A Unified Framework for MAP Estimation in Remote Sensing Image Segmentation

Aly A. Farag, *Senior Member, IEEE*, Refaat M. Mohamed, *Student Member, IEEE*, and Ayman El-Baz, *Student Member, IEEE*

***Abstract*—A complete framework is proposed for applying the maximum *a posteriori* (MAP) estimation principle in remote sensing image segmentation. The MAP principle provides an estimate for the segmented image by maximizing the posterior probabilities of the classes defined in the image. The posterior probability can be represented as the product of the class conditional probability (CCP) and the class prior probability (CPP). In this paper, novel supervised algorithms for the CCP and the CPP estimations are proposed which are appropriate for remote sensing (RS) images where the estimation process might to be done in high-dimensional spaces. For the CCP, a supervised algorithm which uses the support vector machines (SVM) density estimation approach is proposed. This algorithm uses a novel learning procedure, derived from the main field (MF) theory, which avoids the (hard) quadratic optimization problem arising from the traditional formulation of the SVM density estimation. For the CPP estimation, Markov random field (MRF) is a common choice which incorporates contextual and geometrical information in the estimation process. Instead of using predefined values for the parameters of the MRF, an analytical algorithm is proposed which automatically identifies the values of the MRF parameters. The proposed framework is built in an iterative setup which refines the estimated image to get the optimum solution. Experiments using both synthetic and real remote sensing data (multispectral and hyperspectral) show the powerful performance of the proposed framework. The results show that the proposed density estimation algorithm outperforms other algorithms for remote sensing data over a wide range of spectral dimensions. The MRF modeling raises the segmentation accuracy by up to 10% in remote sensing images.**

***Index Terms*—Image modeling, Markov random field (MRF), parameters estimation, segmentation, support vector machines (SVM).**

I. Introduction

REMOTE sensing image analysis is attracting a growing interest in real world applications such as urban planning, mapping, agriculture, forestry, and disaster prevention and monitoring. This interest is due to a number of factors including the following: the higher quality and larger quantity of information that can be extracted through the analysis of remote sensing images; the continuous improvement of the satellite remote sensor's characteristics; and with the recent availability of commercial high-resolution satellite multispectral imagery from sensors such as IKONOS and QuickBird. To accommodate such a growing interest, high quality segmented images are required for significant exploitation of the remote sensing features. Thus, there is a continuous demand for sophisticated segmentation approaches so that significantly accurate segmentation of images from high-resolution remote sensing images can be obtained. Usually, the exact segmented image is difficult to obtain, instead the segmentation algorithms aim to get an estimate of the segmented image.

During the past couple of decades, extensive work had been done toward the application of pattern recognition methods in remote sensing data analysis, [1]–[3]. Many of these methods extend the classical methods for pattern recognition in low-dimensional space to high-dimensional. Some of these methods are *parametric*, meaning they assume a model for the class conditional density functions, while others are *nonparametric*, meaning they do not assume such a model.

In general, classification methods fall into two main categories, unsupervised and supervised methods. The classification problem is solved without the need for training samples in unsupervised classification. In [4], an unsupervised classification algorithm is derived by modeling observed data as a mixture of several mutually exclusive classes that are each described by linear combinations of independent, non-Gaussian densities. The algorithm estimates the density in each class by using parametric nonlinear functions that fit to the non-Gaussian structure of the data. An unsupervised texture segmentation algorithm based on feature extraction using multichannel Gabor filtering is presented in [5]. The main idea of this segmentation algorithm is to decompose the actual segmented image into disjunct areas called scrap images and use them after lowpass filtering as additional features for repeated k-means clustering and minimum distance classification. In [6], an unsupervised approach is proposed for identification of changes in multitemporal remote sensing images. This approach is based on the formulation of the unsupervised change-detection problem in terms of the Bayesian decision theory. In this context, an adaptive semiparametric technique for the unsupervised estimation of the statistical terms associated with the gray levels of changed and unchanged pixels in a difference image is presented. Other examples for unsupervised classification algorithms can be found in [7], [8]. In supervised classification [9]–[11], a training sample is used by a learning machine to extract the information regarding the different classes of the classification problem. This information is then used in the classification of unseen data points. A comparison between the two categories can be found in [12]. The proposed framework in this paper falls into the supervised category with statistical-based building.

Manuscript received September 2, 2004; revised March 22, 2005. This work was supported by the Air Force Office of Scientific Research (AFOSR) under Grant F49620-01-1-0367.

The authors are with the Computer Vision and Image Processing Laboratory, Department of Electrical and Computer Engineering, University of Louisville, Louisville, KY 40292 USA (e-mail: farag@cvip.uofl.edu).

Digital Object Identifier 10.1109/TGRS.2005.849059

Among the themes of the statistical segmentation algorithms, one main direction is to maximize the posterior probability of each class defined in the original image [maximum *a posteriori* (MAP) estimation]. By Bayes' rule, the posterior probability of each class is defined in terms of the class conditional probability (CCP) and the class prior probability (CPP) [9]. In this paper, the algorithms needed to build each component of the MAP estimation setup are proposed.

Support vector machines (SVM) were developed by Vapnik [13] to solve the classification problem, but recently, SVM have been successfully extended to regression and density estimation problems [14]. SVM are gaining popularity due to many attractive features and promising empirical performance. For instance, the formulation of SVM density estimation employs the structural risk minimization (SRM) principle which has been shown to be superior to the traditional empirical risk minimization (ERM) principle employed in conventional learning algorithms (e.g., neural networks) [15]. SRM minimizes an upper bound on the generalization error as opposed to ERM which minimizes the error on the training data. The SRM characteristic gives the SVM the ability for global minima search. Since, the optimization process is done in the feature space (after mapping the input through the kernel) makes the SVM more robust to input space dimensions. These differences makes SVM more attractive in statistical learning applications. In this paper, the SVM is used as a supervised nonparametric estimator for the CCP.

The traditional formulation of the SVM density estimation problem raises a quadratic optimization problem of the same size as the training dataset. This computationally demanding optimization problem prevents the SVM from being the default choice of the pattern recognition community because solving such optimization problem is not trivial, if it is at all possible [16]. In this paper, an algorithm based on the mean field (MF) theory will be used in training the SVM estimator. The MF methods provide efficient approximations which are able to cope with the complexity of probabilistic data models [17]. MF methods replace the intractable task of computing high-dimensional sums and integrals by the much easier problem of solving a system of linear equations. The CCP estimation problem is formulated so that the MF method can be used to approximate the learning procedure in a way that avoids the quadratic programming optimization. This proposed approach is suitable for high-dimensional density estimation problems and it is successfully applied to various remote sensing datasets.

Generally, no accurate segmentation can be achieved using marginal probability distributions alone (CCP and CPP) because in most cases signal ranges for different classes overlap. Typically, a segmented image after initial pixelwise classification is further refined by optimal statistical estimation of a hidden model of the segmented regions. The Markov random field (MRF) is a natural choice for implementing the hidden model for the segmented regions since MRF is the best way for incorporating spatial correlations into a segmentation process. The refinement of the segmented image using MRF modeling for the regions can be considered as a refinement of the CPP [18]. The images (raw data) and segmented regions are specified with a joint Markov model that combines an unconditional model of interdependent region labels and a conditional model of independent image signals in each region [19]–[21]. The initial segmented image is then iteratively refined by using the MRF model. In principle, this paper follows this conventional scheme but in contrast to previous solutions, [22], this paper focuses on the most accurate identification of this region model. The intra- and interregion label cooccurrences are specified by a MRF model with the nearest neighbors of each pixel. Under the assumed symmetric relationships between the neighboring labels, the model resembles the conventional autobinomial ones [23]. But rather than using the (empirically) predefined Gibbs parameters, an algorithm is proposed for the identification of the MRF parameters which uses an analytical model derived in accordance with the work of Gimel'farb; see [21].

This paper features some contributions which add up to build a successful segmentation procedure for the remote sensing image segmentation. Among these contributions is the presentation of the MF-based SVM density estimator which has been proven to be a successful density estimation approach in multidimensional spaces. The simple and efficient analytical approach for the estimation of the MRF parameters is another contribution. The iterative setup of the proposed framework enables sequential evolution of the model estimation process to get a maximized estimate for the segmented image.

The rest of this paper is organized as follows. Section II presents an overview for image modeling. It emphasizes the main components which are required for implementing the MAP segmentation approach. The proposed SVM-based approach for estimating the CCP is presented in Section III while Section IV presents the proposed analytical method which is used for the estimation of the MRF parameters. The proposed iterative setup for the framework is discussed in Section V. Experimental work using both synthetic data and real remote sensing images is presented in Section VI while the proposed work is concluded in Section VII.

II. Image Modeling and MAP Estimation

The MAP estimate of the segmented image $(\mathbf{X})$, given the observed image $(\mathbf{Y})$, involves the determination of $\mathbf{X}$ that maximizes $\mathbf{P}(\mathbf{X}=\mathbf{x} \mid \mathbf{Y}=\mathbf{y})$ with respect to $\mathbf{X}$. By Bayes' rule

$$p(\mathbf{X}=\mathbf{x} \mid \mathbf{Y}=\mathbf{y}) = \frac{p(\mathbf{Y}=\mathbf{y} \mid \mathbf{X}=\mathbf{x})p(\mathbf{X}=\mathbf{x})}{p(\mathbf{Y}=\mathbf{y})}. \quad (1)$$

The denominator of (1) does not affect the optimization required to obtain an estimate of $\mathbf{X}$. Thus, generally the MAP estimate can be obtained from

$$\mathbf{X}^* = \arg\max_{\mathbf{x}\in\mathcal{X}} \Gamma(\mathbf{X},\mathbf{Y}) \quad (2)$$

where $\mathcal{X}$ is the set of all region maps with labels $k \in \mathbf{K}$ on the lattice $\mathbf{S}$ and

$$\Gamma(\mathbf{X},\mathbf{Y}) = \frac{1}{|\mathbf{S}|}\big(\log p(\mathbf{Y} \mid \mathbf{X}) + \log p(\mathbf{X})\big). \quad (3)$$

The first term in (3) is the likelihood for the conditional distribution of an image given its segmented image, i.e., class conditional probability "CCP." The second term is the unconditional distribution of the segmented image which can be considered as a variation of the class prior probability "CPP." The Bayesian

MAP estimate can be obtained based on the models of the CCP and CPP. The following two sections describe the proposed algorithms for model identification of both the MAP estimator elements.

III. Class Conditional Probability Estimation

In this paper, the CCP estimation is divided into a couple of estimation problems where each class probability density function is estimated independently from the other classes. In this way, the training sample is divided into subsets where each set belongs to a specific class. The density estimation of each class is then treated as a regular probability density estimation problem. In this section, the outlines of the density estimation problem are presented, then a supervised density estimation algorithm which is based on the SVM approach is presented.

A. Density Estimation Problem Formulation

Given a random vector $\mathbf{Y}$, the relation

$$F(\mathbf{y}) = P(\mathbf{Y} < \mathbf{y}) \tag{4}$$

defines the cumulative probability distribution function, CDF, of the random vector $\mathbf{Y}$. The probability density function, PDF, $p(\mathbf{y})$, of the random vector $\mathbf{Y}$ at a specific point $\mathbf{y}$ is a nonnegative quantity and it is related to the CDF by the relation

$$F(\mathbf{y}) = \int_{-\infty}^{\mathbf{y}} p(\mathbf{y}')\, d\mathbf{y}'. \tag{5}$$

Hence, in order to estimate the probability density function it is required to obtain a solution for the inverse of the integral equation

$$\int_{-\infty}^{\mathbf{y}} p(\mathbf{y}',\boldsymbol{\alpha})\, d\mathbf{y}' = F(\mathbf{y}) \tag{6}$$

on a given set of densities $p(\mathbf{y},\boldsymbol{\alpha})$. Where, the integration is a vector integration, and $\boldsymbol{\alpha}$ is the parameter set which characterizes the density function.

From another point of view, the estimation problem in (6) can be regarded as solving the linear operator equation

$$\boldsymbol{A}[p(\mathbf{y})] = F(\mathbf{y}) \tag{7}$$

where the operator $\boldsymbol{A}[.]$ is a one-to-one mapping for the elements of the Hilbert space $\boldsymbol{E_1}$ where $p(\mathbf{y})$ is defined into elements of the Hilbert space $\boldsymbol{E_2}$ where $F(\mathbf{y})$ is defined. But, neither $p(\mathbf{y})$ nor $F(\mathbf{y})$ in (7) is known. However, from the principles of probability theory [24], given a random sample $\mathcal{D} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n\}$ from an unknown distribution, a practical estimate for $F(\mathbf{y})$ can be obtained by

$$F_n(\mathbf{y}) = \frac{1}{n}\sum_{k=1}^{n} I_{(-\infty,\, \mathbf{y}]}(\mathbf{y}_k) \tag{8}$$

where, n is the size of the sample and $I_{(-\infty,\, \mathbf{y}]}(u)$ is the indicator function which is defined as

$$I_{(-\infty,\, y]}(u) = \begin{cases} 1, & \text{if } u \le y \\ 0, & \text{else} \end{cases} \tag{9}$$

if both of y and u are scalars [one-dimensional (1-D) data]. If $\mathbf{y}$ and $\mathbf{u}$ are vectors of length d, then

$$I_{(-\infty,\, \mathbf{y}]}(\mathbf{u}) = \prod_{i=1}^{d} I_{(-\infty,\, y_i]}(u_i). \tag{10}$$

This function $F_n(\mathbf{y})$, which is called the empirical distribution function, converges (at least weakly) to the original distribution function $F(\mathbf{y})$ [25]. Therefore, the pairs $(\mathbf{y}_1, F_n(\mathbf{y}_1)), (\mathbf{y}_2, F_n(\mathbf{y}_2)), \ldots, (\mathbf{y}_n, F_n(\mathbf{y}_n))$ are constructed from the sample $\mathcal{D}$ to generate the training dataset

$$\mathfrak{D} = \{(\mathbf{y}_i, t_i) \mid t_i = F_n(\mathbf{y}_i); i = 1, 2, \ldots, n\}. \tag{11}$$

Now, a regression algorithm uses this training dataset to solve the density estimation problem [(7)] in the image space [right-hand side of (7)] to get *a continuous* approximation for the distribution function $F(\mathbf{y})$. This approximation can be used to express the solution in the preimage space [left-hand side of (7)] to get an estimate for the density function using the known operator $\boldsymbol{A}$. In this paper, the SVM is used as a regression algorithm to get a continuous approximation for the distribution function $F(\mathbf{y})$. The motivation behind using the SVM as a regression tool in this paper is that a dense continuous approximation for $F(\mathbf{y})$ is obtained which should be *safely differentiable* so that the density function $p(\mathbf{y})$ can be obtained.

B. Support Vector Machines Regression

The above discussion shows how the *supervised* density estimation problem is reduced to a regression problem. In this section, the SVM is presented as a supervised regression tool and later on, it will be shown how can it be used as a density estimator for the CCP. In the following discussion, the SVM as a regression tool is considered as the maximum *a posteriori* prediction with a Gaussian prior, under the Bayesian framework (Bayes' theorem is used to relate the prior and posterior distributions). The idea is that, instead of defining prior distributions over parameters of the learning machine, a Gaussian prior distribution is assumed over the function space on which the machine computes. In general, the supervised regression learning problem can be stated as follows.

Given a training dataset $\mathfrak{D} = \{(\mathbf{y}_i, t_i) \mid i = 1, 2, \ldots, n\}$, of input vectors $\mathbf{y}_i$'s and associated targets t_i's, the goal is to infer the output t for a new input data point $\mathbf{y}$. Generally, a loss function which relates the estimated target $g(\mathbf{y})$ and the true target t is defined to characterize the regression problem. In this paper, the Vapnik's loss function is used which is defined as

$$\mathcal{L}(t, g(\mathbf{y})) = \begin{cases} 0, & \text{if } |t - g(\mathbf{y})| \le \epsilon \\ |t - g(\mathbf{y})| - \epsilon, & \text{otherwise} \end{cases} \tag{12}$$

where $\epsilon > 0$ is a predefined constant which controls the noise tolerance.

To construct a Bayesian framework under the assumed loss function in (12), an exponential model is employed. In this model, the likelihood for the probability of the true output t at

a given point $\mathbf{y}$, providing that the machine output is $\mathbf{g}(\mathbf{y})$, is assumed by the following relationship:

$$p\left(t \mid g\left(\mathbf{y}\right)\right) = \frac{C}{2\left(\epsilon C+1\right)} \exp\left\{-C\mathcal{L}(t, g\left(\mathbf{y}\right))\right\}. \quad (13)$$

Since the elements of the training sample are assumed to be statistically independent random vectors, the probabilistic interpretation of the SVM regression can be considered to have the following likelihood (see [17]):

$$p(\tau \mid \mathbf{g}(\mathcal{D})) = \left(\frac{C}{2\left(\epsilon C+1\right)}\right)^{n} \exp\left\{-C\sum_{i=1}^{n}\mathcal{L}(t_i, g(\mathbf{y}_i))\right\} \quad (14)$$

where $\tau = [t_1, t_2, \ldots, t_n]$ and $\mathbf{g}\left(\mathcal{D}\right) = [g\left(\mathbf{y}_1\right), g\left(\mathbf{y}_2\right), \ldots, g\left(\mathbf{y}_n\right)]$.

Since, the SVM is considered as a MAP probability estimator with a Gaussian prior, the prior probability distribution of the prediction $g\left(\mathbf{y}\right)$ is assumed as a Gaussian process (GP). Generally, a GP is a stochastic process which is completely specified by its mean vector and covariance matrix [26]. Thus, the prior probability for a sample $\mathcal{D}$ can be expressed as a GP with zero mean (for simplicity) and a covariance function $\mathcal{K}\left(\mathbf{y}, \mathbf{y}'\right)$ as

$$p(\mathbf{g}(\mathcal{D})) = \frac{1}{\sqrt{(2\pi)^{n}\, det\left(\mathcal{K}_n\right)}} \exp\left\{-\frac{1}{2}\mathbf{g}\left(\mathcal{D}\right)\mathcal{K}_n^{-1}\mathbf{g}\left(\mathcal{D}\right)^{t}\right\} \quad (15)$$

where $\mathcal{K}_n = [\mathcal{K}\left(\mathbf{y}_i, \mathbf{y}_j\right)]$ is the covariance matrix at the points of $\mathcal{D}$ ($\mathcal{K}(.,.)$ is a kernel function). This can be parameterized with respect to $\mathcal{D}$ by Bayes' theorem [see (16), shown at the bottom of the page] where $\mathcal{M} = \left(C/2\left(\epsilon C+1\right)\right)^{n}$. The estimate of the posterior prediction distribution is the one that maximizes the numerator of (16). Equivalently, the MAP estimate is obtained from

$$\min_{\mathbf{g}(\mathcal{D})} C\sum_{i=1}^{n}\mathcal{L}(t_i, g(\mathbf{y}_i)) + \frac{1}{2}\mathbf{g}(\mathcal{D})\mathcal{K}_n^{-1}\mathbf{g}(\mathcal{D})^{t}. \quad (17)$$

Direct solution of (17) can be obtained by quadratic programming optimization (e.g., [13]). The size of the optimization problem is the same as the size of the training sample. Since quadratic programming (QP) Optimization routines have high complexity and require huge memory and computational time for large data applications, solving the QP, especially with a dense $n \times n$ matrix, limits the use of the SVM algorithm for large datasets (e.g., [16]). One way to avoid raising such a QP problem is to consider an approximate formulation for the SVM regression [27]. The rest of this section and the following section present one of such methods.

Using the posterior prediction distribution $p\left(\mathbf{g}\left(\mathcal{D}\right) \mid \mathcal{D}\right)$ which is defined in (16), the prediction (expectation) on a new test point $\mathbf{y}$ is given by

$$\begin{aligned}\langle g\left(\mathbf{y}\right)\rangle &= \int g\left(\mathbf{y}\right) p\left(g\left(\mathbf{y}\right) \mid \mathcal{D}\right) dg(\mathbf{y}) \\ &= \int g(\mathbf{y})\, p\left(g\left(\mathbf{y}\right), \mathbf{g}\left(\mathcal{D}\right) \mid \mathcal{D}\right) dg(\mathbf{y})\, d\mathbf{g}(\mathcal{D}). \quad (18)\end{aligned}$$

Substituting from (16) into (18) and with some mathematical reduction

$$\langle g\left(\mathbf{y}\right)\rangle = \frac{\mathcal{M}}{\sqrt{(2\pi)^{n}\, det\left(\mathcal{K}_n\right)}} \int g\left(\mathbf{y}\right)\mathcal{A}\, dg(\mathbf{y})\, d\mathbf{g}(\mathcal{D}) \quad (19)$$

where

$$\mathcal{A} = \frac{\exp\left\{-C\sum_{i=1}^{n}\mathcal{L}(t_i, g(\mathbf{y}_i)) - \frac{1}{2}\mathbf{g}(\mathcal{D}, \mathbf{y})\mathcal{K}_{n+1}^{-1}\mathbf{g}(\mathcal{D}, \mathbf{y})^{t}\right\}}{p\left(\mathcal{D}\right)}$$

$$\mathcal{K}_{n+1} = \begin{pmatrix} \mathcal{K}_n & \mathcal{K}_n\left(\mathbf{y}\right)^{t} \\ \mathcal{K}_n\left(\mathbf{y}\right) & \mathcal{K}\left(\mathbf{y}, \mathbf{y}\right)\end{pmatrix}$$

$$\mathcal{K}_n(\mathbf{y}) = \left[\mathcal{K}\left(\mathbf{y}_1, \mathbf{y}\right), \mathcal{K}\left(\mathbf{y}_2, \mathbf{y}\right), \ldots, \mathcal{K}\left(\mathbf{y}_n, \mathbf{y}\right)\right].$$

But

$$\mathbf{g}(\mathcal{D})p(\mathbf{g}(\mathcal{D})) = \mathcal{K}_n{\mathcal{K}_n}^{-1}p(\mathbf{g}(\mathcal{D})) = -\mathcal{K}_n\frac{\partial}{\partial\, \mathbf{g}(\mathcal{D})}p(\mathbf{g}(\mathcal{D})).$$

Then by extending the prior to include the new point (test point $\mathbf{y}$) we get

$$\begin{aligned}&g\left(\mathbf{y}\right)\, \exp\left\{-\frac{1}{2}\mathbf{g}(\mathcal{D}, \mathbf{y})\mathcal{K}_{n+1}^{-1}\mathbf{g}(\mathcal{D}, \mathbf{y})^{t}\right\} \\ &= \sum_{i=1}^{n+1}\mathcal{K}(\mathbf{y}, \mathbf{y}_i)\frac{\partial}{\partial\, g(\mathbf{y}_i)}\exp\left\{-\frac{1}{2}\mathbf{g}(\mathcal{D}, \mathbf{y})\mathcal{K}_{n+1}^{-1}\mathbf{g}(\mathcal{D}, \mathbf{y})^{t}\right\}. \quad (20)\end{aligned}$$

Substituting from (20) into (19) and apply integration by parts to shift the differentiation from the prior to the likelihood, then we have (21), shown at the bottom of the page, where w_i is a constant defined as

$$\begin{aligned}w_i = &\frac{\mathcal{M}}{P\left(\mathcal{D}\right)} \int \mathcal{N}\left(\mathbf{g}\left(\mathcal{D}\right) \mid \mathbf{0}, \mathcal{K}_n\right) g\left(\mathbf{y}\right) \\ &\cdot \frac{\partial}{\partial g\left(\mathbf{y}_i\right)}\, \exp\left\{-C\sum_{j=1}^{n}\mathcal{L}(t_j, g(\mathbf{y}_j))\right\}\, d\mathbf{g}\left(\mathcal{D}\right). \quad (22)\end{aligned}$$

$$p(\mathbf{g}\left(\mathcal{D}\right) \mid \mathcal{D}) = \frac{p\left(\mathcal{D} \mid \mathbf{g}\left(\mathcal{D}\right)\right) p\left(\mathbf{g}\left(\mathcal{D}\right)\right)}{p\left(\mathcal{D}\right)} = \frac{\mathcal{M}\exp\left\{-C\sum_{i=1}^{n}\mathcal{L}(t_i, g(\mathbf{y}_i)) - \frac{1}{2}\mathbf{g}(\mathcal{D})\,\mathcal{K}_n^{-1}\mathbf{g}\left(\mathcal{D}\right)^{t}\right\}}{\sqrt{(2\pi)^{n}\, det\left(\mathcal{K}_n\right)}\, p\left(\mathcal{D}\right)} \quad (16)$$

$$\langle g\left(\mathbf{y}\right)\rangle = \frac{\mathcal{M}}{P\left(\mathcal{D}\right)}\sum_{i=1}^{n}\mathcal{K}\left(\mathbf{y}, \mathbf{y}_i\right)\int \mathcal{N}\left(\mathbf{g}\left(\mathcal{D}\right) \mid \mathbf{0}, \mathcal{K}_n\right) g\left(\mathbf{y}\right) \cdot \frac{\partial}{\partial g\left(\mathbf{y}_i\right)}\, \exp\left\{-C\sum_{j=1}^{n}\mathcal{L}(t_j, g(\mathbf{y}_j))\right\}\, d\mathbf{g}\left(\mathcal{D}\right) = \sum_{i=1}^{n} w_i\, \mathcal{K}\left(\mathbf{y}, \mathbf{y}_i\right) \quad (21)$$

C. Mean Field Theory for SVM Learning

The learning process suggests that the weights w_i's in (22) should be estimated using the training sample. But as can be seen, (22) is highly complicated and computationally expensive since it contains a lot of integrations which need to be evaluated numerically. In this paper, the mean field theory is used to get an approximate and easy expression for w_i's. The basic idea of the mean field theory is to approximate the statistics of a random variable which is correlated to other random variables by assuming that the influence of the other variables can be compressed into a single effective mean "field" with a rather simple distribution [17]. In this paper, this approach is used to approximate the so called *cavity distribution*. The cavity distribution is defined as $p\left(g\left(\mathbf{y}_i\right) \mid \overline{\mathcal{D}}\right)$, where $g\left(\mathbf{y}_i\right)$ is the regressed SVM output corresponding to an instant $\mathbf{y}_i$ which is *left* from the training sample, and $\overline{\mathcal{D}}$ is the training sample without the instant $\mathbf{y}_i$.

For the cavity derivation, it is useful to introduce a new predictive posterior for the output corresponding to the instant $\mathbf{y}_i$ as

$$p\left(g\left(\mathbf{y}_i\right) \mid \overline{\mathcal{D}}\right) = \frac{\int p(\mathbf{g}(\mathcal{D}))p(\overline{\boldsymbol{\tau}} \mid \mathbf{g}(\overline{\mathcal{D}}))\, d\mathbf{g}(\overline{\mathcal{D}})}{\int p(\mathbf{g}(\mathcal{D}))p(\overline{\boldsymbol{\tau}} \mid \mathbf{g}(\overline{\mathcal{D}}))\, d\mathbf{g}(\mathcal{D})} \tag{23}$$

where $\overline{\boldsymbol{\tau}}$ is the target vector $\boldsymbol{\tau}$ excluding t_i.

For the predictive posterior in (23), an average (expected value) can be defined as

$$\langle \mathcal{V} \rangle_i = \int \mathcal{V}\, p\left(g\left(\mathbf{y}_i\right) \mid \overline{\mathcal{D}}\right)\, dg(\mathbf{y}_i) \tag{24}$$

where $\langle \mathcal{V} \rangle_i$ denotes the expected value for $\mathcal{V}$ given only the data sample $\overline{\mathcal{D}}$. Then the expression for the weight w_i in (22) can be rewritten as

$$w_i = \frac{\left\langle \mathcal{M} \frac{\partial}{\partial g(\mathbf{y}_i)} \exp\left\{-C\mathcal{L}(t_j, g(\mathbf{y}_j))\right\} \right\rangle_i}{\left\langle \mathcal{M} \exp\left\{-C\mathcal{L}(t_j, g(\mathbf{y}_j))\right\} \right\rangle_i}. \tag{25}$$

To enable the weight's calculation from (25), a closed form for the cavity distribution in (23) is required, and that is where the mean field theory comes to play. The MF considers that it is possible to calculate averages over $p\left(g\left(\mathbf{y}_i\right) \mid \overline{\mathcal{D}}\right)$ because of the fact that it is a predictive posterior of the field at an input $\mathbf{y}_i$. The MF approximates $p\left(g\left(\mathbf{y}_i\right) \mid \overline{\mathcal{D}}\right)$ with a Gaussian distribution in the form

$$p\left(g\left(\mathbf{y}_i\right) \mid \overline{\mathcal{D}}\right) \approx \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left\{-\frac{\left(g\left(\mathbf{y}_i\right) - \langle g\left(\mathbf{y}_i\right)\rangle_i\right)^2}{2\sigma_i^2}\right\} \tag{26}$$

where the variance is defined as $\sigma_i^2 = \langle g\left(\mathbf{y}_i^2\right)_i \rangle - \langle g\left(\mathbf{y}_i\right)\rangle_i^2$.

Inserting (26) into (24) and evaluating (25), the weight coefficients can be obtained as

$$w_i \approx \frac{\mathcal{F}\left(\langle g\left(\mathbf{y}_i\right)\rangle_i, \sigma_i^2\right)}{\mathcal{G}\left(\langle g\left(\mathbf{y}_i\right)\rangle_i, \sigma_i^2\right)} = \frac{\mathcal{F}_i}{\mathcal{G}_i} \tag{27}$$

where

$$\begin{aligned}
\mathcal{F}_i = {} & \frac{C}{2}\exp\left\{\frac{C}{2}\left(2\langle g\left(\mathbf{y}_i\right)\rangle_i - 2t_i + 2\varepsilon + C\sigma_i^2\right)\right\} \\
& \times \left(1 - \operatorname{erf}\left\{\frac{\langle g\left(\mathbf{y}_i\right)\rangle_i - t_i + \varepsilon + C\sigma_i^2}{\sqrt{2\sigma_i^2}}\right\}\right) \\
& - \frac{C}{2}\exp\left\{\frac{C}{2}\left(-2\langle g\left(\mathbf{y}_i\right)\rangle_i + 2t_i + 2\varepsilon + C\sigma_i^2\right)\right\} \\
& \times \left(1 - \operatorname{erf}\left\{\frac{-\langle g\left(\mathbf{y}_i\right)\rangle_i + t_i + \varepsilon + C\sigma_i^2}{\sqrt{2\sigma_i^2}}\right\}\right) \\
\mathcal{G}_i = {} & \frac{1}{2}\operatorname{erf}\left\{\frac{t_i - \langle g\left(\mathbf{y}_i\right)\rangle_i + \varepsilon}{\sqrt{2\sigma_i^2}}\right\} \\
& - \frac{1}{2}\operatorname{erf}\left\{\frac{t_i - \langle g\left(\mathbf{y}_i\right)\rangle_i - \varepsilon}{\sqrt{2\sigma_i^2}}\right\} \\
& + \frac{C}{2}\exp\left\{\frac{C}{2}\left(2\langle g\left(\mathbf{y}_i\right)\rangle_i - 2t_i + 2\varepsilon + C\sigma_i^2\right)\right\} \\
& \times \left(1 - \operatorname{erf}\left\{\frac{\langle g\left(\mathbf{y}_i\right)\rangle_i - t_i + \varepsilon + C\sigma_i^2}{\sqrt{2\sigma_i^2}}\right\}\right) \\
& - \frac{C}{2}\exp\left\{\frac{C}{2}\left(-2\langle g\left(\mathbf{y}_i\right)\rangle_i + 2t_i + 2\varepsilon + C\sigma_i^2\right)\right\} \\
& \times \left(1 - \operatorname{erf}\left\{\frac{-\langle g\left(\mathbf{y}_i\right)\rangle_i + t_i + \varepsilon + C\sigma_i^2}{\sqrt{2\sigma_i^2}}\right\}\right).
\end{aligned} \tag{28}$$

Equations (27) and (28) are called the mean field equations corresponding to the weight coefficient w_i. To evaluate the weight coefficients in (27), it is required to get both the mean (average) $\langle g\left(\mathbf{y}_i\right)\rangle_i$ and the variance σ_i^2 of the assumed Gaussian model for the local predictive distribution $p\left(g\left(\mathbf{y}_i\right) \mid \overline{\mathfrak{D}}\right)$. The detailed derivation for both $\langle g\left(\mathbf{y}_i\right)\rangle_i$ and σ_i^2 depending on the mean field theory can be found in [17], but only the final results are summarized here. The posterior average at $\mathbf{y}_i$ is given by

$$\langle g\left(\mathbf{y}_i\right)\rangle = \sum_{j=1}^{n} w_j\, \mathcal{K}\left(\mathbf{y}_i, \mathbf{y}_j\right). \tag{29}$$

From [17], the following results are obtained:

$$\langle g\left(\mathbf{y}_i\right)\rangle_i \approx \langle g\left(\mathbf{y}_i\right)\rangle - \sigma_i^2 w_i \tag{30}$$

and

$$\sigma_i^2 \approx \frac{1}{\left[\left(\Sigma + \mathcal{K}\right)^{-1}\right]_{ii}} - \Sigma_i \tag{31}$$

where

$$\Sigma = \operatorname{diag}\left(\Sigma_1, \Sigma_2, \ldots, \Sigma_n\right) \qquad \Sigma_i = -\sigma_i^2 - \left(\frac{\partial w_i}{\partial \langle g\left(\mathbf{y}_i\right)\rangle_i}\right)^{-1}$$

The expression for $\partial w_i/\partial\langle g(\mathbf{y}_i)\rangle_i$ can be obtained from (27) and (28) as

$$\frac{\partial w_i}{\partial\langle g(\mathbf{y}_i)\rangle_i} \approx C^2 - w_i^2 - \frac{w_i\langle g(\mathbf{y}_i)\rangle_i + \sigma_i^2 C^2 \int_{t_i-\varepsilon}^{t_i+\varepsilon} p\left(g(\mathbf{y}_i) \mid \overline{\mathfrak{D}}\right) dg(\mathbf{y}_i)}{\sigma_i^2 \mathcal{G}\left(\langle g(\mathbf{y}_i)\rangle_i, \sigma_i^2\right)} \approx C^2 - w_i^2 - \frac{w_i\langle g(\mathbf{y}_i)\rangle_i + \sigma_i^2 C^2 + \mathcal{IG}_i}{\sigma_i^2 \mathcal{G}\left(\langle g(\mathbf{y}_i)\rangle_i, \sigma_i^2\right)} \tag{32}$$

where

$$\mathcal{IG}_i = \frac{1}{2}\mathrm{erf}\left\{\frac{t_i - \langle g(\mathbf{y}_i)\rangle_i + \varepsilon}{\sqrt{2\sigma_i^2}}\right\} - \frac{1}{2}\mathrm{erf}\left\{\frac{t_i - \langle g(\mathbf{y}_i)\rangle_i - \varepsilon}{\sqrt{2\sigma_i^2}}\right\}.$$

D. Obtaining the Probability Density Function Estimate and Choosing of the Kernel Function Shape

The above discussion shows how the SVM can be used for approximating the distribution function $F(\mathbf{y})$ from the training sample $\mathcal{D}$. The approximation will be in the form of a weighted sum of the kernel function working on the instants of the training sample as [see (21)]

$$F(\mathbf{y}) = \sum_{i=1}^{n} w_i \, \mathcal{K}(\mathbf{y}, \mathbf{y}_i). \tag{33}$$

Consequently, the estimate of the density function will be simply in the form

$$p(\mathbf{y}) = \sum_{i=1}^{n} w_i \, \mathcal{K}'(\mathbf{y}, \mathbf{y}_i) = \sum_{i=1}^{n} w_i \, \mathfrak{K}(\mathbf{y}, \mathbf{y}_i) \tag{34}$$

where $\mathfrak{K}(\mathbf{y}, \mathbf{y}_i)$ is the derivative of $\mathcal{K}(\mathbf{y}, \mathbf{y}_i)$.

There are some conditions on the kernel function $\mathfrak{K}(\mathbf{y}, \mathbf{y}_i)$ so that a valid density function estimate can be obtained from (34), see [13]. These conditions are as follows:

1) $\mathfrak{K}_\gamma = a(\gamma)\,\mathfrak{K}((\mathbf{y} - \mathbf{y}_i)/\gamma)$;
2) $a(\gamma)\int \mathfrak{K}((\mathbf{y} - \mathbf{y}_i)/\gamma)\, d\mathbf{y} = 1$;
3) $\mathfrak{K}(0) = 1$.

In this paper, a Gaussian radial basis function (GRBF) kernel is used which satisfies the above conditions (see, [13]) and it has the form

$$\mathfrak{K}(\mathbf{y}, \mathbf{y}_i) = \exp\left(-\frac{1}{2}(\mathbf{y} - \mathbf{y}_i)\Lambda^{-1}(\mathbf{y} - \mathbf{y}_i)^T\right) \tag{35}$$

where Λ is a parameter which is assumed to be predefined in this paper.

E. Summary of the Proposed SVM Density Estimation Algorithm

The implementation steps of the proposed approach for density estimation using an SVM with the mean field theory being applied to the learning process are presented below.

Algorithm 1

1) *Generate the training dataset* $\mathfrak{D}$ *defined in* **(11)**.
2) *Set a learning rate* η *and randomly initialize* w_i*'s.*
3) *Choose a kernel* $\mathfrak{K}(\mathbf{y}, \mathbf{y}')$ *and accordingly, calculate the covariance matrix* $\mathcal{K}_n$ *and let* $\sigma_i^2 = [\mathcal{K}_n]_{ii}$.
4) *Iterate steps 5) and 6) until convergence in* w_i*'s.*
5) *"inner loop": For* $i = 1, 2, \ldots, n$ *do*
   5.1 *calculate* $\langle g(\mathbf{y}_i)\rangle$ *from* **(29)**
   5.2 *calculate* $\langle g(\mathbf{y}_i)\rangle_i$ *from* **(30)**.
   5.3 *calculate* $\mathcal{F}_i$ *and* $\mathcal{G}_i$ *from* **(28)**
   5.4 *update* w_i *by:*

$$w_i = w_i + \eta\left(\frac{\mathcal{F}_i}{\mathcal{G}_i} - w_i\right).$$

6) *"outer loop": For every* M *iterations for* w_i*, update* σ_i^2 *from* **(31)**.
7) *Calculate* $p(\mathbf{y})$ *from* **(34)**.

The most computationally expensive step in the above algorithm is the inversion of the matrix $\mathcal{K}_n + \Sigma$ in step 6). So, it is recommended that step 6) at the "outer loop" iterate less frequently than step 5) of the "inner loop." For example, after $M = 10$ iterations of updating w_i, there will be one update of σ_i^2.

IV. Class Prior Probability Estimation

The MAP principle requires the estimation of the Class Prior Probability (CPP) for each class defined in the image. The CPP represents the "initial belief" that a class may present at a specific pixel. Markov random field (MRF) modeling for image regions (classes) is considered as an intuitive choice for the CPP estimation since it models the relative interactions of the sites which are neighbors to the one under consideration. The parameters of the MRF (MRF parameters) specifies the strengths of the interactions of the neighboring sites. Rather than using predefined (empirical) values for Gibb's parameters, which is the case in previous work [28], [29], the MRF parameters are automatically identified in the proposed framework. In the following, an overview of MRF image modeling is presented and then the proposed algorithm for the identification of the MRF parameters using an analytical method is discussed.

A. MRF Model

According to the Boltzmann equation, the distribution of the probability of a molecule to be in a state with energy ε is

$$p(\varepsilon) = \frac{1}{Z} e^{-(1/K_B T)\varepsilon} \tag{36}$$

where Z is a normalization constant which makes the sum of probabilities equal to 1, T is the absolute temperature and K_B is the Boltzmann's constant. For simplicity, the temperature is assumed to be measured in energy units such that $K_B T$ is replaced by T.

Gibbs used a similar distribution in 1901 to express the probability of a whole system with many degrees of freedom to be in a state with a certain energy. A Gibbs random field (GRF) provides a global model for an image by specifying a probability mass function in the form

$$p(x) = \frac{1}{Z} e^{-E(x)/T} \tag{37}$$

where the normalization constant $Z = \sum_{x \in \Omega} e^{-E(x)/T}$, Ω is the set of all possible configurations on the given grid, and the function $E(x)$ is called the energy function. A GRF describes the global properties of an image in terms of the joint distributions of colors for all pixels while a Markov random field (MRF) is defined in terms of local properties. Thus, an MRF is a GRF [has the same representation in (37)], but it is defined in terms of local properties only. Before showing the basic properties of MRF, some terms related to MRF will be briefly defined [30]–[32].

Definition 1: A clique is a subset of the lattice support of the image for which every pair of sites are neighbors. Single pixels are also considered cliques.

Definition 2: A random field $\mathbf{X}$ is a Markov random field with respect to the neighborhood system $\zeta = \{\zeta_s, s \in \Omega\}$, where $s = (i, j)$ in the image representation, if and only if

- $p(\mathbf{X} = \mathbf{x}) > 0 \ \forall \ x \in \Omega$,
- $p\left(\mathbf{X}_s = \mathbf{x}_s \mid \mathbf{X}_{s|r} = \mathbf{x}_{s|r}\right) = p(\mathbf{X}_s = \mathbf{x}_s \mid \mathbf{X}_{\zeta_s} = \mathbf{x}_{\zeta_s})$ where $s \mid r$ refers to all the sites $s \in \Omega$ excluding the neighbor sites r, and ζ_s refers to the neighborhood of the site s.

The structure of the neighborhood system determines the order of the MRF model. For a first-order MRF model, the neighborhood of a pixel consists of its four nearest neighbors. In a second-order MRF the neighborhood consists of the eight nearest neighbors. The clique structures are shown in Fig. 1.

Consider a graph (t, ζ) as shown in Fig. 2 which has a set of N^2 sites. The energy function for a pairwise interaction model can be written in the form

$$E(x) = \sum_{t=1}^{N^2} F(x_t) + \sum_{t=1}^{N^2} \sum_{r=1}^{w} H(x_t, x_{t:+r}) \tag{38}$$

where $F(.)$ is the potential function for single-pixel cliques and $H(.,.)$ is the potential function for all cliques of size 2. The parameter w depends on the neighborhood around each site. For example, w is 2, 4, 6, 10, and 12 for neighborhoods of orders 1, 2, 3, 4, and 5, respectively. Using Derin–Elliott model [32] to compute $F(x)$ and $H(x_t, x_{t:+r})$, then

$$F(x_t) = \gamma_0 \quad H(x_t, x_{t:+r}) = \gamma_r I(x_t, x_{t:+r}) \tag{39}$$

where $I(s_t, s_{t:+r})$ is the indicator function which is defined as

$$I(a, b) = +1 \text{ if } a = b \quad I(a, b) = -1 \text{ if } a \neq b.$$

As mentioned in (3), one of the requirements of the MAP principle is to maximize the estimate of the CPP probability. From the above outlines of the MRF image modeling, one can see that the maximization of the CPP corresponds to the maximization of (37) which in turn corresponds to a proper choice of the parameters of the MRF model. The following section discusses an algorithm which analytically identifies the parameters of the Gibbs–Markov random field (GMRF) model.

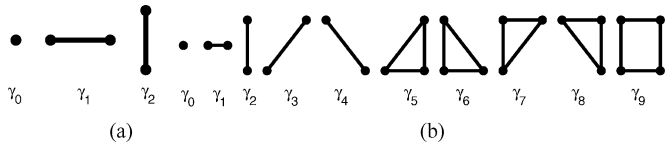


Fig. 1. Cliques for (a) first-order and (b) second-order neighborhood, where γ's are the cliques' coefficients.

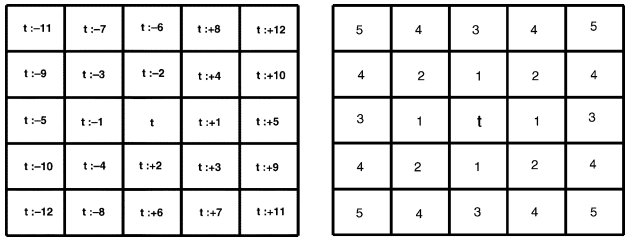


Fig. 2. (a) Numbering. (b) Order coding of the neighborhood structure.

B. Parameters Estimation of the GMRF Using an Analytical Method

In this paper, the GMRF model is simplified by restricting it to the second-order structure which means that the neighborhood pixels are restricted to eight-neighbors. Also, symmetric label interactions are assumed in this paper which means that the Gibbs parameters are independent of relative orientation of pixel pairs, the same for all classes, and depend only on whether the pair of labels are equal or not. Under these simplifications the model parameters γ's are equal (i.e., $\gamma_0 = \gamma_1 = \cdots = \gamma_4$) and the model contains only one parameter that will be indicated by γ hereafter. The model is closely similar to the conventional autobinomial one [23] but differs in that the parameters have no predefined functional form.

Let the set

$$\mathbf{V} = \{V(k, c) = \gamma \text{ if } k = c \quad V(k, c) = -\gamma \text{ if } k \neq c; \ k, c \in \mathbf{K}\}$$

denotes the centered bivalued Gibbs potential governing symmetric pairwise cooccurrences of the region labels and the set $\mathbf{N} = \{(1, 0), (0, 1), (-1, 1), (1, 1)\}$ specifies the interpixel offsets for the eight-neighboring pixel pairs. Then the MRF model of a region is specified by the following Gibbs probability distribution (GPD):

$$\begin{aligned} P(\mathbf{X}) &= \frac{1}{\mathbf{Z_N}} \exp\left(\sum_{(i,j) \in \mathbf{S}} \sum_{(\xi,\eta) \in \mathbf{N}} V(\mathbf{X}_{i,j}, \mathbf{X}_{i+\xi, j+\eta}) \right) \\ &= \frac{1}{\mathbf{Z_N}} \exp\left(\gamma \left| \mathbf{T_N} \right| (2 f_{\text{eq}}(\mathbf{X}) - 1)\right) \end{aligned} \tag{40}$$

where $\mathbf{Z_N}$ is the normalizing factor (partition function) and

$$\mathbf{T_N} = \Big\{ ((i, j), (i + \xi, j + \eta)) : (i, j) \in \mathbf{S}; \ (i + \xi, j + \eta) \in \mathbf{S}; (\xi, \eta) \in \mathbf{N} \Big\}$$

is the family of the neighboring pixel pairs supporting the Gibbs parameters, $|\mathbf{T_N}|$ is the cardinality of that family, and $f_{\rm eq}(\mathbf{X})$ denotes the relative frequency of the equal labels in the pixel pairs of that family

$$f_{\rm eq}(\mathbf{X}) = \frac{1}{|\mathbf{T_N}|} \sum_{((i,j),(i+\xi,j+\eta))\in \mathbf{T_N}} \delta(X_{i,j} - X_{i+\xi,j+\eta}) \quad (41)$$

where $\delta(.)$ is the Kronecker delta function: $\delta(0) = 1$ and otherwise 0. To identify this model, it is sufficient to estimate only the potential value γ.

To compute the second term in (3), $(1/|\mathbf{S}|)\log P(\mathbf{X})$, the approximate partition function $\mathbf{Z_N}$ (which has been proposed in [18]) is used. It is reduced in the current case as follows:

$$\begin{aligned}\mathbf{Z_N} &\approx \exp\left(\sum_{i,j\in\mathbf{S}}\sum_{\xi,\eta\in\mathbf{N}}\sum_{k\in\mathbf{K}} V\left(\mathbf{k}, \mathbf{X}_{i+\xi,j+\eta}\right)\right)\\ &= \exp\left(|\mathbf{T_N}| \sum_{k\in\mathbf{K}} \big(\gamma f_k(\mathbf{X}) - \gamma(1 - f_k(\mathbf{X}))\big)\right)\\ &= \exp\big(\gamma\, |\mathbf{T_N}|(2-K)\big)\end{aligned}$$

where $f_k(\mathbf{X})$ is the marginal frequency of the label k in the region map $\mathbf{X}$ so that

$$\begin{aligned}\frac{1}{|\mathbf{S}|}\log\left(P(\mathbf{X}=\mathbf{x})\right) &= \frac{|\mathbf{T_N}|}{|\mathbf{S}|}\gamma\big(2f_{\rm eq}(\mathbf{X}) + K - 3\big)\\ &\approx |\mathbf{N}|\gamma\big(2f_{\rm eq}(\mathbf{X}) + K - 3\big)\\ &= 4\gamma\big(2f_{\rm eq}(\mathbf{X}) + K - 3\big). \quad (42)\end{aligned}$$

Let $\rho = |\mathbf{T_N}|/|\mathbf{S}|$ denote the scaling factor which has the approximate value: $\rho \approx 4$. The first approximation of the MLE of the Gibbs parameter [21] is obtained by truncating the Taylor series expansion of the unconditional log-likelihood $\Gamma(\mathbf{X} \mid \gamma) = (1/|\mathbf{S}|)\ \log(P(\mathbf{X}=\mathbf{x}))$

$$\begin{aligned}\Gamma(\mathbf{X} \mid \gamma) &= \gamma\,\rho\,\big(2f_{\rm eq}(\mathbf{X}) - 1\big)\\ &\quad - \frac{1}{|\mathbf{S}|}\ \log\left(\sum_{\mathbf{x}\in\mathcal{X}} \exp\big(\gamma\,\rho\,\big(2f_{\rm eq}(\mathbf{X}) - 1\big)\big)\right) \quad (43)\end{aligned}$$

to the first three terms in the close vicinity of the zero potential, $\gamma = 0$

$$\begin{aligned}\Gamma(\mathbf{X} \mid \gamma) = \Gamma(\mathbf{X} \mid 0) &+ \gamma \frac{d\Gamma(\mathbf{X} \mid \gamma)}{d\gamma}\bigg|_{\gamma=0}\\ &+ \frac{1}{2}\gamma^2\ \frac{d^2\Gamma(\mathbf{X} \mid \gamma)}{d\gamma^2}\bigg|_{\gamma=0}. \quad (44)\end{aligned}$$

Zero potential produces an independent random field (IRF) of equiprobable region labels $k \in \mathbf{K}$. In this case the relative frequency of the equal pairs of labels over $\mathbf{T_N}$ has the expected value $1/K$ and the variance $(k-1)/k^2$. Then the following relationships hold:

$$\begin{aligned}\frac{d\Gamma(\mathbf{X} \mid \gamma)}{d\gamma}\bigg|_{\gamma=0} &= 2\rho\left(f_{\rm eq}(\mathbf{X}) - \frac{1}{K}\right);\\ \frac{d^2\Gamma(\mathbf{X} \mid \gamma)}{d\gamma^2}\bigg|_{\gamma=0} &= -4\rho\frac{k-1}{k^2}\end{aligned}$$

where $f_{\rm eq}(\mathbf{X})$ is the relative frequency of the equal label pairs in the region map $\mathbf{X}$ specified in (41). The likelihood in (44) results in the following approximate MLE of γ for a given map $\mathbf{X}$:

$$\gamma = \frac{K^2}{2(K-1)}\left(f_{\rm eq}(\mathbf{X}) - \frac{1}{K}\right). \quad (45)$$

The relationship allows for computing the parameters of the MRF model for each current region map obtained by the Bayesian classification based on the estimated CCP model.

V. Segmentation Algorithm

To make the search for a local maximum of the log-likelihood of (3) computationally feasible, a conventional iterative process of estimating/reestimating the conditional image model is used (i.e., given a current region map, then update the map model given the image). The process terminates when the current and previously estimated model parameters coincide to within a given accuracy range [19], [20], [22]. In our case, most of these computations are analytical.

Therefore, the whole iterative segmentation process is summarized as follows.

Algorithm 2

- *Initialization:* Find an initial map by the classical pixelwise Bayesian classification of a given image after an initial estimation of the CCP, $\mathbf{Y}$, using MF-based SVM density estimator.
- *Iterative refinement:* Refine the initial map by iterating the following two steps:

1) Estimate the potential value γ for each region map model using **(45)**.
2) Refine the segmented image using the ICM algorithm **[18]**.
3) Calculate the log-likelihood from **(3)** and terminate if the change in the values of the log-likelihood is less than ϵ.

Because at each step the approximate log-likelihood is greater than or equal to its previous value, the proposed algorithm converges to a locally optimum solution. The experimental section presents some experimental evolution of the log-likelihood values in (3) with the iterations of the proposed segmentation algorithm.

VI. Experimental Results

This section presents an extensive elaborated experimental work which has been carried out to evaluate the proposed framework. Each part of the proposed framework is evaluated separately to assess its performance, then the performance of the framework is evaluated as a whole. The experiments are carried out using synthetic datasets in various dimensional spaces, and real remote sensing multispectral and hyperspectral datasets. In the synthetic data experiments, the Kullback–Leibler distance (KLD) [33] is used to measure the density estimation accuracy

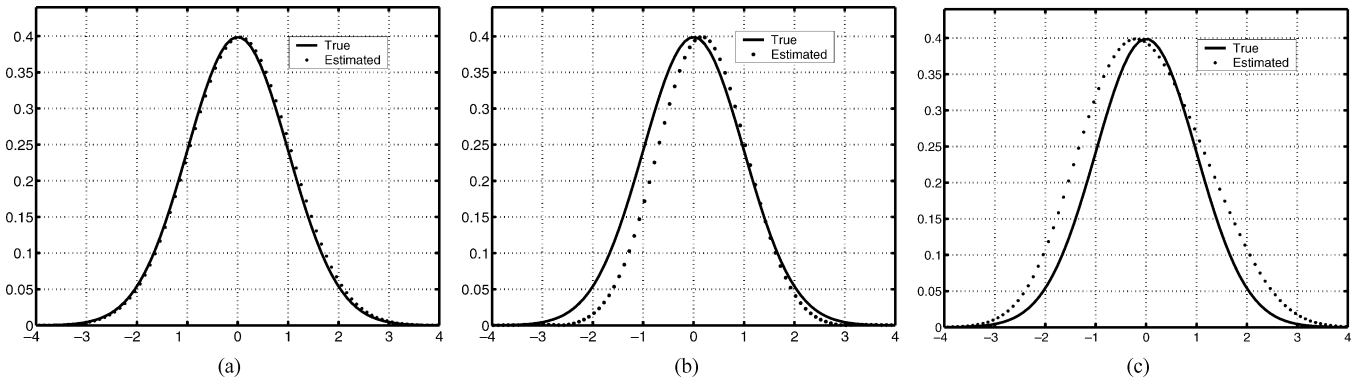


Fig. 3. Estimation of the 1-D Gaussian density function with the SVM density estimation which is formulated with (a) the proposed and (b) traditional formulation. (c) Effect of the regularization constant C on the proposed algorithm performance.

TABLE I
PARAMETERS OF THE 1-D MIXTURE OF GAUSSIANS DENSITY FUNCTION

Parameter	μ_1	μ_2	σ_1^2	σ_2^2	α_1	α_2
Value	-1	7	9	4	0.4	0.6

of a density estimation algorithm. The density estimation accuracy in real remote sensing data experiments is reflected in the segmentation accuracy of the datasets using only class conditional probability modeling with prespecified priors. Refinement of the segmented images using modeling of the classes priors reflects the effectiveness of the Markov random field in modeling CPP.

The experiments in this paper are programmed using Matlab 6.1 with standard toolboxes. The web site: http://www.cvip.uofl.edu/Publications/TechnicalReports contains tutorials on the details of the paper contents. It contains also samples of the Matlab code used in the experiments.

A. Experiments for Density Estimation Using Synthetic Data

In this part of the experiments, several synthetic data examples are used to illustrate the performance of the proposed algorithm for density estimation. The datasets are generated with standard random generators in 1-D and two-dimensional (2-D) spaces. The performance of the proposed algorithm is evaluated visually and using the Kullback–Leibler distance measure which is perhaps the most frequently used information-theoretic distance measure between two probability densities.

1) Density Estimation for a 1-D Gaussian Distribution: This is a simple and standard experiment but illustrative and it is used here for comparison purposes. In this experiment, a data sample of 100 instants from a 1-D standard normal distribution is used to illustrate the performance of the proposed MF-based SVM density estimation algorithm. The results are compared to those obtained in a previous work which had been done using the traditional formulation of the SVM-based density estimation approach. The comparison is done based on the visual evaluation, the convergence speed, and the KLD measure.

As shown in Fig. 3, the MF-based SVM approximates closely the reference density function, while there is an apparent error in the approximation produced by the traditionally formulated SVM. In the case of the proposed MF-based SVM estimator, the KLD are $\mathrm{KLD}(p_1 \mid p_0) = 0.12$ and $\mathrm{KLD}(p_0 \mid p_1) = -0.094$. The two KLDs are close enough to each other to show that the estimation is a good one. On the other hand, in the traditional formulation-based SVM approximation Fig. 3(b), the distances are $\mathrm{KLD}(p_1 \mid p_0) = -0.85$ and $\mathrm{KLD}(p_0 \mid p_1) = 1.86$ which are not close to each other like the case in the proposed algorithm. The computational cost of the proposed algorithm is of order $\mathcal{O}(N^2)$ while the traditional SVM learning algorithm has the order $\mathcal{O}(N^3)$; see [34]. The current experiment takes 0.015 s for the optimization process in the proposed MF-based SVM, while it takes 0.22 s with the traditional SVM which emphasizes the faster response of the proposed algorithm. One thing to be mentioned here is that the proposed algorithm depends on the choice of the parameters (Step 1 of the algorithm in Section III-E). Therefore, a careful selection of the optimal values should be done. Fig. 3(c) shows that the improper choice of the regularization constant C ($C = 2.1$ in this case while it was 0.1 in the previous case) results in a bad performance of the algorithm. Currently, these parameters are chosen empirically and the experiments show that the algorithm is not sensitive to them within a range.

2) Density Estimation for 1-D Mixture of Gaussians Distribution: In this little more challenging experiment, a dataset of 100 instants is generated from a 1-D mixture of Gaussians. The mixture consists of two components and has the form

$$p(\mathbf{x}) = \alpha_1 \mathcal{N}(\mu_1, \sigma_1^2) + \alpha_2 \mathcal{N}(\mu_2, \sigma_2^2) \tag{46}$$

with the parameters shown in Table I.

The results in Fig. 4 show that the proposed algorithm approximates well the density function in (46). There are little errors at the tails of the density function components. These tail-errors add up at the intersection of the two components which produces a noticeable error. The distance measure values in this case are $\mathrm{KLD}(p_1 \mid p_0) = 0.26$ and $\mathrm{KLD}(p_0 \mid p_1) = -0.09$ which are affected by the error discussed before. This experiment takes 0.313 s for the optimization process to converge which means that the algorithm still maintains a considerably fast convergence.

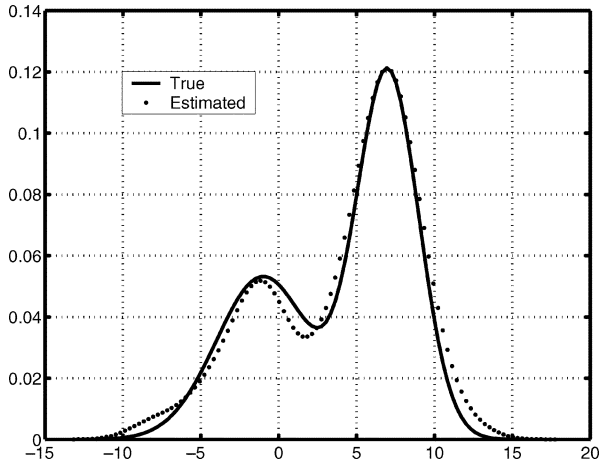


Fig. 4. Estimation of a 1-D mix of Gaussian density functions.

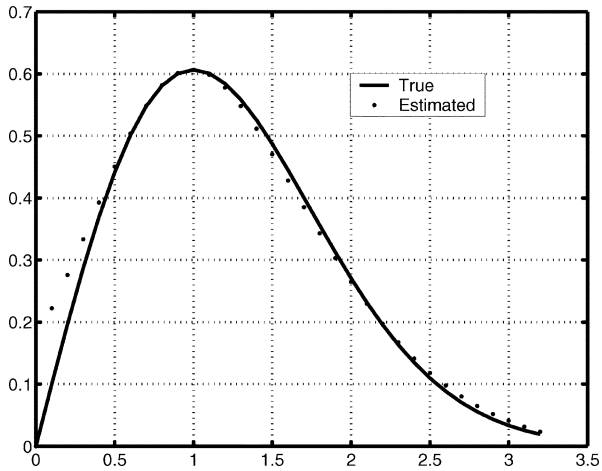


Fig. 5. Estimation of a Rayleigh density function.

3) Density Estimation for 1-D Rayleigh Distribution: In this experiment, a data sample of 100 instants from a Rayleigh distribution which has the form

$$p(x) = \frac{xe^{-x^2/2s^2}}{s^2} \tag{47}$$

where the parameter s is set to 1 in the experiment. The Rayleigh distribution is chosen because there is a special interest in the medical imaging applications for the Rayleigh distribution, and also this distribution represents a good nonsymmetric variation other than the Gaussian. The results shown in Fig. 5 illustrate that the proposed algorithm approximates the density function very well. The KLD in this case are $\mathrm{KLD}(p_1 \mid p_0) = 0.3$ and $\mathrm{KLD}(p_0 \mid p_1) = -0.2$. The apparent difference between the two distances may be, is due to the small error in the left tail. It is interesting to note here that the peak values of the densities (reference and estimated) occur at the same value of x which is a positive argument for the proposed estimation algorithm. This experiment takes 0.016 s for the optimization process to converge.

4) Density Estimation for a 2-D Gaussian Distribution: This experiment is carried out to assess the performance of the proposed algorithm in high-dimensional spaces. A dataset of 100 instants from a 2-D Gaussian distribution is used. Again, this experiment is used to compare the proposed algorithm with the traditionally formulated SVM algorithm. Fig. 6 shows both the density function and its contour for the reference density function, the estimated density function using the traditionally formulated SVM estimator and the estimated density function using the proposed MF-based SVM estimator.

As can be noted from the figure, there is a significant improvement in the estimation using the MF-based SVM over the traditionally formulated SVM. In the contour plot for the estimated density there is a slight deformation in the contour of the estimated density function using the traditional SVM and there is a shift in the mean vector. The distance measures in the case of the traditionally formulated SVM estimator are $\mathrm{KLD}(p_1 \mid p_0) = 0.39$ and $\mathrm{KLD}(p_0 \mid p_1) = -8.4$, which shows that there is a large difference due to the shift mentioned before. The significant improvement can be noted from the contour of the estimated density function using the MF-based SVM estimator. The distance measures are $\mathrm{KLD}(p_1 \mid p_0) = 4.029$ and $\mathrm{KLD}(p_0 \mid p_1) = -3.6$, showing a close fit. This experiment takes 0.172 s for the optimization process to converge with the proposed MF-based SVM learning, while it takes 0.578 s with the traditional SVM.

B. Experiments for Density Estimation Using Real Remote Sensing Multispectral Data

The following experiments are used to illustrate the performance of the proposed density estimation algorithm in real datasets of relatively high-dimensional spaces. In the current experiments, two sevens multispectral datasets, with 30-m resolution are used. In the multispectral experiments, each point in the datasets is represented by a vector of length 7, giving a dimensionality of the seventh order. Since, the reference density function is not known for these real datasets, the above evaluation methods (visual inspection and KLD measure) for the performance of the proposed density estimation algorithm can *not* be used. Instead, the classification accuracy is used as a practical measure of the performance of the density estimation algorithm. When carrying the classification experiments to compare the performance of the different density estimation algorithms, the operating conditions (in a Bayes classification setup) are the same except for the density estimation algorithm. Thus, the argument that the classification accuracy is an indication for the density estimation performance is applicable.

1) Test-of-Agreement for the Response of Two Classifiers: To compare the performance of two classifiers against each other, a rule is proposed which will be discussed here. Suppose there are two classifiers with the rules $M_1(.)$ and $M_2(.)$, respectively, which are applied to the test dataset. Define the statistic

$$S_n = \sum_{i=1}^{n} z_i \tag{48}$$

where $z_i = 1$ if $M_1(\mathbf{y}_i) = M_2(\mathbf{y}_i)$ and 0 otherwise. The statistic S_n measures the agreement between the two classifiers' outputs, reflecting how much the responses of the two classifiers

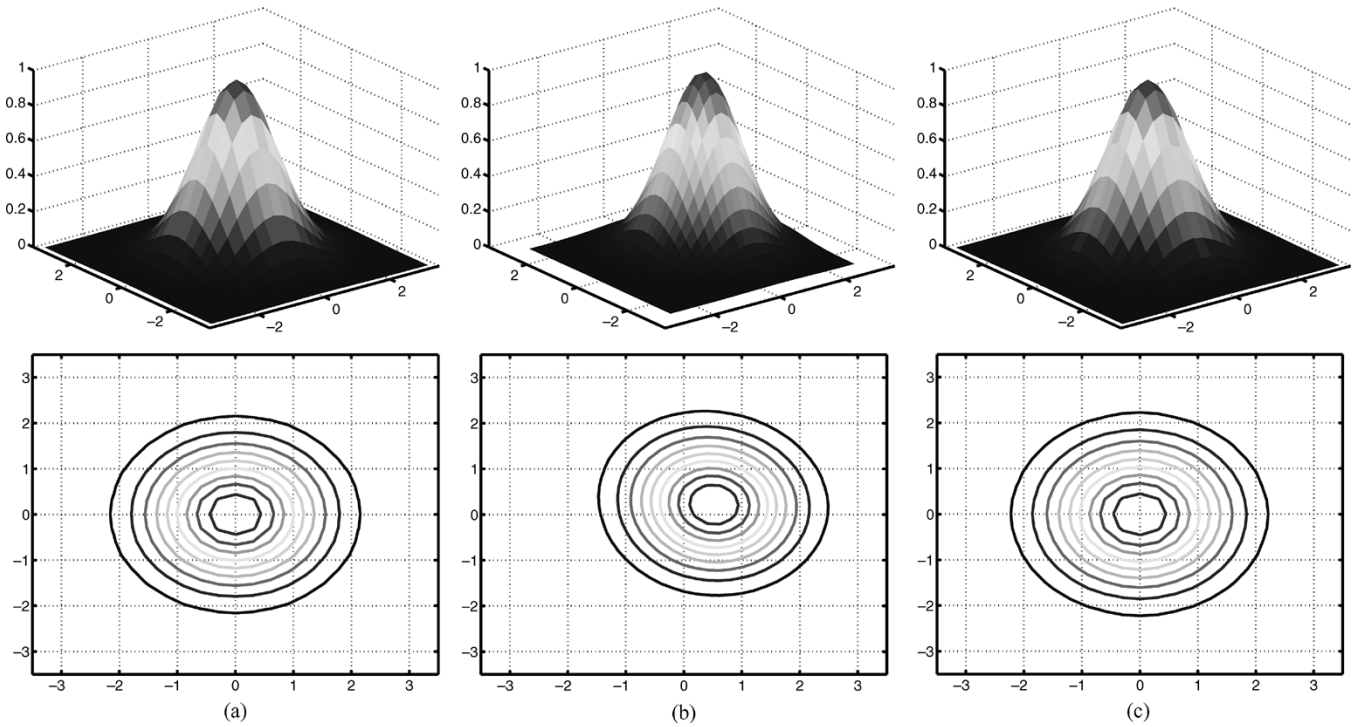


Fig. 6. Estimation of a 2-D Gaussian density function. (a) The reference density function and its contour. (b) The estimated density using the traditional formulation-based SVM and its contour. (c) The estimated density using MF-based SVM and its contour.

agree in response to the same data point. From the central limit theorem (CLT) [24]

$$\frac{S_n - \widehat{S_n}}{\sqrt{n\widehat{S_n}^2}} \sim \mathcal{N}(0, \widehat{S_n}). \tag{49}$$

The hypothesis

$$H_0 : \text{probability that the classifiers agree.} \tag{50}$$

has the 95% confidence interval $[\widehat{S_n}/n - 2A \quad (\widehat{S_n}/n) + 2A]$, where $A = \sqrt{\widehat{p}(1-\widehat{p})/n}$; $\widehat{p} = \widehat{S_n}/n$.

If this confidence interval contains the point 1, then the probability that the responses of the two classifiers agree is 1. If the confidence interval does not contain 1, then there is a chance that the two classifiers disagree, and thus the hypothesis is rejected. With this argument in mind, the rule to compare the performance of two classifiers is as follows.

1) If there is a difference in the performance between the two classifier but they disagree, this means that this difference is significant.
2) If there is a difference in the performance between the two classifier and they agree, this means that this difference is not significant.

Throughout the following experiments, the 95% confidence interval between the Bayes classifier which uses the MF-based SVM density estimator against that uses MLE, Parzen-window, KNN, or traditionally formulated SVM density estimators are calculated. If an interval contains the point 1, then the apparent difference in the performance between the two classifiers is not *significant*.

2) Experiments for Density Estimation Using a Multispectral Agricultural Area: This dataset represents an agricultural area in the state of Kentucky, in the U.S. and it is of size 169×169 pixels. Nine classes are defined in this dataset: Background, Corn, Soybean, Wheat, Oats, Alfalfa, Clover, Hay/Grass, and Unknown. The ground truth labels are available for the whole dataset. For the evaluation purposes, a subset from each class is used for training the density estimator and the rest of the data is used for testing. Fig. 7 shows the reference land cover of the area and the classification results based on the SVM density estimators.

The confusion matrix for the classification based on the density estimation using MF-based SVM is shown in Table II. The average true classification accuracy for the classes is 78.5%. The largest source of error is due to the misclassification between the Background and other classes, with 46% of the Background reference pixels are classified to the other classes and 9.6% from the other classes are classified to Background. A specific noticeable example for the misclassification between the Background and the other classes is the misclassification between the Soybean and the Background, with 18.6% from the Background reference points are classified as Soybean and 5% from the Soybean are classified as Background. Other large errors can be noted in the Alfalfa and Hay/Grass classes. However, the reason for the later error is due to the prior probability assumption which is assumed as the share of the class reference points in the dataset. Since, each of the Alfalfa and Hay/Grass classes is less represented in the dataset, their priors are small and thus a noticeable error is generated. This realization calls for another estimation method for the prior probabilities which is done in this paper using the MRF modeling for the CPP as discussed before and will be demonstrated through some experiments later. Another observation that can be shown from the classified image in Fig. 7(b), in which

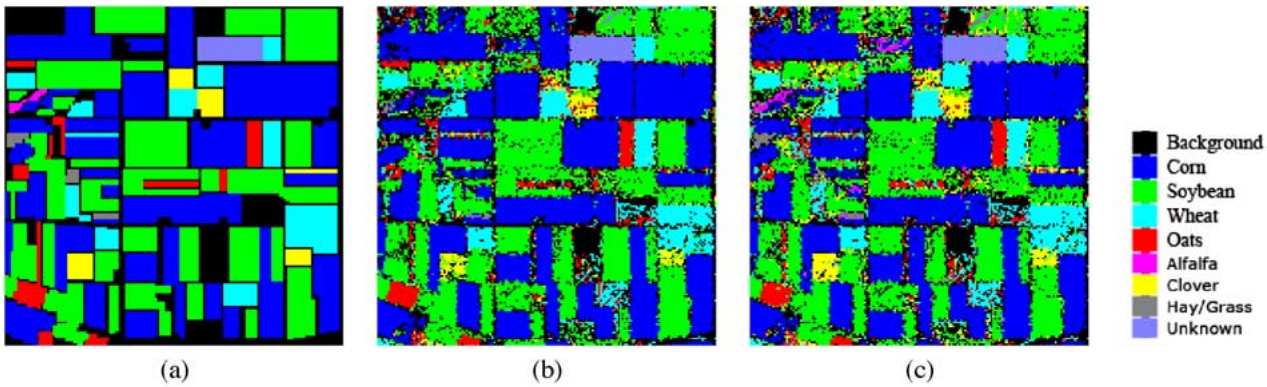


Fig. 7. Multispectral agricultural area. (a) Land cover and classification results using (b) SVM and (c) MF-SVM as a density estimator.

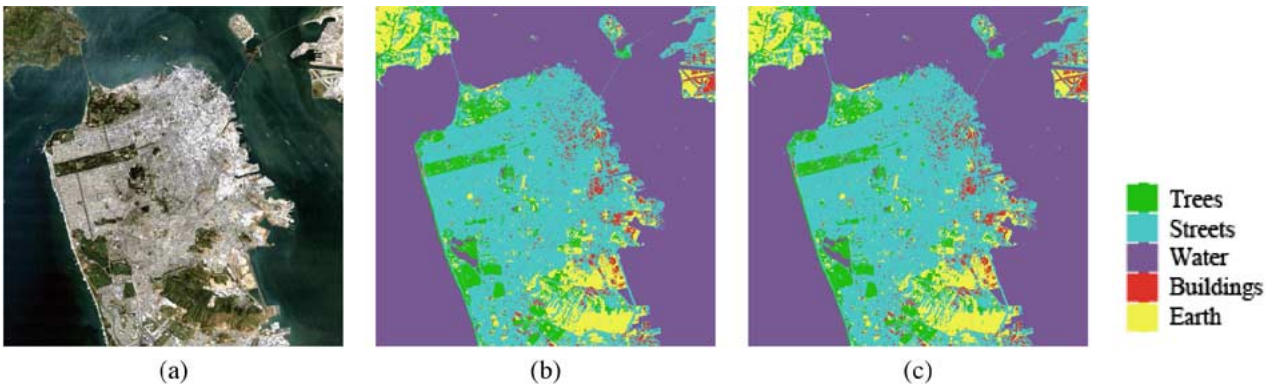


Fig. 8. Multispectral urban area. (a) RGB snap-shot and color-coded classification results using (b) SVM and (c) MF-SVM as a density estimator.

TABLE II
CLASSIFICATION CONFUSION MATRIX FOR THE MULTISPECTRAL AGRICULTURAL AREA USING THE MF-BASED SVM ESTIMATOR

Class	Total Points	Back-ground	Corn	Soy-bean	Wheat	Oats	Alfalfa	Clover	Hay/ Grass	Unknown	% True per Class
Background	6790	3661	770	1262	555	358	4	144	15	21	53.92
Corn	9371	475	8787	93	1	7	3	1	3	1	93.77
Soybean	8455	1090	101	6985	74	85	0	92	6	22	82.61
Wheat	1923	199	0	37	1581	74	0	31	1	0	82.22
Oats	800	121	2	22	47	598	0	10	0	0	74.75
Alfalfa	65	19	22	6	0	0	13	0	5	0	20
Clover	619	120	7	103	19	54	0	316	0	0	51.05
Hay/Grass	142	54	17	27	1	6	1	5	29	2	20.42
Unknown	396	8	0	3	6	0	0	0	1	378	95.45
% +ve Rate		63.7	90.53	95.51	69.22	50.59	61.9	52.75	48.33	89.15	

TABLE III
CLASSIFICATION ACCURACY USING DIFFERENT DENSITY ESTIMATORS FOR THE MULTISPECTRAL AGRICULTURAL AREA

Class	% Accuracy				
	MLE	Parzen Window	KNN (k=15)	Traditional SVM	MF-based SVM
Background	52	37	46	50.4	53.9
Corn	94	97	96	91.5	93.77
Soybean	78	92	82	77.9	82.61
Wheat	44	31	40	84.2	82.22
Oats	7	9	4	72.5	74.75
Alfalfa	0	0	0	76.9	20
Clover	5	4	4	69.8	51.05
Hay/Grass	0	0	1	66.2	20.42
Unknown	94	94	94	95.7	95.45
Average	71	72	71	76.1	78.5

most of the regions contain some random misclassification points (which appear like a random salt-and-pepper noise in the image) although they should be smooth and clean. This is mainly due to the fact that the Bayes classification setup treats the points in the dataset as realizations of independent random variables regardless of the contextual interactions. The MRF modeling overcomes also this problem as will be illustrated later.

TABLE IV
CLASSIFICATION CONFUSION MATRIX FOR THE MULTISPECTRAL URBAN AREA USING THE MF-BASED SVM ESTIMATOR

Class	Total Points	Trees	Streets	Water	Buildings	Earth	% True per Class
Trees	212	212	0	0	0	0	100
Streets	521	2	495	0	4	20	95
Water	595	0	0	595	0	0	100
Buildings	292	1	37	0	254	0	87
Earth	410	0	4	0	0	406	99
% +ve Rate		98.6	92.35	100	98.45	95.31	

To evaluate the proposed MF-based SVM density estimator, other classical and new density estimation algorithms are

TABLE VI
CLASSIFICATION CONFUSION MATRIX FOR THE HYPERSPECTRAL 34-BAND DATA USING THE MF-BASED SVM ESTIMATOR

Class	Total Points	Agricultural	Coniferous	Herbaceous	Other Impervious	Roads	Soil	Water	% True per Class
Agricultural	5138	4122	170	497	221	11	102	15	80.23
Coniferous	15182	3	12878	2228	66	2	1	4	84.82
Herbaceous	7481	27	947	5914	291	197	75	30	79.05
Other Impervious	925	4	56	175	595	62	30	3	64.32
Roads	627	0	17	145	44	418	3	0	66.67
Soil	6362	229	4	409	609	124	4922	64	77.37
Water	4285	5	323	282	328	2	133	3212	74.96
% +ve Rate		93.89	89.46	61.28	27.6	51.221	93.46	96.5	

TABLE V
CLASSIFICATION ACCURACY USING DIFFERENT DENSITY ESTIMATORS FOR THE MULTISPECTRAL URBAN AREA

Class	% Accuracy				
	MLE	Parzen Window	KNN (k=3)	Traditional SVM	MF-based SVM
Trees	99	85	89	97.5	100
Streets	91	97	94	95.6	95
Water	97	100	100	100	100
Buildings	90	68	89	91.8	87
Earth	80	82	84	91.3	99
Average	92	89	92.7	95.7	96.7

TABLE VII
CLASSIFICATION ACCURACY USING DIFFERENT DENSITY ESTIMATORS FOR THE HYPERSPECTRAL 34-BAND DATA

Class	% Accuracy				
	MLE	Parzen Window	KNN (k=3)	Traditional SVM	MF-based SVM
Agricultural	86.65	85.95	85.66	80.87	80.23
Coniferous	71.63	91.4	62.4	88.33	84.82
Herbaceous	77.8	44.53	70.85	67.73	79.05
Other Impervious	66.7	46.38	60.43	69.73	64.32
Roads	83.41	36.36	76.24	67.3	66.67
Soil	90.77	80.27	82.27	70.7	77.37
Water	80.98	68.59	72.7	75.08	74.96
Average	78.8	75.8	71.4	78.53	80.1

applied on the same dataset. Table III summarizes the classification accuracies obtained with different density estimators. It can be noted from the table that the MF-based SVM density estimation algorithm outperforms the other algorithms (this is reflected in the classification accuracy as discussed before). With the classical algorithms (MLE with Gaussian assumptions, Parzen-Window estimation and K-Nearest Neighbors "KNN") there are some classes which have been completely disappeared (e.g., Alfalfa and Hay/Grass); however, both the SVM-based algorithms manage to recover part of these classes. The proposed MF-based SVM outperforms the traditional SVM in the overall classification accuracy.

To justify the latest stated argument regarding the performance of the classifier, which uses the MF-based SVM density estimation, the test-of-agreement rule stated above is used to analyze the results in Table III. The 95% confidence intervals (see Section VI-B1) are [0.7462 0.7567], [0.7906 0.8005], [0.8033 0.8130], and [0.883 0.891]. None of these intervals contains the point 1 which indicates that the apparent difference in the performance of the classifier which uses the MF-based SVM density estimator and the others is *significant*. But the Bayes classifier based on the MF-based SVM density estimator has an apparently better performance than the others, reflecting the better performance of the density estimator.

3) Experiments for Density Estimation Using a Multispectral Urban Area: This dataset represents an urban area around the Golden Gate Bay at the city of San Francisco, CA and shown in Fig. 8. It is of size 700×700 pixels, and there are five classes which are defined in this dataset: Trees, Streets, Water, Buildings, and Earth. The available ground truth set contains 5076 data points. For the evaluation purposes, a subset from each class is used for training and the rest of the data are used for testing.

The confusion matrix for the classification using the MF-based SVM density estimator is shown in Table IV, which indicates that this experiment is an easy experiment with respect to the previous one. The overall average true classification accuracy for the classes is 96.7%. There is a little classification confusion between the Streets and the Earth classes where 3.8% from the Streets' points are classified as Earth and 1% from the Earth points are classified as Streets. Another noticeable cofusion is between the Buildings and Streets classes. There are 12.67% from the Buildings' points are classified as Streets while there are 0.8% from the Streets' points which are classified as Buildings. The misclassification between Streets, Earth, and Buildings classes is reasonable due to the similarity between these classes in the real world.

The proposed MF-based SVM density estimator is evaluated against some other density estimation algorithms by noting the classification rate of the Bayes classification setup using different density estimation algorithms. Table V summarizes the obtained results with different density estimators. It can be noted from the table that the SVM density estimators outperform the classical algorithms however there is a little improvement using the MF-based SVM estimator over the traditionally formulated SVM estimator.

The 95% confidence intervals for the results in Table V are [0.93 0.94], [0.93 0.94], [0.95 0.96], and [0.997 0.999]. None of these intervals contains the point 1, which emphasizes the better performance of the proposed density estimator.

C. Experiments for Density Estimation Using Real Remote Sensing Hyperspectral Data

The performance of the proposed density estimator algorithm in real high-dimensional spaces is illustrated in this section. In the current experiments, two hyperspectral datasets, one has 34 bands and the other has 58 bands are used. These hyperspectral datasets will raise a density estimation problem of the 34th and 58th dimensionally orders, respectively.

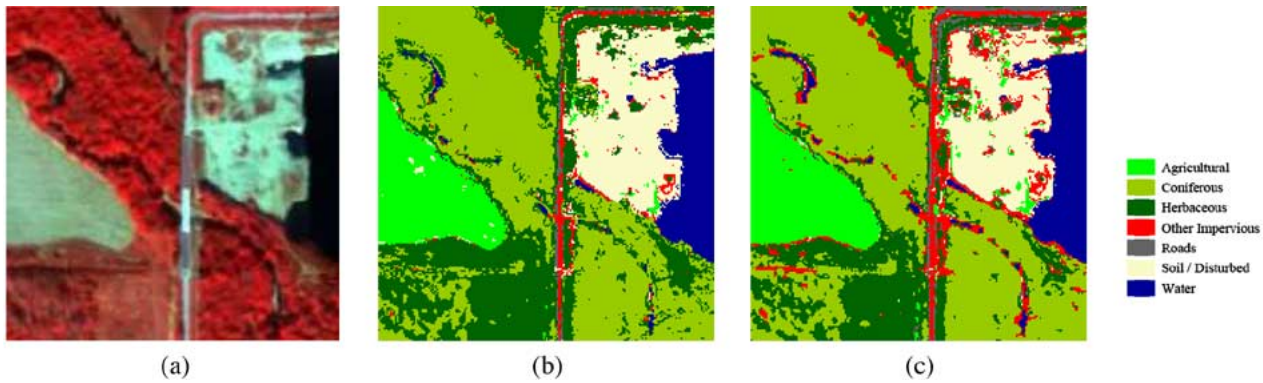


Fig. 9. Hyperspectral 34-band urban area. (a) RGB snap-shot and color-coded classification results using (b) SVM and (c) MF-SVM as a density estimator.

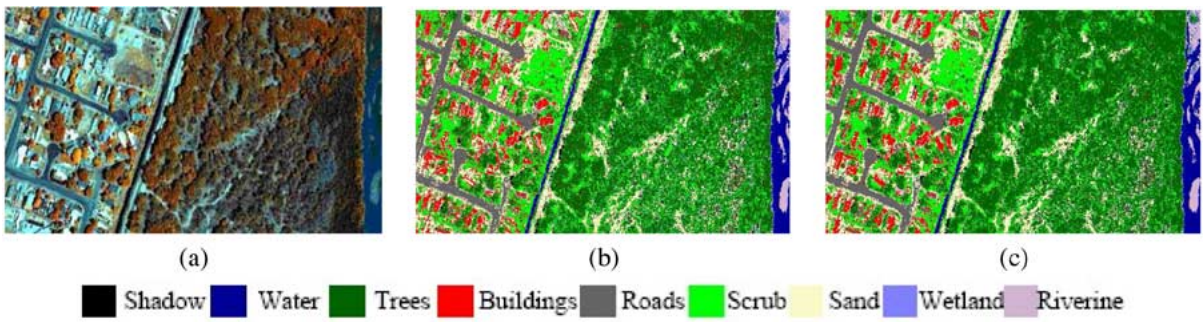


Fig. 10. Hyperspectral 58-band urban area. (a) RGB snap-shot and color-coded classification results using (b) SVM and (c) MF-SVM as a density estimator.

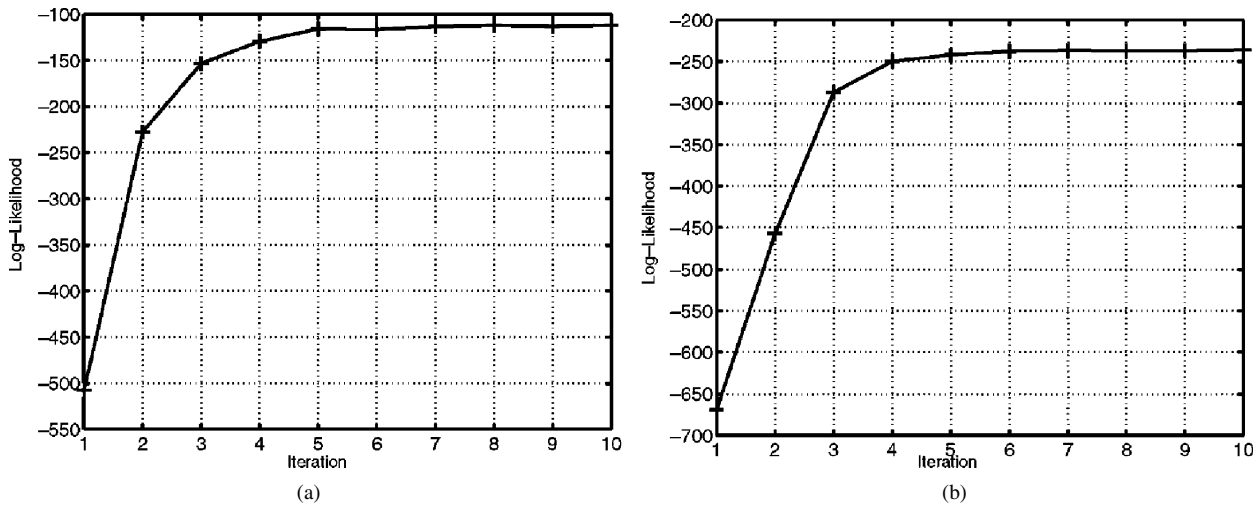


Fig. 11. Evolution of the log-likelihood in the multispectral. (a) Agricultural area example and (b) urban area example.

1) Experiments for Density Estimation Using a Hyperspectral 34-Band Dataset: This experiment uses a hyperspectral dataset of size 200×200 for an urban area in the state of Indiana, in the U.S. This dataset has a 3-m resolution, and there are seven classes defined in it: Agricultural, Coniferous, Herbaceous, Other Impervious, Roads, Soil/Disturbed, and Water. The ground truth labels are available for the whole dataset and only a subset from each class is used for training the density estimators. This dataset is illustrated in Fig. 9.

The confusion matrix for the classification based on the density estimation using MF-based SVM is shown in Table VI. The average true classification accuracy for the classes is 80.1%. The largest source of error is due to the misclassification between the different types of vegetation. Actually, the "Other Impervious" class has the lowest classification rate because it shares its characteristics with other vegetation classes.

Table VII summarizes the classification accuracies obtained with different density estimators. From that table it can be noted that the MF-based SVM density estimation algorithm outperforms the other algorithms.

The 95% confidence intervals for the results in Table VII are [0.76 0.77], [0.73 0.74], [0.74 0.75], and [0.856 0.863]. None of these intervals contains the point 1, which reflects the better performance of the MF-based SVM density estimator in high-dimensional spaces.

2) Experiments for Density Estimation Using a Hyperspectral 58-Band Dataset: Fig. 10 shows a hyperspectral dataset of size 300×600 for an urban area in the state of New

TABLE VIII
CLASSIFICATION CONFUSION MATRIX FOR THE HYPERSPECTRAL 58-BAND URBAN AREA USING THE MF-BASED SVM ESTIMATOR

Class	Total Points	Shadow	Water	Trees	Build-ings	Asph-alt	Scrub	Sand	Wetland	Substrate	% True per Class
Shadow	1189	657	0	355	8	0	160	9	0	0	55.26
Water	7382	6	6930	73	6	13	2	194	0	158	93.88
Trees	91883	190	86	74607	1125	436	10477	4670	30	262	81.20
Buildings	11655	30	17	1133	5615	905	1236	2573	0	146	48.18
Asphalt	9349	32	2	245	91	7369	142	1465	0	3	78.82
Scrub	30542	1	0	12458	725	98	16569	684	1	6	54.25
Sand	25741	9	71	2599	2431	2156	413	17965	1	96	69.79
Wetland	437	0	0	51	0	0	27	7	350	2	80.1
Substrate	1822	0	84	72	4	3	0	106	1	1552	85.18
% +ve Rate		71	96.38	82.12	56.12	67.11	57.03	64.89	51.47	69.56	

TABLE IX
CLASSIFICATION ACCURACY USING DIFFERENT DENSITY ESTIMATORS FOR THE HYPERSPECTRAL 58-BAND URBAN AREA

Class	% Accuracy				
	MLE	Parzen Window	KNN (k=3)	Traditional SVM	MF-based SVM
Shadow	87	69.13	97.9	80.82	55.26
Water	96	91.5	90.4	91	93.88
Trees	69.6	86.12	63.8	65.47	81.2
Buildings	51.3	39.76	46.7	48	48.18
Asphalt	52.9	70.8	84.9	81.42	78.82
Scrub	61	34.88	69.6	61.52	54.25
Sand	60.6	65.34	68.55	65.51	69.79
Wetland	82.38	2.29	90.85	89.47	80.1
Substrate	71.1	52.14	80.57	93.8	85.18
Average	66.2	70.2	67.02	65.99	73

TABLE X
CLASSIFICATION CONFUSION MATRIX FOR THE MULTISPECTRAL AGRICULTURAL AREA AFTER APPLYING THE MRF MODELING

Class	Total Points	Back-ground	Corn	Soy-bean	Wheat	Oats	Alfalfa	Clover	Hay/ Grass	Unknown	% True per Class
Background	6790	4050	877	1304	374	129	0	24	1	31	59.65
Corn	9371	305	8997	60	0	7	0	0	1	1	96
Soybean	8455	696	75	7643	19	17	0	3	1	1	90.4
Wheat	1923	114	0	10	1753	29	0	17	0	0	91.16
Oats	800	75	0	14	11	700	0	0	0	0	87.5
Alfalfa	65	31	16	0	0	0	18	0	0	0	27.7
Clover	619	87	11	403	4	12	0	465	0	0	75.12
Hay/Grass	142	64	11	7	0	0	0	0	60	0	42.25
Unknown	396	0	0	0	12	0	0	0	0	384	97
% +ve Rate		74.7	90.1	80.96	80.67	78.3	100	91.36	95.24	92.1	

TABLE XI
CLASSIFICATION CONFUSION MATRIX FOR THE MULTISPECTRAL URBAN AREA AFTER APPLYING THE MRF MODELING

Class	Total Points	Trees	Streets	Water	Buildings	Earth	% True per Class
Trees	212	212	0	0	0	0	100
Streets	521	2	496	0	3	20	95.2
Water	595	0	0	595	0	0	100
Buildings	292	1	34	0	257	0	88
Earth	410	0	4	0	0	406	99
% +ve Rate		98.6	92.88	100	98.85	95.31	

Mexico, in the U.S. This dataset has a 1-m resolution and there are nine classes defined in it: Unclassified/Shadow, Water, Trees, Buildings, Asphalt Roads, Scrub Shrub/Herbaceous, Sand/Soil/Gravel, Riverine Wetland, and Fiverine Substrate. The ground truth labels are available for the whole dataset.

The confusion matrix for the classification based on the density estimation using MF-based SVM is shown in Table VIII. The average true classification accuracy for the classes is 73%. The largest source of error is due to the misclassification between the Trees and other classes, with 18.8% of the Trees reference pixels are classified to the other classes and 19.6% from the other classes are classified to Trees. Due to the inherent similarities of the materialistic structure between the Scrub Shrub/Herbaceous (low vegetation) and the Trees, there is a significant misclassification between these two classes. There, 11.4% from the Trees points are classified as Scrub Shrub/Herbaceous and 40.78% from the Scrub Shrub/Herbaceous are classified as Tress. Other similar error can be seen between the Sand/Soil/Gravel and Fiverine Substrate classes. The MF-based SVM density estimation algorithm outperforms the other algorithm which is clear from Table IX.

The 95% confidence intervals in this experiment are [0.659 0.663], [0.725 0.729], [0.696 0.7], and [0.807 0.811] which emphasizes the better performance of the proposed density estimator in hyperspectral spaces.

D. Experiments on MRF Modeling and the Proposed Classification Setup

In this section, experiments are presented to illustrate the effectiveness of the MRF modeling and the proposed segmentation setup (iterative setup) on the improvement of the segmentation results. In the proposed segmentation setup, as shown in Section V, an initial guess for the segmented image is obtained by a pixelwise Bayes classifier with the MF-based SVM density estimator (as discussed before in the experimental work). Then, the parameter γ in (45) of the MRF model is calculated from the segmented image, the MAP segmentation in (2) is applied and the log-likelihood from (3) is calculated. If there is a significant difference between the consecutive values of the log-likelihood, the γ value is recalculated, and the segmentation procedure is repeated again. Otherwise, if there is no significant change in the log-likelihood values, the segmentation process is ended. The multispectral and hyperspectral datasets presented before are used in the experiments for the proposed MRF modeling and the segmentation setup evaluation in this section. In all experiments, the evolutions of the log-likelihood with the iterations are shown and the final segmentation results are presented.

1) Experiments Using Multispectral Data: In Fig. 11(a), the evolution of the log-likelihood for the multispectral agricultural area dataset is shown. It is clear that the log-likelihood is monotonically increasing which is the expected case. Also, the log-likelihood converges and starts to saturate without major changes after five iterations of the proposed algorithm which means that the segmentation process gradually converges to the final possible solution. The final segmented image in Fig. 12(a) illustrates the significant effect of the CPP modeling on the segmentation results. The apparent improvement can be noted from within the regions themselves. Comparing the classified

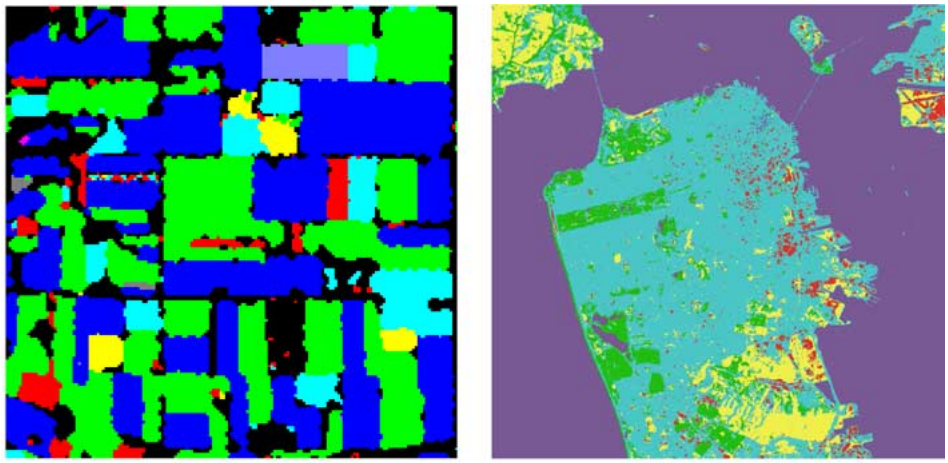


Fig. 12. Final segmented image with the proposed segmentation setup for the multispectral (a) agricultural area example and (b) urban area example.

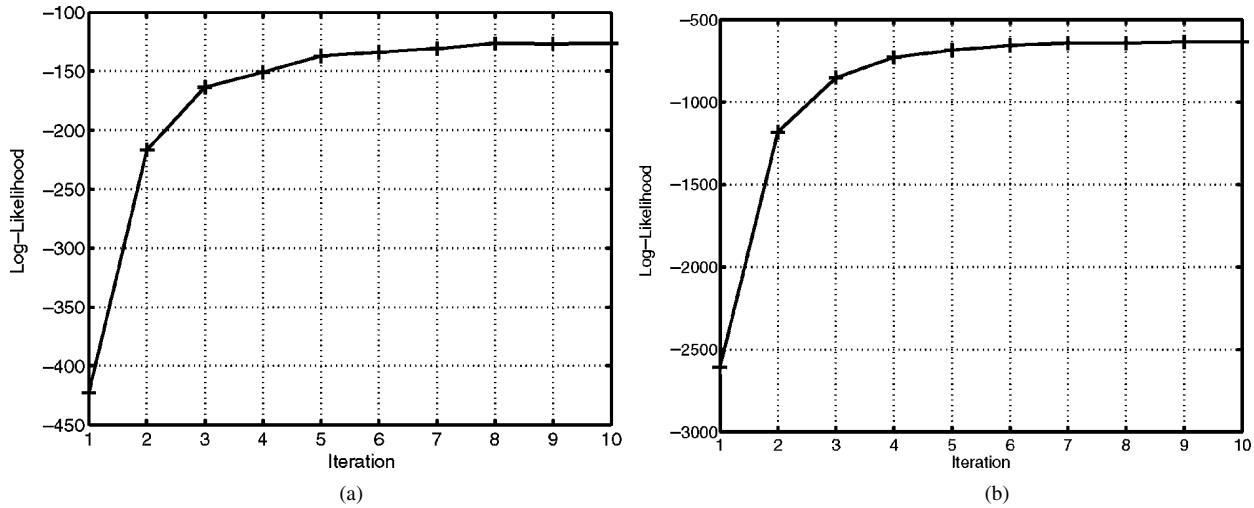


Fig. 13. Evolution of the log-likelihood in the hyperspectral (a) 34-band area example and (b) 58-band area example.

image in Fig. 7 with that of Fig. 12(b), it can be easily noted that there is a smoothness effect over the regions of the segmented image due to the incorporation of the contextual information with MRF modeling.

The confusion matrix in Table X illustrates quantitatively the improvements in the classification results. The average class accuracy rate increases to 84.3%, and the individual accuracy for each class increases also, especially the accuracies for the low-represented classes (e.g., Hay/Grass and Alfalfa). Also, the confidence in the points assigned to each class increases (see last rows in Tables II and X).

For the multispectral urban area, Fig. 11(b) illustrates the evolution of the log-likelihood, which converges and starts to saturate without major changes after 6 iterations in this experiment. The final segmented image in Fig. 12(b) and the confusion matrix in Table XI illustrate the improvement effect of the CPP modeling on the segmentation results. The average class accuracy rate increases to 97%, and both the individual class accuracies and the confidence in the points assigned to each class increase too, although the changes are not so much since the classification without MRF modeling is already good.

2) Experiments Using Hyperspectral Data: The evolution of the log-likelihood for the hyperspectral urban areas sets are shown in Fig. 13. The log-likelihood converges and starts to saturate without major changes after eight iterations for the 34-band dataset, and after seven iterations for the 58-band dataset. The final segmented images in Fig. 14 and the confusion matrices in Tables XII and XIII illustrate the improvement effect of the CPP modeling on the segmentation results. The average class accuracy rates increases to 83.75% in the 34-band dataset and to 83.38% in the 58-band dataset. Also, both the individual class accuracies and the confidence in the points assigned to each class increase for most of the classes.

VII. Concluding Remarks

This paper presented a complete framework for the segmentation of remote sensing datasets using the MAP estimation setup. The proposed mean field-based SVM algorithm for the class conditional probability density estimation is examined using gradually more challenging experiments in various dimensions. The results show that this algorithm outperforms

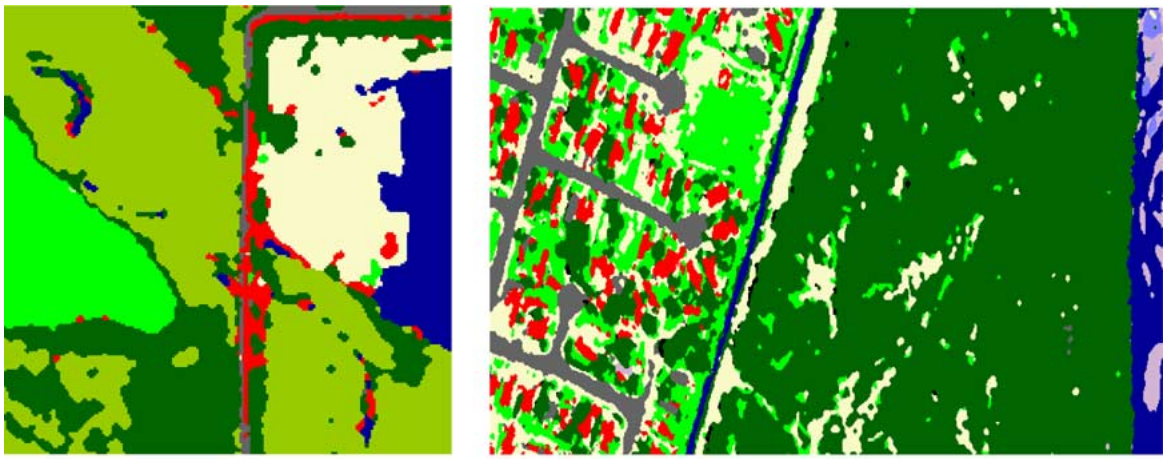


Fig. 14. Final segmented image with the proposed segmentation setup for the hyperspectral (a) 34-band area example and (b) 58-band area example.

TABLE XII
CLASSIFICATION CONFUSION MATRIX FOR THE HYPERSPECTRAL 34-BAND URBAN AREA AFTER APPLYING THE MRF MODELING

Class	Total Points	Agricultural	Coniferous	Herbaceous	Other Impervious	Roads	Soil	Water	% True per Class
Agricultural	5138	4177	209	508	142	10	90	2	81.3
Coniferous	15182	0	13270	1883	21	0	0	8	87.4
Herbaceous	7481	8	939	6124	196	155	23	36	82.86
Other Impervious	925	1	64	152	663	41	3	1	71.68
Roads	627	0	12	51	33	529	2	0	84.37
Soil	6362	92	4	448	309	1	5450	58	85.66
Water	4285	3	419	265	204	0	107	3287	76.7
% +ve Rate		97.57	88.95	64.93	42.28	71.88	96.03	96.9	

TABLE XIII
CLASSIFICATION CONFUSION MATRIX FOR THE HYPERSPECTRAL 58-BAND URBAN AREA AFTER APPLYING THE MRF MODELING

Class	Total Points	Shadow	Water	Trees	Buildings	Asphalt	Scrub	Sand	Wetland	Substrate	% True per Class
Shadow	1189	404	1	682	15	6	27	54	0	0	55.26
Water	7382	0	7157	102	2	0	6	45	0	70	96.95
Trees	91883	9	60	83609	634	240	4933	2322	0	76	91.00
Buildings	11655	8	4	1167	6495	773	1221	1905	0	82	55.73
Asphalt	9349	29	0	160	15	7383	130	1332	0	0	82.18
Scrub	30542	9	10	13244	136	10	15808	1324	0	1	52
Sand	25741	12	26	2625	2285	1772	853	18127	0	41	70.42
Wetland	437	0	1	8	0	0	0	2	386	40	88.33
Substrate	1822	0	154	10	0	0	0	26	32	1600	87.82
% +ve Rate		85.77	96.54	82.28	67.78	72.49	68.79	72.11	92.34	83.38	

other algorithms in average accuracy and time considerations. The learning procedure for the SVM which is based on the MF theory reduces the optimization process which makes the algorithm much faster. In the 1-D Gaussian example for density estimation, the optimization time is reduced from 0.22 s to 0.015 while it is reduced from 0.578 to 0.172 s in the 2-D Gaussian example. The closeness of the KLD measures to each other reflect the better accuracy of the proposed density estimation algorithm in the 1-D case, the measures values are $\mathrm{KLD}(p_1 \mid p_0) = 0.12$ and $\mathrm{KLD}(p_0 \mid p_1) = -0.094$ while they are $\mathrm{KLD}(p_1 \mid p_0) = -0.85$ and $\mathrm{KLD}(p_0 \mid p_1) = 1.86$ in the traditional formulation case. In the 2-D case, the distances are $\mathrm{KLD}(p_1 \mid p_0) = 4.029$ and $\mathrm{KLD}(p_0 \mid p_1) = -3.6$ while they are $\mathrm{KLD}(p_1 \mid p_0) = 0.39$ and $\mathrm{KLD}(p_0 \mid p_1) = -8.4$ in the traditional case.

The experiments using real remote sensing data illustrate also the better performance of the proposed density estimation algorithm. The segmentation accuracy increases by up to 5% upon using the MF-based SVM density estimation algorithm.

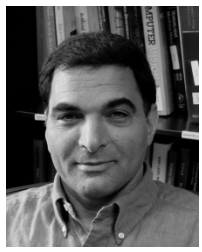
The MRF modeling for the image regions with the proposed iterative setup for the segmentation framework increases the segmentation accuracy significantly. A gain of up to 10% in segmentation accuracy is achieved (e.g., the 58-band hyperespectral dataset). The smoothness and cleanness of the regions in the segmented image can be easily observed. In conclusion, the experiments show that the proposed framework is a promising setup for applying the MAP estimation principle in remote sensing imagery segmentation.

The main point for future work to complete the proposed framework is to make it as automatic as possible. The automation includes the choice of the parameters for the learning algorithm, the best shape for the kernel, and also the best parameters for the kernel. Introducing shape constraints in the segmentation process is another critical concern.

REFERENCES

[1] D. Landgrebe, *Signal Theory Methods in Multispectral Remote Sensing*. New York: Wiley, 2003.
[2] ——, "Hyperspectral image data analysis," *IEEE Signal Process. Mag.*, no. 1, pp. 17–69, Jan. 2002.
[3] M. Dundar and D. Landgrebe, "Toward an optimal supervised classifier for the analysis of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 1, pp. 271–277, Jan. 2004.
[4] T. Lee and M. Lewicki, "Unsupervised image classification, segmentation, and enhancement using ICA mixture models," *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 270–279, Mar. 2002.
[5] O. Pichler, A. Teuner, and B. Hosticka, "An unsupervised texture segmentation algorithm with feature space reduction and knowledge feedback," *IEEE Trans. Image Process.*, vol. 7, no. 1, pp. 53–61, Jan. 1998.
[6] L. Bruzzone and D. Prieto, "An adaptive semiparametric and context-based approach to unsupervised change detection in multitemporal remote-sensing images," *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 452–466, Apr. 2002.
[7] M. Rezaee, P. Van Der Zwet, B. Lelieveldt, R. Van Der Geest, and J. Reiber, "A multiresolution image segmentation technique based on pyramidal segmentation and fuzzy clustering," *IEEE Trans. Image Process.*, vol. 9, no. 7, pp. 1238–1248, Jul. 2000.
[8] A. Sarkar, M. Biswas, and K. Sharma, "A simple unsupervised MRF model based image segmentation approach," *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 801–812, May 2000.
[9] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, 2nd ed. New York: Wiley, 2000.
[10] P. Pellegretti, F. Roli, S. Serpico, and G. Vernazza, "Supervised learning of descriptions for image recognition purposes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 1, pp. 92–98, Jan. 1994.

[11] S. Sarkar and P. Soundararajan, "Supervised learning of large perceptual organization: Graph spectral partitioning and learning automata," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 5, pp. 504–525, May 2000.
[12] B. Love, "Comparing supervised and unsupervised category learning," *Psychol. Bull. Rev.*, vol. 9, no. 4, pp. 829–35, Dec. 2002.
[13] V. Vapnik, *The Nature of Statistical Learning Theory*, 2nd ed. New York: Springer, 2001.
[14] R. M. Mohamed and A. A. Farag, "Classification of multispectral data using support vector machines approach for density estimation," in *IEEE 7th Int. Conf. Intelligent Engineering Systems*, Assiut, Egypt, Mar. 2003.
[15] V. Vapnik, S. Golowich, and A. Smola, "Support vector method for multivariate density estimation," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 1999, vol. 12, pp. 659–665.
[16] B. Scholkopf, C. Burges, and A. Smola, *Advances in Kernel Methods-Support Vector Learning*. Cambridge, MA: MIT Press, 1999.
[17] M. Opper and O. Winther, "Gaussian processes for classification: Mean field algorithms," *Neural Comput.*, vol. 12, pp. 2655–2684, 2000.
[18] J. Besag, "On the statistical analysis of dirty pictures," *J. R. Statist. Soc.*, vol. B48, no. 3, pp. 259–302, 1986.
[19] C. Bouman and M. Shapiro, "A multiscale random field model for Bayesian image segmentation," *IEEE Trans. Image Process.*, vol. 3, no. 2, pp. 162–177, Mar. 1994.
[20] A. A. Farag and E. Delp, "Image segmentation based on composite random field models," *J. Opt. Eng.*, vol. 12, p. 25 942 607, Dec. 1992.
[21] G. Gimel'farb, *Image Textures and Gibbs Random Fields*. Dordrecht, The Netherlands: Kluwer, 1999.
[22] A. S. El-Baz and A. A. Farag, "Image segmentation using GMRF models: Parameters estimation and applications," in *Proc. IEEE Int. Conf. Image Processing*, Barcelona, Spain, Sep. 14–17, 2003, pp. II 173–176.
[23] A. Jain and R. Dubes, "Random field models in image analysis," *J. Appl. Statist.*, vol. 16, no. 2, pp. 131–164, 1989.
[24] J. Lamperti, *Probability-A Survey of the Mathematical Theory*. New York: Wiley, 1996, Series in Probability and Statistics.
[25] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer-Verlag, 2001.
[26] A. Papoulis and S. Pillai, *Probability, Random Variables and Stochastic Processes*, 4th ed. New York: McGraw-Hill, 2001.
[27] J. Gao, S. Gunn, and C. Harris, "Mean field method for the support vector machine regression," *Neurocomputing*, vol. 50, pp. 391–405, Nov. 2003.
[28] A. A. Farag, R. M. Mohamed, and H. Mahdi, "Experiments in image classification and data fusion," in *Proc. 5th Int. Conf. Information Fusion*, Annapolis, MD, Jul. 11–17, 2002, pp. I 299–308.
[29] R. M. Mohamed and A. A. Farag, "Two stages classifier for multispectral data," presented at the *Int. Conf. Computer Vision and Pattern Recognition, Workshop on Learning in Computer Vision and Pattern Recognition*, Madison, WS, Jun. 16–22, 2003, LCVPR03.
[30] J. Besag, "Spatial interaction and the statistical analysis of lattice system," *J. R. Statist. Soc.*, vol. B36, no. 2, pp. 192–225, 1974.
[31] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distribution, and Bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-6, no. 6, pp. 721–741, Jun. 1984.
[32] H. Derin and H. Elliot, "Modeling and segmentation of noisy and texture images using Gibbs random field," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-9, no. 1, pp. 39–55, Jan. 1987.
[33] S. Kullback, *Information Theory and Statistics*. New York: Wiley, 1959.
[34] M. Girolami and C. He, "Probability density estimation from optimally condensed data samples," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 10, pp. 1253–1264, Oct. 2003.



Aly Farag (SM'99) received the B.S. degree in electrical engineering from Cairo University, Cairo, Egypt, the M.S. degree in biomedical engineering from The Ohio State University, Columbus, the M.S. degree in bioengineering from the University of Michigan, Ann Arbor, and the Ph.D. degree in electrical engineering from Purdue University, West Lafayette, IN.

He joined the University of Louisville in August 1990, where he is currently a Professor of electrical and computer engineering. He is the founder and director of the Computer Vision and Image Processing Laboratory (CVIP Lab) at the University of Louisville, which supports a group of over 20 graduate students and postdoctoral students. His contribution has been mainly in the areas of active vision system design, volume registration, segmentation, and visualization, where he has authored or coauthored over 130 technical articles in leading journals and international meetings in the fields of computer vision, image processing, and medical imaging. He is a regular reviewer for a number of technical journals and to national agencies including the NSF and the NIH.

Dr. Farag was awarded a University Scholar designation in 2002. He has been an Associate Editor of IEEE TRANSACTIONS ON IMAGE PROCESSING. He is a Senior Member of the SME.

Refaat Mohamed (S'03) received the B.Sc. (with honor, first on class) and M.Sc. (very good) degrees in electrical engineering from Assiut University, Assiut, Egypt, in 1995 and 2001, respectively. He is currently pursuing the Ph.D. degree at the University of Louisville, Louisville, KY.

Since August 2001, he has been with the Computer Vision and Image Processing Laboratory, University of Louisville, as a Research Assistant. His current research interests are in statistical modeling, pattern recognition, remote sensing data analysis, and computer vision.

Mr. Mohamed received the Speed School of Engineering Award for Outstanding Student, Research and Creative Activity in 2004. He is a member of nationally and internationally honorable organizations and is a member of Eta Kappa Nu "HKN" and nominated for *Who's Who Among Students in American Universities* in 2005.

Ayman S. El-Baz (S'03) received the B.Sc. and M.S. degrees in electrical engineering from Mansoura University, Mansoura, Egypt, in 1997 and 2000, respectively. He is currently pursuing the Ph.D. degree at University of Louisville, Louisville, KY.

He joined the Computer Vision and Image Processing Laboratory at the University of Louisville in May 2001. During his stay at the University of Louisville, he has been involved in the applications of image processing and computer vision for medical image analysis. His current research includes image modeling, image segmentation, 2-D and 3-D registration, visualization, and surgical simulation including finite element analysis, where he has authored or coauthored more than 35 technical articles.

Mr. El-Baz was the recipient of the Research Louisville Award in 2002 and Speed School Award for Outstanding Student, Research and Creative Activity in 2003. He is a regular reviewer for a number of technical journals and conferences including the IEEE TRANSACTIONS ON IMAGE PROCESSING and the International Conference on Medical Image Computing and Computer Assisted Intervention. He is a member of member of Eta Kappa Nu.