML algorithm to three other classification methods: the Gaussian ML classifier, the SVM classifier, and the MC-SVM [16]. The ML algorithm is selected as a baseline method, since GP-ML is a spatially adaptive alteration of ML. GP-ML is identical to the ML classifier if there is no spatial variation in the expected signature of a class. SVM is a popular classification algorithm particularly for high-dimensional data and has been widely employed for the classification of hyperspectral data [54]. MC-SVM is a spatially adaptive SVM technique that incorporates the standard SVM with the MC algorithm, which is a preprocessing method for hyperspectral data analysis to incorporate spatial information as augmented features. The MC algorithm processes each instance in the training data by selecting its 3 × 3 neighborhood and then dividing the pixels in this neighborhood into two groups using the MC of the fully connected neighborhood graph. Pixels in the same partition as the training instance are assumed to belong to the same land-cover class, and averaged feature values from those pixels are concatenated to the feature of the given instance. As a result, the dimensionality of the data is doubled. It has been shown that the MC-SVM algorithm performs better than many other previously proposed methods including Markov random field and majority filtering algorithms that also exploit spatial information [16]. MC-SVM results are included in our experiments since [16] compares the results to other spatially adaptive classification techniques using the Botswana data, which is also used in our experiments. For fairness, it should be noted that the MC-SVM requires more information than other algorithms, since it utilizes information from unlabeled samples in the 3 × 3 neighborhoods of training instances.

Four-fold cross validations with subsampling are used for evaluation. The entire data set is divided into four equally sized subsets, and for each experiment, one of those subsets is used as the test data, and the other three subsets are used as the training data. Cross-validation sets are kept the same for different classification algorithms for better comparison. This setup is equivalent to having four different hold-out sets, where all algorithms are tested with the same four hold-out test sets. Each set of the training data is then further subsampled at 20%, 50%, 75%, and 100% sampling rates to observe how each classification algorithm performs with different amounts of training data. Thus, a 100% sampling rate means that we used 75% of the entire data for training for each of the four cross-validation runs, a 50% sampling rate means that we used 37.5% of the entire data for training, etc. The ML classifier also uses Fisher's multidimensional LDA for dimensionality reduction. Radial basis function (RBF) kernels are used for SVM and MC-SVM. For multiclass SVM classification, the one-versus-all with continuous output scheme is used [55]. Parameters for RBF kernels are also searched by cross validation, i.e., in each experiment, the given training set is further divided into training–training and training–validation sets, using four-fold cross validation once again. Thus, now, with 20% sampling rate, we use $20\% \times 75\% = 15\%$ of the training data (which is $15\% \times 75\%$ of the entire data) as training–training and the remaining 5% of the training data as training–validation for parameter search. Hyperparameter search for the proposed GP-ML algorithm is also performed in the same manner. The MC-SVM setup is identical to the SVM setup except that it uses the stacked vectors as input features. All results reported are averaged over the hold-out sets and are, thus, indicative of the true generalization error.

TABLE I
AVERAGE CLASSIFICATION ACCURACY [WITH STANDARD DEVIATIONS (SD)] ON HOLD-OUT SETS FOR INDIAN PINE DATA

	Percentage of Available Training Data Used			
Classifier	20%	50%	75%	100%
ML	77.86 (±1.02) %	83.94 (±1.17) %	84.61 (±1.34) %	85.52 (±1.15) %
SVM	79.83 (±0.19) %	86.31 (±0.43) %	88.27 (±1.02) %	89.74 (±0.63) %
MC-SVM	86.38 (±1.02) %	92.76 (±0.81) %	95.16 (±0.48) %	96.47 (±0.47) %
GP-ML	**92.87** (±0.53) %	**95.97** (±0.39) %	**97.84** (±0.61) %	**98.26** (±0.31) %



(a)

(b)

(c)

(d)

(e)

Fig. 5. Classified maps for the Indian Pine data, comparing groundtruth with the results of four different methods. 50% of the available training data is used. (a) Groundtruth. (b) ML. (c) SVM. (d) MC-SVM. (e) GP-ML.

A. Indian Pine Data

The Indian Pine data and the ground references were provided by the Laboratory for Applications of Remote Sensing at Purdue University [51], [56]. This data set is one of the most well-known hyperspectral data and has been used in multiple studies [54]. The size of the image is 145 × 145 pixels, and each pixel consists of 220 bands with 17-m spatial resolution [57]. The data were obtained by NASA/Ames using the AVIRIS on June 12, 1992. The ground reference image originally contained 16 different land-cover classes, and we discarded four classes that have less than 100 samples to avoid the small sample-size problem. Twelve classes are used in the experiments as shown in Table II. Water absorption bands and noisy bands (band 1, 104–109, 150–163, and 219–220) are removed from the data before experiments.

Table I shows the overall classification accuracies for the Indian Pine data. The proposed GP-ML classifier consistently shows significantly better results than ML, SVM, and MC-SVM results. In particular, with the minimum amount of training data (20%), GP-ML shows an average overall accuracy of 92.87%, while the second best result from MC-SVM is only 86.38%. Fig. 5 shows the example classification results using 50% of the training data together with the ground reference map. Twelve land-cover classes are shown in different colors. Table II shows confusion matrices from all four algorithms. Each row indicates the number of instances classified as the corresponding class, and each column indicates the number of instances originally labeled as the corresponding class. It is noticeable that although SVM generally shows better overall accuracies than the ML classifier, for certain classes, the SVM result is much worse than that of the ML classifier. Techniques with spatial information generally show much lower error rates for all classes, and GP-ML dominates other classifiers in most cases. The worst case error of GP-ML is around 12% for the *Corn* class, which is also significantly better than the worst-case error of MC-SVM, which is about 18% for the *Soy-clean* and the *Building–grass–trees–drives* classes. Detailed explanation about these classes can be found in [56]. From Table II(a) and (b), nonspatial classifiers make many errors distinguishing different types of tillage. For example, both ML and SVM methods show many errors between *Son-notill* and *Soy-min*. It is clear that spatial information helps a lot in these classes, as Table II(c) and (d) shows significantly reduced number of errors in the same categories. The proposed GP-ML shows better performances not only for the *Soy* classes (classes 7 to 9) than other classifiers but also for the different types of tillages in the *Corn* classes (classes 1 to 3).

B. KSC

The KSC data set was acquired by the NASA AVIRIS sensor over the KSC area on March 23, 1996 [52]. The data originally consist of 242 bands, and the bands that were noisy or impacted by water absorption were removed, which leaves 176 bands for use. The groundtruth was developed using land-cover maps derived by the KSC staff from color infrared photography, Landsat Thematic Mapper imagery, and field surveys. Land-cover identification for this area is difficult because of the similarity of the spectral signatures between certain classes and the existence of mixed classes. There exist 13 different land-cover types including water and mixed classes. The hyperspectral image used for experiments has 512 × 614 pixels with 18-m spatial resolution.

TABLE II
CONFUSION MATRICES FOR INDIAN PINE DATA USING 50% OF AVAILABLE TRAINING DATA. (a) ML. (b) SVM. (c) MC-SVM. (d) GP-ML

(a)

No.	Class Name	1	2	3	4	5	6	7	8	9	10	11	12
1	Corn-notill	1193	62	27	0	0	0	46	173	4	0	0	0
2	Corn-min	51	598	39	0	0	0	3	97	38	0	0	0
3	Corn	7	27	154	0	0	0	0	3	0	0	0	0
4	Grass/Pasture	2	0	2	457	2	0	1	17	2	0	2	3
5	Grass/Trees	0	0	1	24	725	0	7	2	0	0	2	8
6	Hay-windrowed	0	0	0	0	0	488	0	0	1	0	0	0
7	Soy-notill	38	3	0	0	3	0	698	135	4	0	0	0
8	Soy-min	131	113	2	4	1	0	211	1968	61	1	0	0
9	Soy-clean	11	31	9	11	0	0	0	69	504	0	0	1
10	Wheat	0	0	0	0	0	0	0	0	0	210	0	0
11	Woods	0	0	0	1	7	1	0	1	0	1	1233	58
12	Bldg-grass-tree-drive	1	0	0	0	9	0	2	3	0	0	57	310

(b)

No.	Class Name	1	2	3	4	5	6	7	8	9	10	11	12
1	Corn-notill	1184	17	7	0	0	0	9	53	6	0	0	1
2	Corn-min	101	681	39	5	0	0	12	60	23	0	0	2
3	Corn	20	57	174	0	0	0	5	36	16	0	0	0
4	Grass/Pasture	8	10	2	463	1	1	3	15	5	0	3	7
5	Grass/Trees	12	4	1	8	726	0	9	17	3	0	4	29
6	Hay-windrowed	3	4	0	4	5	485	1	5	7	0	2	12
7	Soy-notill	51	14	5	7	0	0	786	166	23	1	0	5
8	Soy-min	54	41	1	4	6	1	141	2104	35	1	0	6
9	Soy-clean	1	6	4	5	0	1	2	9	496	2	0	2
10	Wheat	0	0	0	0	2	0	0	1	0	207	0	4
11	Woods	0	0	0	1	4	0	0	0	0	1	1249	88
12	Bldg-grass-tree-drive	0	0	1	0	3	1	0	2	0	0	36	224

(c)

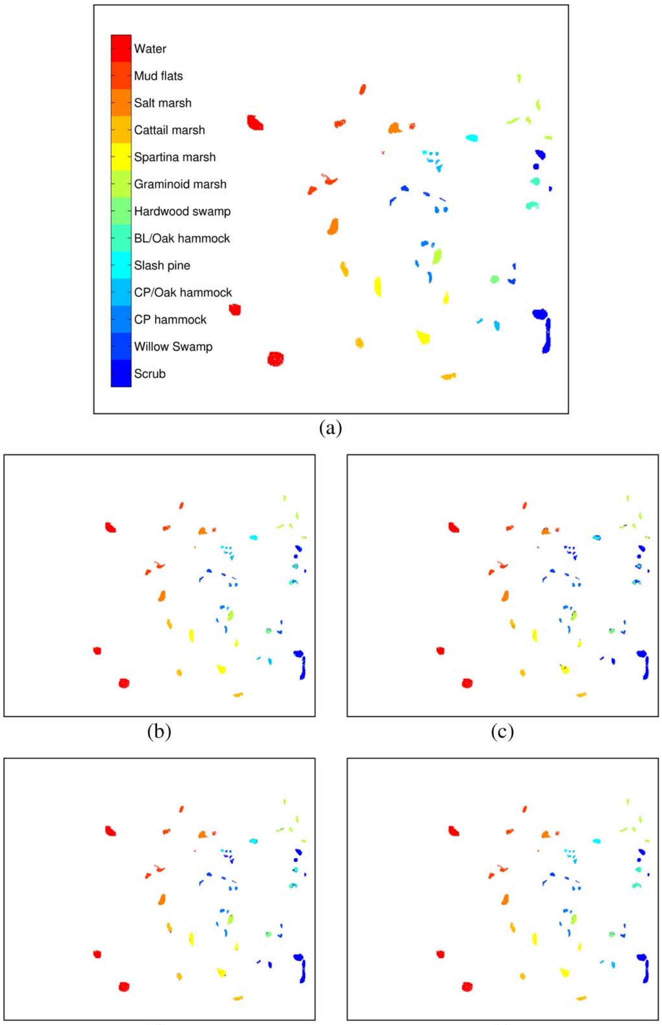
No.	Class Name	1	2	3	4	5	6	7	8	9	10	11	12
1	Corn-notill	1297	9	1	0	0	0	6	27	8	0	0	1
2	Corn-min	74	759	13	0	0	0	11	24	29	0	0	1
3	Corn	11	35	212	0	0	0	7	14	9	0	2	0
4	Grass/Pasture	4	5	7	462	0	0	3	4	8	1	2	7
5	Grass/Trees	5	4	1	18	732	0	7	11	3	1	1	17
6	Hay-windrowed	1	1	0	2	4	484	3	6	9	2	3	7
7	Soy-notill	24	5	0	6	1	4	878	70	26	0	1	4
8	Soy-min	18	12	0	6	4	1	49	2309	16	0	0	5
9	Soy-clean	0	4	0	3	2	0	4	1	506	0	0	2
10	Wheat	0	0	0	0	1	0	0	2	0	207	3	8
11	Woods	0	0	0	0	1	0	0	0	0	1	1276	15
12	Bldg-grass-tree-drive	0	0	0	0	2	0	0	0	0	0	6	313

(d)

No.	Class Name	1	2	3	4	5	6	7	8	9	10	11	12
1	Corn-notill	1370	4	3	1	0	0	29	50	1	0	0	0
2	Corn-min	11	800	17	0	0	0	1	15	8	0	0	0
3	Corn	1	4	206	1	0	0	1	2	0	0	0	0
4	Grass/Pasture	1	0	0	473	2	0	0	1	0	0	0	2
5	Grass/Trees	1	0	1	8	742	0	1	3	0	0	2	1
6	Hay-windrowed	0	0	0	0	0	488	0	0	1	0	0	0
7	Soy-notill	15	5	0	5	0	0	891	33	7	0	0	0
8	Soy-min	27	10	2	0	0	0	42	2344	0	0	0	0
9	Soy-clean	1	10	4	3	0	0	2	11	596	0	0	0
10	Wheat	0	0	0	0	0	0	0	0	0	211	0	0
11	Woods	0	0	0	1	0	0	0	0	0	0	1263	0
12	Bldg-grass-tree-drive	7	1	1	5	3	1	1	9	1	1	29	377

TABLE III
AVERAGE CLASSIFICATION ACCURACY ON HOLD-OUT SETS FOR KSC DATA

	Percentage of Available Training Data Used			
Classifier	20%	50%	75%	100%
ML	87.07 (±0.67) %	93.63 (±0.43) %	94.45 (±0.47) %	94.98 (±0.54) %
SVM	87.93 (±1.01) %	92.62 (±0.66) %	93.50 (±0.81) %	94.30 (±0.69) %
MC-SVM	90.24 (±1.60) %	94.51 (±0.77) %	96.02 (±0.29) %	96.88 (±0.27) %
GP-ML	**91.78** (±0.55) %	**97.89** (±0.28) %	**98.89** (±0.22) %	**98.87** (±0.26) %



(a)

(b)

(c)

(d)

(e)

Fig. 6. Classified maps for the KSC data, comparing groundtruth with the results of four different methods; 50% of the available training data is used. (a) Ground reference map. (b) ML. (c) SVM. (d) MC-SVM. (e) GP-ML.

Table III shows average classification accuracies with standard deviations. As in the Indian Pine experiments, the proposed GP-ML algorithm consistently performs better than other algorithms in a statistically meaningful manner. Full confusion matrices for KSC and Botswana experiments are provided in [58] as well as more detailed explanations, where the GP-ML algorithm also shows lower errors than other algorithms for most classes. The only exception is the hardwood swamp class, where the MC-SVM result shows the lowest average error. Fig. 6 shows the example classification results using 50% of the training data together with the ground reference map. Land-cover classes of KSC data set are categorized into upland classes (classes 1 to 7), wetland classes (classes 8 to 12), and the water class. Spectral signatures within the same category are more similar to each other than those in different categories, which makes it more challenging to differentiate land-cover classes within the same category. In [58], MC-SVM shows generally better classification results than ML and SVM, but it is also observable that misclassification rates between wetland classes are not much improved. GP-ML shows better overall classification accuracies across most classes, better results for mixed classes, and less confusion between wetland classes as well.

C. Botswana

The Botswana data set was obtained from the Okavango Delta by the NASA EO-1 satellite with the Hyperion sensor on May 31, 2001 [53]. The acquired data originally consisted of 242 bands, but only 145 bands are used after removing noisy and water absorption bands. The area used for experiments has 1476 × 256 pixels with 30-m spatial resolution. Fourteen different land-cover classes are used for experiments including seasonal swamps, occasional swamps, and dried woodlands. The groundtruth labels are collected using a combination of vegetation surveys, aerial photography, and a high-resolution IKONOS multispectral imagery.

Overall, classification accuracies for Botswana data are shown in Table IV. Unlike the previous cases, the GP-ML result is worse than the SVM-based results at the 20% sampling rate. It turns out that, with 20% sampling, classes 2 (Hippo grass) and 14 (exposed soil) sometimes suffer from the small sample-size problem; thus, the Gaussian ML-based methods fail for those classes. At all other sampling rates, GP-ML dominates other methods. Confusion matrices for Botswana data are also provided in [58]. ML and SVM results also show high error rates for the Ripirian class, which correspond to narrow regions along the river. MC-SVM and GP-ML both show fewer errors in these classes than nonspatial methods. GP-ML particularly shows lower misclassification rates between different types of Acacia classes. This result clearly demonstrates that the proposed GP-ML framework effectively minimizes within-class variation, which leads to better separation of land-cover classes having similar spectral responses.

D. Enhanced Dimensionality Reduction

The experimental evaluations of ML and GP-ML algorithms presented earlier were all generated using Fisher's multidimensional LDA, which has several limitations such as the number of features being upper bounded by $c - 1$. As mentioned earlier, however, the proposed framework can be combined with any dimensionality reduction technique. We employed NWFE [50] in the GP-ML framework and evaluated it using the Indian Pine data. The number of extracted features is varied from 11, the same as the number of features from the LDA algorithm, to 40. Table V compares the NWFE-based results with LDA-based ones. GP-ML results are significantly better than the baseline ML results in all cases. It is interesting to observe that NWFE results are much better than LDA results when 20% of the training data is used and less than 20 features are used. In this setting, using additional features is detrimental as there are not enough data to properly estimate the larger number of parameters, i.e., NWFE also suffers from small sample-size problem in these cases. When more data are available, the advantage of NWFE is lost, supporting our original hypothesis that LDA is quite well matched with the GP-ML framework.

V. Conclusion

We have proposed a novel framework for the classification of hyperspectral data with spatially adaptive model parameters.

TABLE IV
AVERAGE CLASSIFICATION ACCURACY ON HOLD-OUT SETS FOR BOTSWANA DATA

	Percentage of Available Training Data Used			
Classifier	20%	50%	75%	100%
ML	79.50 (±2.56) %	94.80 (±0.46) %	96.46 (±0.50) %	96.64 (±0.32) %
SVM	89.75 (±0.99) %	92.95 (±0.51) %	94.27 (±1.53) %	95.44 (±1.00) %
MC-SVM	**93.75** (±1.33) %	96.40 (±0.54) %	97.91 (±0.53) %	98.49 (±0.52) %
GP-ML	87.72 (±0.99) %	**98.21** (±0.24) %	**99.08** (±0.29) %	**99.17** (±0.06) %

TABLE V
COMPARISON OF TWO DIFFERENT FEATURE EXTRACTION ALGORITHMS USED IN CONJUNCTION WITH ML/GP-ML ON INDIAN PINE DATA [AVERAGE CLASSIFICATION ACCURACIES (WITH SD) ON HOLD-OUT SETS ARE SHOWN]

		Percentage of Available Training Data Used			
Classifier	Dim. Red. (# dim)	20%	50%	75%	100%
ML	LDA (11)	77.86 (±1.02) %	83.94 (±1.17) %	84.61 (±1.34) %	85.52 (±1.15) %
	NWFE (11)	83.02 (±1.02) %	84.87 (±1.15) %	85.26 (±1.19) %	85.44 (±1.16) %
	NWFE (20)	81.51 (±0.73) %	85.37 (±1.27) %	86.08 (±1.33) %	86.42 (±1.48) %
	NWFE (30)	76.55 (±1.23) %	85.03 (±0.99) %	86.27 (±0.86) %	86.92 (±0.98) %
	NWFE (40)	72.54 (±1.31) %	84.19 (±0.81) %	85.94 (±0.84) %	86.93 (±0.93) %
GP-ML	LDA (11)	92.87 (±0.53) %	95.97 (±0.39) %	**97.84** (±0.61) %	**98.26** (±0.31) %
	NWFE (11)	**94.71** (±0.44) %	96.57 (±0.62) %	96.93 (±0.49) %	97.33 (±0.27) %
	NWFE (20)	94.15 (±0.43) %	96.78 (±0.55) %	97.33 (±0.37) %	97.53 (±0.25) %
	NWFE (30)	89.96 (±0.97) %	**96.94** (±0.59) %	97.62 (±0.49) %	97.92 (±0.22) %
	NWFE (40)	85.49 (±0.99) %	96.69 (±0.31) %	97.70 (±0.47) %	97.97 (±0.29) %

The proposed algorithm models spatially varying means of each spectral band of a given class using a GPR model. For a given location, the predictive distribution of a given class is modeled by a multivariate Gaussian distribution with spatially adjusted parameters obtained from the proposed algorithm. Experiments on three different hyperspectral data sets show that the proposed framework performs significantly better than the baseline ML classifier, the popular SVM classifier, and prior methods that exploit spatial information.

ACKNOWLEDGMENT

The authors would like to thank M. Crawford for the collaboration of many years and for the helpful comments.

REFERENCES

[1] D. Landgrebe, "Hyperspectral image data analysis as a high dimensional signal processing problem," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 17–28, Jan. 2002.

[2] N. Cressie, *Statistics for Spatial Data*. New York: Wiley, 1993.

[3] M. Kanevski and M. Maignan, *Analysis and Modelling of Spatial Environmental Data*. Lausanne, Switzerland: EPFL Press, 2004.

[4] M. Kanevski, Ed., *Advanced Mapping of Environmental Data*. London, U.K.: ISTE Press, 2008.

[5] W. R. Tobler, "A computer movie simulating urban growth in the Detroit region," *Econ. Geogr.*, vol. 46, no. 2, pp. 234–240, 1970.

[6] A. Krause and C. Guestrin, "Nonmyopic active learning of Gaussian processes: An exploration-exploitation approach," in *Proc. 24th ICML*, 2007, pp. 449–456.

[7] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2005.

[8] G. Jun and J. Ghosh, "Spatially adaptive classification of hyperspectral data with Gaussian processes," in *Proc. Int. Geosci. Remote Sens. Symp.*, 2009, pp. II-290–II-293.

[9] E. Hestir, S. Khanna, M. Andrew, M. Santos, J. Viers, J. Greenberg, S. Rajapakse, and S. Ustin, "Identification of invasive vegetation using hyperspectral remote sensing in the California Delta ecosystem," *Remote Sens. Environ.*, vol. 112, no. 11, pp. 4034–4047, Nov. 2008.

[10] A. Plaza, J. Benediktsson, J. Boardman, J. Brazile, L. Bruzzone, G. Camps-Valls, J. Chanussot, M. Fauvel, P. Gamba, A. Gualtieri, M. Marconcini, J. C. Tilton, and G. Trianni, "Recent advances in techniques for hyperspectral image processing," *Remote Sens. Environ.*, vol. 113, pp. S110–S122, Sep. 2009.

[11] G. Camps-Valls and L. Bruzzone, "Kernel-based methods for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 6, pp. 1351–1362, Jun. 2005.

[12] L. Bruzzone and C. Persello, "A novel approach to the selection of spatially invariant features for the classification of hyperspectral images with improved generalization capability," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 9, pp. 3180–3191, Sep. 2009.

[13] M. Fauvel, J. Benediktsson, J. Chanussot, and J. Sveinsson, "Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, pt. 2, no. 11, pp. 3804–3814, Nov. 2008.

[14] D. A. Griffith, "Modeling spatial dependence in high spatial resolution hyperspectral data sets," *J. Geogr. Syst.*, vol. 4, no. 1, pp. 43–51, Mar. 2002.

[15] R. Haralick and K. Shanmugam, "Combined spectral and spatial processing of ERTS imagery data," *Remote Sens. Environ.*, vol. 3, no. 1, pp. 3–13, 1974.

[16] Y. Chen, M. Crawford, and J. Ghosh, "Knowledge based stacking of hyperspectral data for land cover classification," in *Proc. IEEE Symp. CIDM*, 2007, pp. 316–322.

[17] L. Jiménez, J. Rivera-Medina, E. Rodríguez-Díaz, E. Arzuaga-Cruz, and M. Ramírez-Vélez, "Integration of spatial and spectral information by means of unsupervised extraction and classification for homogenous objects applied to multispectral and hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 4, pp. 844–851, Apr. 2005.

[18] Y. Tarabalka, J. Benediktsson, and J. Chanussot, "Spectral-spatial classification of hyperspectral imagery based on partitional clustering techniques," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 8, pp. 2973–2987, Aug. 2009.

[19] W. Davis and F. Peet, "A method of smoothing digital thematic maps," *Remote Sens. Environ.*, vol. 6, no. 1, pp. 45–49, 1977.

[20] Q. Jackson and D. Landgrebe, "Adaptive Bayesian contextual classification based on Markov random fields," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 11, pp. 2454–2463, Nov. 2002.

[21] R. Vatsavai, S. Shekhar, and T. Burk, "An efficient spatial semi-supervised learning algorithm," *Int. J. Parallel, Emergent Distrib. Syst.*, vol. 22, no. 6, pp. 427–437, Jan. 2007.

[22] Y. Tarabalka, M. Fauvel, J. Chanussot, and J. Benediktsson, "SVM- and MRF-based method for accurate classification of hyperspectral images," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 4, pp. 736–740, Oct. 2010.

[23] G. Thoonen, S. De Backer, S. Provoost, P. Kempeneers, and P. Scheunders, "Spatial classification of hyperspectral data of dune vegetation along the Belgian coast," in *Proc. Int. Geosci. Remote Sens. Symp.*, 2008, vol. 3, pp. 483–486.

[24] Y. Bazi and F. Melgani, "Gaussian process approach to remote sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 1, pp. 186–197, Jan. 2010.

[25] K. Chen, C. Huo, Z. Zhou, H. Lu, and J. Cheng, "Semi-supervised change detection via Gaussian processes," in *Proc. Int. Geosci. Remote Sens. Symp.*, 2009, pp. II-996–II-999.

[26] L. Pasolli, F. Melgani, and E. Blanzieri, "Estimating biophysical parameters from remotely sensed imagery with Gaussian processes," in *Proc. Int. Geosci. Remote Sens. Symp.*, 2008, pp. II-851–II-854.

[27] T. Han and D. Goodenough, "Investigation of nonlinearity in hyperspectral imagery using surrogate data methods," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 10, pp. 2840–2847, Oct. 2008.

[28] P. Goovaerts, "Geostatistical incorporation of spatial coordinates into supervised classification of hyperspectral data," *J. Geogr. Syst.*, vol. 4, no. 1, pp. 99–111, Mar. 2002.

[29] M. Dundar and D. Landgrebe, "A model-based mixture-supervised classification approach in hyperspectral data analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 12, pp. 2692–2699, Dec. 2002.

[30] Y. Chen, M. Crawford, and J. Ghosh, "Applying nonlinear manifold learning to hyperspectral data for land cover classification," in *Proc. Int. Geosci. Remote Sens. Symp.*, 2005, pp. 4311–4314.

[31] Y. Chen, M. M. Crawford, and J. Ghosh, "Improved nonlinear manifold learning for land cover classification via intelligent landmark selection," in *Proc. Int. Geosci. Remote Sens. Symp.*, 2006, pp. 545–548.

[32] W. Kim, Y. Chen, M. Crawford, J. Tilton, and J. Ghosh, "Multiresolution manifold learning for classification of hyperspectral data," in *Proc. Int. Geosci. Remote Sens. Symp.*, 2007, pp. 3785–3788.

[33] W. Kim, M. Crawford, and J. Ghosh, "Spatially adapted manifold learning for classification of hyperspectral imagery with insufficient labeled data," in *Proc. Int. Geosci. Remote Sens. Symp.*, 2008, pp. I-213–I-216.

[34] S. Rajan, J. Ghosh, and M. M. Crawford, "Exploiting class hierarchies for knowledge transfer in hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 44, no. 11, pp. 3408–3417, Nov. 2006.

[35] S. Rajan, J. Ghosh, and M. M. Crawford, "An active learning approach to hyperspectral data classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 4, pp. 1231–1242, Apr. 2008.

[36] D. Tuia, F. Ratle, F. Pacifici, M. Kanevski, and W. Emery, "Active learning methods for remote sensing image classification," *IEEE Trans. Geosci. Remote Sens*, vol. 47, no. 7, pp. 2218–2232, Jul. 2009.

[37] G. Jun and J. Ghosh, "An efficient active learning algorithm with knowledge transfer for hyperspectral remote sensing data," in *Proc. Int. Geosci. Remote Sens. Symp.*, 2008, pp. I-52–I-55.

[38] G. Jun, R. R. Vatsavai, and J. Ghosh, "Spatially adaptive classification and active learning of multispectral data with Gaussian processes," in *Proc. SSTDM/ICDM Workshop*, 2009, pp. 597–603.

[39] B. Ripley, *Spatial Statistics*. New York: Wiley, 1981.

[40] G. Matheron, "Principles of statistics," *Econ. Geol.*, vol. 58, pp. 1246–1266, 1963.

[41] G. Matheron, "The intrinsic random functions and their applications," *Adv. Appl. Probab.*, vol. 5, no. 3, pp. 439–468, 1973.

[42] A. Fotheringham, C. Brunsdon, and M. Charlton, *Geographically Weighted Regression: The Analysis of Spatially Varying Relationships*. New York: Wiley, 2002.

[43] P. Harris, A. Fotheringham, R. Crespo, and M. Charlton, "The use of geographically weighted regression for spatial prediction: An evaluation of models using simulated data sets," *Math. Geosci.*, vol. 42, no. 6, pp. 657–680, Aug. 2010.

[44] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: Wiley-Interscience, 2000.

[45] C. Rasmussen, "Evaluation of Gaussian processes and other methods for non-linear regression," Ph.D. dissertation, Univ. Toronto, Toronto, ON, Canada, 1996.

[46] D. Hand and K. Yu, "Idiot's Bayes: Not so stupid after all?" *Int. Stat. Rev./Revue Internationale de Statistique*, vol. 69, no. 3, pp. 385–398, 2001.

[47] A. Berge, A. Jensen, and A. Solberg, "Sparse inverse covariance estimates for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 5, pp. 1399–1407, May 2007.

[48] A. Jensen, A. Berge, and A. Solberg, "Regression approaches to small sample inverse covariance matrix estimation for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 10, pp. 2814–2822, Oct. 2008.

[49] C. Lee and D. Landgrebe, "Feature extraction based on decision boundaries," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 4, pp. 388–400, Apr. 1993.

[50] B.-C. Kuo and D. Landgrebe, "Nonparametric weighted feature extraction for classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 5, pp. 1096–1105, May 2004.

[51] S. Tadjudin and D. Landgrebe, "Covariance estimation with limited training samples," *IEEE Trans. Geosci. Remote Sens.*, vol. 37, no. 4, pp. 2113–2118, Jul. 1999.

[52] J. T. Morgan, "Adaptive hierarchical classifier with limited training data," Ph.D. dissertation, Univ. Texas, Austin, TX, 2002.

[53] J. Ham, Y. Chen, M. M. Crawford, and J. Ghosh, "Investigation of the random forest framework for classification of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 492–501, Mar. 2005.

[54] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, Aug. 2004.

[55] R. Rifkin and A. Klautau, "In defense of one-vs-all classification," *J. Mach. Learn. Res.*, vol. 5, pp. 101–141, 2004.

[56] P. Hsieh and D. Landgrebe, "Classification of high dimensional data," Ph.D. dissertation, School Elect. Comput. Eng., Purdue Univ., West Lafayette, IN, 1998, Tech. Rep. TR-ECE.

[57] J. Nascimento and J. Dias, "Does independent component analysis play a role in unmixing hyperspectral data?" *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 1, pp. 175–187, Jan. 2005.

[58] G. Jun and J. Ghosh, Confusion matrices for KSC and Botswana data, *Supplementary Matrial*. [Online]. Available: http://www.ideal.ece.utexas.edu/pubs/pdf/2010/GPMLconf.pdf

Goo Jun (S'06–M'11) received the B.S. degree from the Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 1997, the M.S. degree from the University of Michigan, Ann Arbor, in 1999, and the Ph.D. degree in electrical and computer engineering from The University of Texas at Austin, Austin, in 2010.

From 1999 to 2005, he was with Samsung Electronics, Suwon, Korea, as a Research Engineer. He is currently a Research Fellow with Biostatistics Department, University of Michigan.

Joydeep Ghosh (S'87–M'88–SM'02–F'06) received the B.Tech. degree from the Indian Institute of Technology Kanpur in 1983 and the Ph.D. degree from the University of Southern California in 1988.

He is currently the Schlumberger Centennial Chair Professor with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, where he has been with the faculty since 1988. He has published more than 250 refereed papers and 35 book chapters, coedited 20 books, and received 14 "best paper" awards.