ELSEVIER

Image and Vision Computing 21 (2003) 87–97

image AND vision COMPUTING

www.elsevier.com/locate/imavis

An observation-constrained generative approach for probabilistic classification of image regions

Sanjiv Kumar[a,*], Alexander C. Loui[b], Martial Hebert[a]

[a]*The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA*
[b]*Imaging Science and Technology Lab, Eastman Kodak Company, Rochester, NY, USA*

Abstract

In this paper, we propose a probabilistic region classification scheme for natural scene images. In conventional generative methods, a generative model is learnt for each class using all the available training data belonging to that class. However, if an input image has been generated from only a subset of the model support, use of the full model to assign generative probabilities can produce serious artifacts in the probability assignments. This problem arises mainly when the different classes have multimodal distributions with considerable overlap in the feature space. We propose an approach to constrain the class generative probability of a set of newly observed data by exploiting the distribution of the new data itself and using linear weighted mixing. A Kullback–Leibler Divergence-based fast model selection procedure is also proposed for learning mixture models in a low dimensional feature space. The preliminary results on the natural scene images support the effectiveness of the proposed approach.

Keywords: Image region classification; Generative model; Semantic interpretation; Image segmentation

1. Introduction

Automatic extraction of the semantic context of a scene is useful for image indexing and retrieval, robotic navigation, surveillance, robust object detection and recognition, auto-albuming, etc. Recent literature reveals increasing attention to this field [18,20,21]. However, most of the attempts have been made to extract the context of a scene at a high level in the abstraction hierarchy. For example, Torralba et al. [20] represent the context by using the power spectra at different spatial frequencies, while Vailaya [21] uses the edge coherence histograms to differentiate between natural and urban scenes. Limited attention has been paid to the task of specific context generation from a scene (scene classification), e.g. if a scene is a *beach* or an *office*. The main hurdle in such context generation is that it requires not only the knowledge of the regions or objects in the image, but the semantic information contained in their spatial arrangement as well.

The motivation for our work comes from the following paradox of scene classification. In absence of any a priori information, the scene classification task requires the knowledge of regions and objects contained in the image. On the other hand, it is increasingly being recognized in vision community that context information is necessary for reliable extraction of the image regions and objects [18,20]. To solve this paradox, an iterative feedback scheme can be envisaged, which refines the scene context and image region hypotheses iteratively. Under this paradigm, an obvious choice for the region classification scheme is one that allows easy modification of the initial classification without requiring to classify the regions afresh. Probabilistic classification of image regions can provide great flexibility in future refinement using the Bayesian approach, as the context information can be encoded as improved priors.

In this paper, we deal with the probabilistic classification of image regions belonging to scenes primarily containing natural objects, e.g. water, sky, sand, skin, etc. as a priming step for the problem of scene context generation. Several techniques have been proposed to classify various image regions in distinct categories, e.g. Refs. [2,17]. However, these are primarily based on discriminative approaches leading to *hard* class assignments and hence are not suitable for the iterative refinement scheme mentioned above.

In conventional schemes [8,12,14,15], a generative model for each class is learnt using all the available training

* Corresponding author.
E-mail addresses: sanjiv@andrew.cmu.edu (S. Kumar), skumar@ri.cmu.edu (S. Kumar), alexander.loui@kodak.com (A.C. Loui), hebert@ri.cmu.edu (M. Hebert).

0262-8856/03/$ - see front matter © 2002 Elsevier Science B.V. All rights reserved.
PII: S0262-8856(02)00125-7

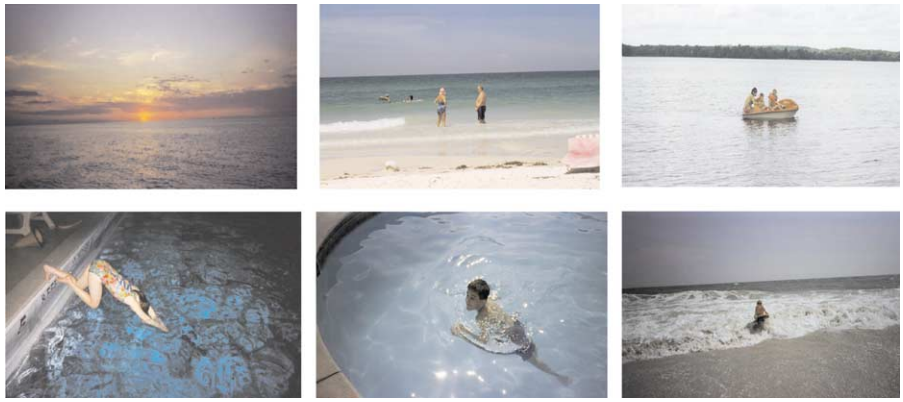

Fig. 1. A subset of the training image set shows wide variations in color and texture properties of the water regions.

data belonging to that class and newly observed data is assigned a probability based on the learnt model. However, it is possible that an input image has been generated from a subset of the full generative model support, and using the full model to assign generative probabilities can produce serious artifacts in the probability assignments. For example, in the training set, various pixels associated with the class *water* may have wide variations in the color and texture depending on the location, illumination conditions, and the scale. A subset of the training image set in Fig. 1 shows these variations. A generative model for class water will try to capture all these variations in the same model. Hence, while assigning probabilities to the newly observed data, the learnt generative model may assign high probability not only to the pixels belonging to the class water, but also to those that may have some semblance to water data (in some arbitrary combination of illumination and other physical conditions). Similar problems can arise with the other classes as well. This problem arises mainly when different classes have multimodal distributions that are close in feature space.

Another problem with generative models is that they tend to give more weight to regions in feature space that contain more data, instead of emphasizing the discriminative boundary between the data belonging to different classes. This implies that the data near the boundary will be assigned similar probabilities irrespective of their class affiliations.

We propose to alleviate these problems by using the simple observation that the newly observed data are usually generated from a small support of the overall generative model. In our previous water example, this means that the data belonging to the water class in a new test image is usually generated at the same location as well as under relatively homogeneous illumination and other physical conditions. Thus, the distribution of the newly observed data can be used to constrain the overall generative model when computing the generative class density maps for that data. That is why the proposed technique has been named as the 'observation-constrained generative approach'. The preliminary results on this approach were presented in Ref. [9].

This idea can be illustrated through the scatter plots of the class data. Fig. 2(a) shows the distribution of the water data from all the training images in a 2D feature space (normalized color space). Given an input test image (Fig. 4(a)), the distribution of the water data contained in this image, superimposed on the overall water data in Fig. 2(a) is shown in Fig. 2(b). Similarly, the distribution of the *sky* data from the input image along with the data in Fig. 2(b) is given in Fig. 2(c). Fig. 2(c) shows that the sky data is contained within the distribution of the overall water data. Thus, the sky regions in the input image will be assigned high probability of being generated from class water. However, it can be noted that the distributions of water and sky data from the input test image are fairly separable. Furthermore, it is clear from Fig. 2(b) that the distribution of the input water data is contained within a small support of the distribution of the overall water data. Thus, for the given test image, if the distribution of the input

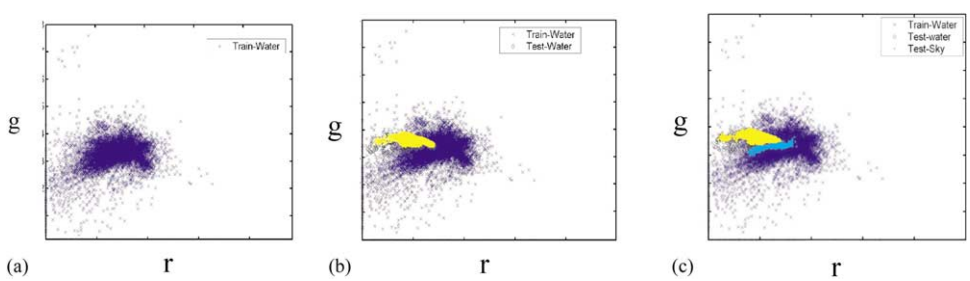

Fig. 2. Scatter plots of class data in a 2D feature space. (a) Distribution of the water data from the training set. (b) Distribution of water data from a test image superposed on (a). (c) Distribution of sky data from the test image along with the data in (b).

water data could be used to constrain the full generative model for the class water, the potential errors in probability assignment to the sky data can be significantly reduced.

2. Generative approach

In the first stage of our approach, we use two well-known techniques, i.e. supervised learning applied to labeled training data, and unsupervised learning applied to test data. The main contribution of our approach lies in the next stage, where the outputs of these two techniques are merged using a Bayesian scheme. We begin by briefly reviewing the unsupervised and supervised learning methods applied in the present context.

Let X_ω be the set of training data associated with class ω, where $X_\omega = \{x_k^\omega : x_k^\omega \in \Re^d\}_{k=1}^{n_\omega}$; and Ω be the set of classes of interest. The set Ω is referred to as the *recognition vocabulary*. The dimension of the feature space, d, is the number of bands in the multiband input image. Each band represents a color or a texture feature associated with the image pixels. To capture this inherent multimodality, we assume a mixture-of-Gaussian generative model for the data. Mixture models can approximate any continuous density to arbitrary accuracy provided the model has sufficiently large number of components and the parameters of the model are chosen correctly [1]. The class conditional density function of a data point $x_k^\omega \in X_\omega$ is given by

$$p(x_k^\omega|\omega) = \sum_{m=1}^{M} p(x_k^\omega|m,\omega)P(m|\omega) \tag{1}$$

where M is the number of components in the mixture model. In this model, each data point x_k^ω belonging to class ω is generated by first choosing a Gaussian component with probability $P(m|\omega)$ and then generating the data point with probability $p(x_k^\omega|m,\omega)$, which is a Gaussian given by

$$p(x_k^\omega|m,\omega) = \frac{1}{|2\pi C_m^\omega|^{1/2}} \exp\left\{-\frac{1}{2}(x_k^\omega - \mu_m^\omega)^{\mathrm{T}} C_m^{\omega^{-1}}(x_k^\omega - \mu_m^\omega)\right\} \tag{2}$$

where μ_m^ω is the mean and C_m^ω is the covariance matrix of the component m, belonging to class ω.

The parameters θ_ω of the generative model in Eq. (1), i.e. μ_m^ω, C_m^ω and $P(m|\omega)$ for each component m are learnt using the standard Maximum Likelihood formulation using the Expectation Maximization (EM) optimization technique [4]. Thus,

$$\hat{\theta}_\omega = \arg\max_{\theta_\omega} \prod_{k=1}^{n_\omega} \sum_{m=1}^{M} p(x_k^\omega|m,\omega)P(m|\omega). \tag{3}$$

In the above formulation, the data points in the set X_ω have been assumed to be conditionally independent given θ_ω.

For a given test image I, the newly observed dataset is defined as $X_I = \{x_t^I : x_t^I \in \Re^d\}_{t=1}^N$ where N is the number of pixels in I. The probability of association of each x_t^I with a given class in the recognition vocabulary (given by Eq. (1)) yields a Class Density Map over the image I. Now, the aim is to constrain the probabilities contained in these maps by enforcing the statistics of the newly observed data obtained from the test image. To do this, at first, the newly observed data is *soft* clustered in an unsupervised manner. It should be noted that the clusters in an image signify the prominent groups of pixels, not the classes. For soft clustering, we again use the mixture of the Gaussian model. According to this, the probability of a newly observed data point, x_t^I is given by

$$p(x_t^I) = \sum_{j=1}^{K} p(x_t^I|j)P(j) \tag{4}$$

where $j = 1,\ldots,K$ are the clusters and $p(x_t^I|j) \sim N(\mu_j, C_j)$ similar to Eq. (2). Note that, instead of hard clustering (where each image pixel is assigned to a particular cluster), we utilize the probability of association of each pixel with a cluster j, i.e. $P(j|x_t^I)$. This leads to soft or probabilistic clustering and the maps representing these probabilities for each cluster are called Cluster Probability Maps. These maps enable the refinement of the Class Density Maps by constraining the overall generative model. To estimate the parameters of the unsupervised learning model mentioned in Eq. (4), similar EM formulation is used as in Eq. (3) except that m and x_k^ω are replaced by j and x_t^I, respectively, along with the suitable cardinalities of their corresponding sets.

Given the test image I, the first step towards constraining the Class Density Maps using soft clustering involves finding the class that has highest probability of being represented by a Cluster Probability Map. This is done by maximizing the conditional probability of class ω_c, $(c = 1,\ldots,\Omega)$ given a cluster j, i.e. $P(\omega_c|j)$. To compute this probability, first the marginal density of the cluster j given class ω_c is obtained from the joint density of the newly observed data x and the cluster as

$$p(j|\omega_c) = \int_{S_x} p(j,x|\omega_c)\mathrm{d}x$$

or

$$p(j|\omega_c) = \int_{S_x} p(j|x,\omega_c)p(x|\omega_c)\mathrm{d}x. \tag{5}$$

It should be noted that j is defined only over the support of newly observed data S_x, which has been explicitly mentioned in the above integrals. We further make the reasonable assumption that clustering is conditionally independent of the class given the data, i.e. $p(j|x,\omega_c) = p(j|x)$. Thus,

$$p(j|\omega_c) = \int_{S_x} p(j|x)p(x|\omega_c)\mathrm{d}x.$$

Since the image data is discrete, the integral over S_x is approximated by the finite sum and the cluster conditional is given by

$$p(j|\omega_c) = \sum_{x \in X_I} P(j|x)p(x|\omega_c). \tag{6}$$

Using Eq. (6), the class posterior can be computed easily from Bayes rule:

$$P(\omega_c|j) = \frac{p(j|\omega_c)P(\omega_c)}{\sum_{c=1}^{\Omega} p(j|\omega_c)P(\omega_c)}. \tag{7}$$

Given the class posterior for a cluster, the class $\hat{\omega}$ that maximizes this posterior is chosen as the consistent class for that cluster. This is equivalent to a MAP selection of the class given a cluster under the assumption of zero–one loss function. This procedure yields the most consistent class for each cluster. An implicit assumption has been made that each cluster belongs to one of the classes in the recognition vocabulary. Let K_ω be the set of the clusters that are consistent with the class ω. In other words, K_ω contains all those clusters that yield ω as the MAP estimate of their corresponding class by Eq. (7). The Constrained Class Density Map is obtained by linear weighting of each pixel density of the Class Density Map by the corresponding pixel probability of the consistent Cluster Probability Maps, i.e.

$$p_{\text{cons}}(x_t^I|\omega) = \sum_{j \in K_\omega} p_{\text{orig}}(x_t^I|\omega)P(j|x_t^I) \tag{8}$$

where $p_{\text{orig}}(\cdot|\cdot)$ is the original and $p_{\text{cons}}(\cdot|\cdot)$ is the constrained class conditional density. An intuitive explanation of Eq. (8) is that the constrained map is a linearly mixed, weighted density map where weights approximately follow a Gaussian distribution in the feature space because each cluster map approximately corresponds to a Gaussian distribution. It can be noted that the final constrained density map for a given class tends to enhance those areas in the original density map that are supported by the statistics of the regions in the test image that show strong association with the given class.

3. Model selection

In the proposed generative approach, we need to know the number of the components in the Gaussian mixture models in Eq. (1) as well as in Eq. (4), which amounts to the problem of model selection. The maximum likelihood approach is not appropriate for this task, as it would always favor more components. Several techniques have been proposed for model selection [5–7,13]. Full Bayesian techniques provide a more principled method of model selection and generally use a parametric or hierarchical form to approximate the prior distribution over the parameters [7]. The BIC (Bayes Information Criterion) or MDL (Minimum Description Length) approach, as proposed by Rissanen [13] can be shown to be an asymptotically consistent version of the full Bayesian model selection techniques. In the MDL technique, the description length (DL) is given by

$$\text{DL} = -\log p(X|\theta) + (l/2)\log n \tag{9}$$

where X is the data, θ is the parameter vector containing all the model parameters, l is the number of parameters and n is the dataset size. The first term in the right hand side of Eq. (9) is the negative log likelihood (i.e. code length of likelihood) and the second term is the code length of the parameters, which acts as a penalty term as the number of parameters increases.

However, there are two main limitations while implementing the full Bayesian or MDL criterion. First, they generally need iterative schemes to compute the model evidence or the likelihood, which is prone to getting stuck in local extrema and second, they are fairly slow and become impractical for the online soft clustering applications. In the present work, we propose a Kullback–Leibler Divergence (KLD)-based method to estimate the number of components in the mixture model based on certain assumptions. The KLD is defined as

$$\text{KL}(\tilde{p}\|p) = \int \tilde{p}(x)\log\frac{\tilde{p}(x)}{p(x)}\,\mathrm{d}x \tag{10}$$

where $\tilde{p}(x)$ is the true density of data x and $p(x)$ is the model density.

To apply the KLD in the component selection procedure, the model density is given by the mixture of Gaussians, but the true density is not known. We assume the data histogram (empirical distribution) to be the representative of the true density. This assumption is reasonable in the case of natural scene images, as they are mostly composed of smooth and homogeneous regions.

3.1. Estimation of model density

The model density is estimated by incrementally fitting Gaussians on the modes of the data histogram. For this purpose, we use second order normal approximation of the empirical distribution [19]. Parameters of the Gaussian are obtained by matching the curvature of the Gaussian with that of the empirical density at the mode. This is equivalent to using the inverse of the information matrix at the mode of the empirical density as the covariance matrix of the Gaussian. The expected information matrix is given by

$$J(x) = E\left\{\left[\frac{\partial}{\partial x}\log p(x)\right]\left[\frac{\partial}{\partial x}\log p(x)\right]^{\mathrm{T}}\right\}$$

where E is the expectation with respect to x. $J(x)$ is approximated by its value at the mode $\hat{x}$ of $p(x)$, and the stochastic or observed information matrix is given by:

$$I(x) = \left[\frac{\partial}{\partial x}\log p(x)\right]\left[\frac{\partial}{\partial x}\log p(x)\right]^{\mathrm{T}}\Bigg|_{\hat{x}}. \tag{11}$$

By using $I(x)$, the covariance matrix C_g of the approximated Gaussian is given by

$$C_g = I^{-1}(x). \tag{12}$$

Since we are using the discrete empirical distribution as $p(x)$, computing $I(x)$ in Eq. (11) is equivalent to computing the negative of the hessian of the best fit curve at the mode of the log distribution. The neighborhood of $\log p(x)$ at the mode was assumed to be locally quadratic for computational purposes. Let $\Gamma(\hat{x}) = \{x_\rho : x_\rho \in \Re^d\}_{\rho=1}^{n_\Gamma}$ be the set of all data points in the neighborhood of the mode $\hat{x}$. In the local neighborhood of $\hat{x}$, the log-density can be expressed as

$$\log p(x_\rho) = \tfrac{1}{2} x_\rho^{\mathrm{T}} H x_\rho + B^{\mathrm{T}} x_\rho + c \tag{13}$$

where H is the Hessian matrix, and B and c are other constants. The hessian can be computed using Singular Value Decomposition (SVD) on the local neighborhood at the mode of the histogram.

In the proposed method, the first estimate of the model density is obtained by fitting the first Gaussian at the mode of the histogram and the KLD between the empirical and the model density is computed. The integral in Eq. (10) is approximated by the finite sum over the discrete bin data. Next, a new density is computed by fitting one more Gaussian on the next highest mode of the histogram and mixing the two Gaussians. The mixing parameters have been assumed uniform. The modes are selected to be the maxima in their local neighborhood of a prespecified size. As per the above formulation, as the number of Gaussians is increased, KLD decreases and starts increasing when the number of Gaussians grows beyond that supported by the data distribution. In addition, it should be noted that if the KLD remains stable as the components are increased, it is an indicator that one or more of the previous components has already explained the new modes. This technique works fairly well in low dimensional, sparse feature space (as is the case with the images containing natural regions) as finding the modes in the local neighborhood of a histogram is relatively easy.

4. Feature extraction

The type of features to be extracted from an image depends on the nature of the scene classification task. In the present work, we deal with the scene images primarily containing natural regions. Although not sufficient, low-level features such as color and texture contain good representation power for the region classification of natural scenes.

4.1. Color features

Color is an important component of the natural scene classes. However, the color-based features suffer from the problem of color constancy. For natural scenes, we argue that given enough variations in the training data set, we can capture the class color distribution in varying illumination conditions. In addition, in outdoor natural scenes, the problem of artificial illuminants is not as serious as that of changes in brightness.

To extract the color features, we need to represent the color in a suitable space. In the present work, we found that the results with three different spaces i.e. rectangular HSV, g-RGB, and L*uv* were similar. We chose generalized RGB (g-RGB) space, as it normalizes the effects of brightness variations effectively. The g-RGB space is given by two coordinates, $(r = R/(R+G+B), \quad g = G/(R+G+B))$ where (R, G, B) are the coordinates in the RGB space. In our method, we make no use of brightness $(R+G+B)$ information.

4.2. Texture features

In contrast to color, texture is a not a point property. Instead, it is defined over a spatial neighborhood. Texture can provide good discrimination between natural classes similar in the color domain e.g. water and sky. However, in our case, texture should be dealt with more care, as any single class does not have a unique texture. Within a semantically coherent region, there might be areas of high or low textures, with different scales and directional uniformity. In the classification of such regions, a very strong texture measure can sometimes undo the good work done by the color features.

Several techniques have been reported in the literature to compute the texture in a pixel neighborhood. The famous ones include Multiresolution Simultaneous Autoregressive (MSAR) model [11], Gabor Wavelets [10] and the Second Moment Eigenstructure (SME) [3,16]. In the present work, we have used a weaker measure of texture yielded by the Second Moment Eigenstructure (SME), which can capture the essential neighborhood characteristics of a pixel. The second moment matrix at each image pixel (i, j) is given by:

$$M(i,j) = \begin{bmatrix} \sum_\tau \sum_W (I_x^\tau)^2 & \sum_\tau \sum_W I_x^\tau I_y^\tau \\ \sum_\tau \sum_W I_x^\tau I_y^\tau & \sum_\tau \sum_W (I_y^\tau)^2 \end{bmatrix} \tag{14}$$

where I_k^τ is the gradient of the image in spatial direction k over the color channel τ for $k = x, y$ and $\tau = R, G, B$. W is the window around pixel (i, j) over which these gradients are summed. We use Gaussian weighting in window W around the pixel of interest to give more weight to the pixels near it. The texture obtained from Eq. (14) is a *colored texture* because the above matrix captures the texture in color space instead of usual intensity space. The second moment matrix can be shown to be a modification of bilinear, symmetric positive definite metric defined over the 2D image manifold embedded in a 5D space of (R, G, B, i, j)

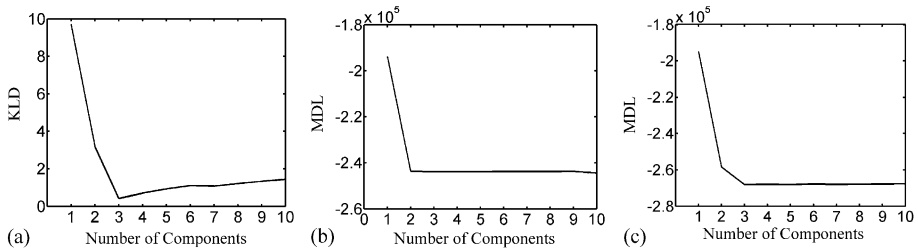

Fig. 3. Model selection results on the synthetic data. (a) KLD-based model selection on the synthetic data. KLD correctly favors three components. (b) A typical run of MDL incorrectly favors two components. (c) Another typical run of MDL correctly favors three components.

[16]. The eigenstructure of the second moment matrix represents the textural properties. Two measures have been defined in Ref. [3] using the eigenvalues λ_1 and λ_2 are the two eigenvalues of matrix $M(i,j)$ and $\lambda_1 > \lambda_2$: (a) anisotropy $= 1 - \lambda_2/\lambda_1$, and (b) normalized strength $= 2\sqrt{(\lambda_1 + \lambda_2)}$. In the present work, we have used the product of anisotropy and normalized strength as the texture measure. This product is defined as *texture strength* (S).

5. Results and discussion

This section is divided into two subsections. Section 5.1 discusses the simulation results of the proposed KLD-based model selection scheme on synthetic data, and Section 5.2 contains the results of the proposed observation-constrained generative approach applied to real natural scenes.

5.1. Model selection results on the synthetic data

To verify the effectiveness of the proposed KLD-based model selection scheme, and compare the results with the MDL-based approach, we applied these techniques on a simulated dataset containing 100,000 samples drawn from a mixture of three Gaussians in two-dimensional space. The mixing parameters used in the mixture model were 0.7, 0.2, and 0.1. The KLD-based model selection scheme was applied to the data histogram and the size of neighborhood for the surface fitting required for Gaussian approximation was chosen to be 3×3. Fig. 3(a) shows the change in KLD as the number of components is increased in the Gaussian mixture. The KLD falls sharply when the components are increased from 1 to 3 and then starts increasing. Thus, this method correctly finds the number of components in the mixture data.

For the comparison, MDL was computed for the given dataset using Eq. (9). The maximum likelihood estimates of the parameters were obtained using EM. Because EM is sensitive to the initialization, we performed the MDL computation several times. Fig. 3(b) and (c) show two typical plots of the change in description length as the number of components is increased. In Fig. 3(b), the MDL metric incorrectly favors two components, possibly because the EM algorithm was stuck in a local maximum. In another typical run when EM yielded the correct parameters caused by convergence to the global maximum, the MDL metric correctly favors three components (Fig. 3(c)). Thus, it can be seen that even for a very low dimensional space, MDL is not robust when EM is not initialized in proximity of the global maximum. In addition, the average time taken by a typical MDL run was about 935 s (in Matlab, 733 MHz, Pentium III) while that by KLD was only 0.21 s. Thus, we can get substantial savings in terms of speed by using KLD. This is especially important for the unsupervised clustering in the test image, which has to be done online unlike the supervised learning.

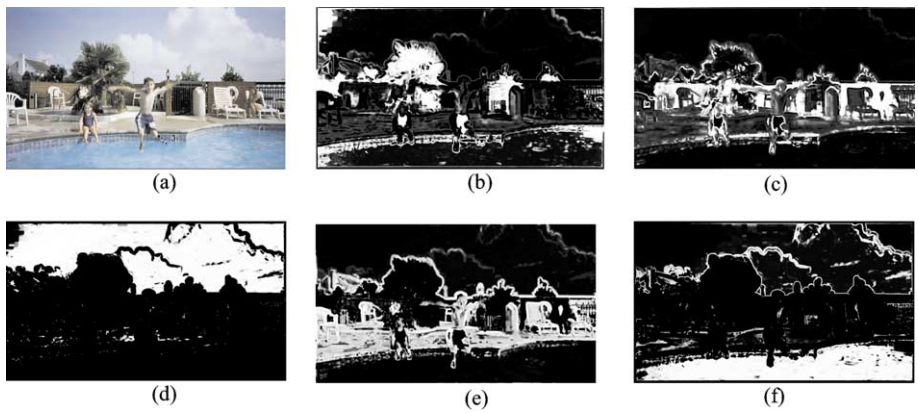

Fig. 4. Cluster Probability Maps corresponding to different clusters (j) obtained from the soft clustering of the input image. (a) Input color image. (b) $j = 1$. (c) $j = 2$. (d) $j = 3$. (e) $j = 4$. (f) $j = 5$.

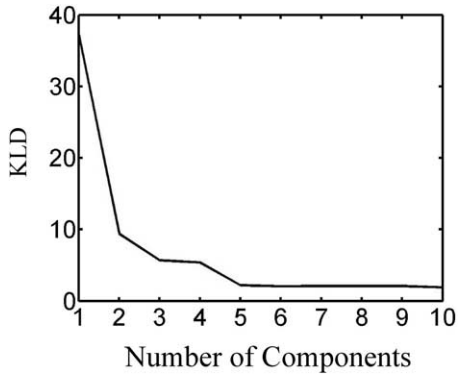

Fig. 5. KLD-based component selection for the image from Fig. 4(a). The selection scheme favors five components.

5.2. Results on real images

The proposed observation-constrained generative approach was tested with a test set containing about forty images primarily containing natural regions. The class recognition vocabulary contained five classes: *sky*, *water*, *skin*, *sand/soil*, and *grass/tree*. A total of 130 pixelwise labeled images, each of size 499 × 874 pixels, were used as a training set for the supervised learning.

Generative mixture models were learnt for each class in the recognition vocabulary. The feature sets corresponding to all the pixels belonging to each class (in 130 training images) were used. The number of components in each class conditional mixture model was learnt using the KLD-based technique explained in Section 5.1.

An input color image is shown in Fig. 4(a). It can be noted that the scene is fairly cluttered and contains regions with varying illumination intensity, e.g. tree regions. Sky pixels in the image show bimodal distribution due to the presence of white textured cloud in a blue sky. At first, the soft clustering is performed on the image and the Cluster Probability Maps are generated. For this purpose, the number of clusters in the image is estimated using the KLD-based technique. Fig. 5 shows the change in KLD when the number of clusters (K) is increased. As K is increased from 1 to 2, a sharp decrease in KLD can be noticed. After $K = 5$, the KLD almost stabilizes indicating the non-significance of further increments in K. Hence, we have chosen $K = 5$ as the number of clusters in the original image. Intuitively, one can observe five broad categories in the input image, i.e. *sky*, *water*, *red wall*, *floor/skin*, and *tree/grass*.

Once the number of clusters has been determined, unsupervised learning of the mixture model generates the Cluster Probability Maps, i.e. $P(j|x_t^I)$. These maps for five clusters in the original image are given in Fig. 4(b)–(f)), where a brighter pixel indicates a higher probability (the brightness within each map has been scaled non-linearly to reduce the dynamic range for display purposes). In a Cluster Probability Map, a brighter pixel indicates a higher probability of association of that pixel with the given cluster. It is clear from Fig. 4 that different clusters have captured the similarities in the image pixels. In order of increasing j, the maps broadly represent grass, red wall, sky, floor, and water. The semantic classes which are represented in relatively small regions have been merged with other clusters, e.g. skin regions were merged with red wall or floor.

There are some intuitively obvious incorrect associations in the cluster maps, e.g. for $j = 1$, parts of sky, water, and swimming costume have been clustered along with the grass/tree class. However, it should be emphasized that at this stage the algorithm has no notion of a semantic class. The results of probabilistic clustering are purely based on the *similarity* within the newly observed data. The main aim of the soft clustering was not to obtain perfect clustering, but to obtain a probabilistic estimate of cohesiveness in various pixels, which could later be used for constraining the results of the class generative models. The errors in clustering do not have limiting influence on the final Constrained Class Density Maps: this will be shown in the later part of this section.

The Class Density Maps displaying $p(x_k^\omega|\omega)$ for every pixel in the input image (Fig. 4(a)) corresponding to the five classes are given in Fig. 6(a)–(e), where again a brighter pixel indicates a higher density. It can be seen that the full generative model for each class has incorrectly predicted significant density for the regions that do not belong to that class. For example, for the class sky, parts of floor and skin have been assigned significant probability of being sky.

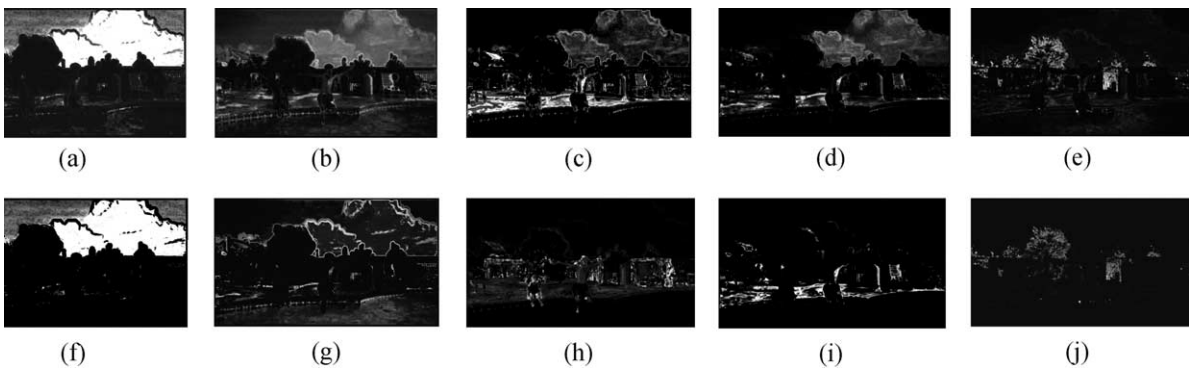

Fig. 6. For the input image shown in Fig. 4(a), (a)–(e) Original Class Density Maps corresponding to the class sky, water, skin, sand/soil, and tree/grass. (f)–(j) Corresponding Constrained Class Density Maps.

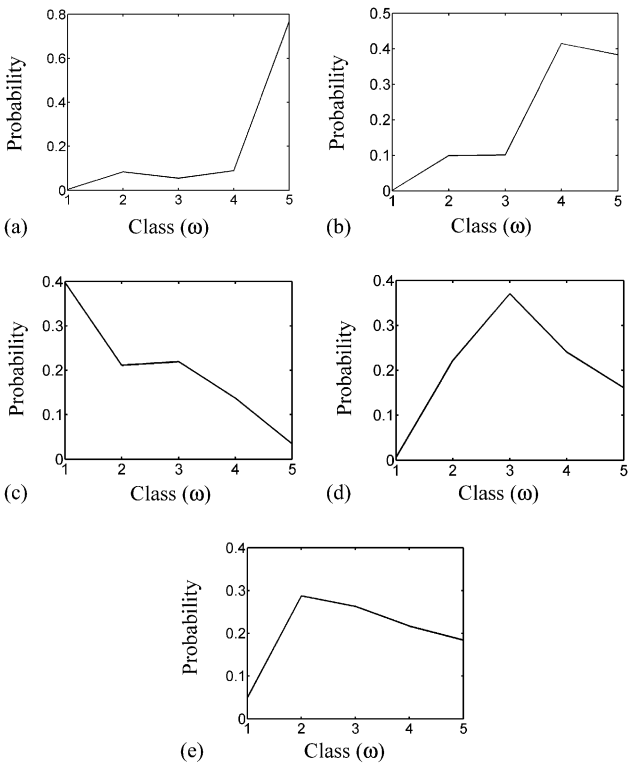

Fig. 7. Posterior distribution $P(\omega|j)$ over classes (ω) given the different Cluster Probability Maps j from Fig. 4. (a) $j = 1$, (b) $j = 2$, (c) $j = 3$, (d) $j = 4$, (e) $j = 5$. The classes are numbered as: (1) sky, (2) water, (3) skin, (4) sand/soil, and (5) grass/tree.

These regions share features close to those of class sky in some arbitrary illumination or physical conditions like scale, weather etc.

To obtain the Constrained Class Density Map for each class, first the Cluster Probability Maps consistent with that class are found. For this purpose, posterior distributions over the classes given each Cluster Probability Map are obtained using Eq. (7). Fig. 7 displays the posterior distributions for Cluster Maps $j = 1,\ldots,5$ displayed in Fig. 4(b)–(f). Each plot is a discrete valued graph where horizontal axis displays the five different classes, i.e. (1) sky, (2) water, (3) skin, (4) sand/soil, and (5) grass/tree. For each cluster map, the maximally consistent class is obtained using the MAP estimate of the posterior. It is clear from the plots that the cluster map for $j = 3$ has much higher probability of being from the class sky ($\omega = 1$) than the other four classes. Also, this is the only cluster that is consistent (in the sense of MAP) with the class sky. This correspondence between the chosen class and the Cluster Probability Map is supported intuitively from Fig. 4(d). All other cluster maps have natural semantics favoring other classes in the recognition vocabulary. Thus, the set of consistent cluster maps, K_ω for the class sky has cardinality one. Similarly, the consistent Cluster Probability Maps for other classes are chosen.

The Cluster Probability Map corresponding to $j = 3$ (Fig. 4(d)) was used to constrain the original Class Density Map for sky (Fig. 6(a)) using Eq. (8). The Constrained Class Density Map containing $p_{\text{cons}}(x_t^j|\omega)$ for each pixel in the input image corresponding to the class sky is given in Fig. 6(f). The densities of false sky regions, e.g. parts of wet floor, skin, water etc. have been significantly reduced. If more than one cluster had been consistent with the class sky, the constrained map could be easily computed using all the consistent cluster maps in Eq. (8). Similar constrained maps were also obtained for the remaining classes in the recognition vocabulary and are displayed in Fig. 6(g)–(j).

To display the results of the probabilistic classification of the image regions in Fig. 4(a) using the observation-constrained generative approach, we have used the MAP paradigm. Each image pixel is classified as belonging to class ω_c, which has highest posterior $P(\omega_c|x_t^j)$. Posteriors are computed using the constrained class conditional densities given in Eq. (8) and assuming equal priors for all the classes, which is equivalent to computing normalized likelihoods. The resulting classification is given as binary images in Fig. 8(b)–(f) where white pixels represent the pixels belonging to the given class. The overall good discriminative classification results are indicative of good Constrained Class Density Maps. Sky, water, and grass/tree regions have been classified almost perfectly except on the edges, because

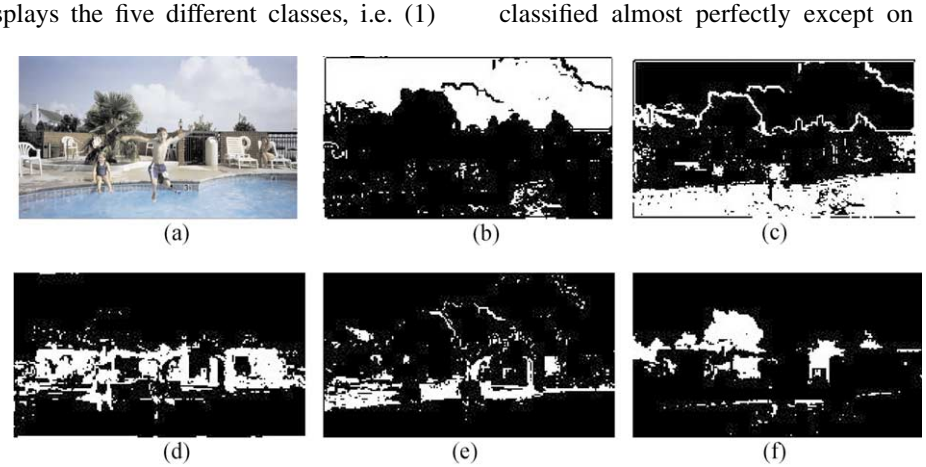

Fig. 8. Discriminative classification of image regions. The maps are binary maps where a white pixel represents the presence of the corresponding class. (a) Input color image. (b)–(f) Class maps for sky, water, skin, sand, and grass/tree, respectively.

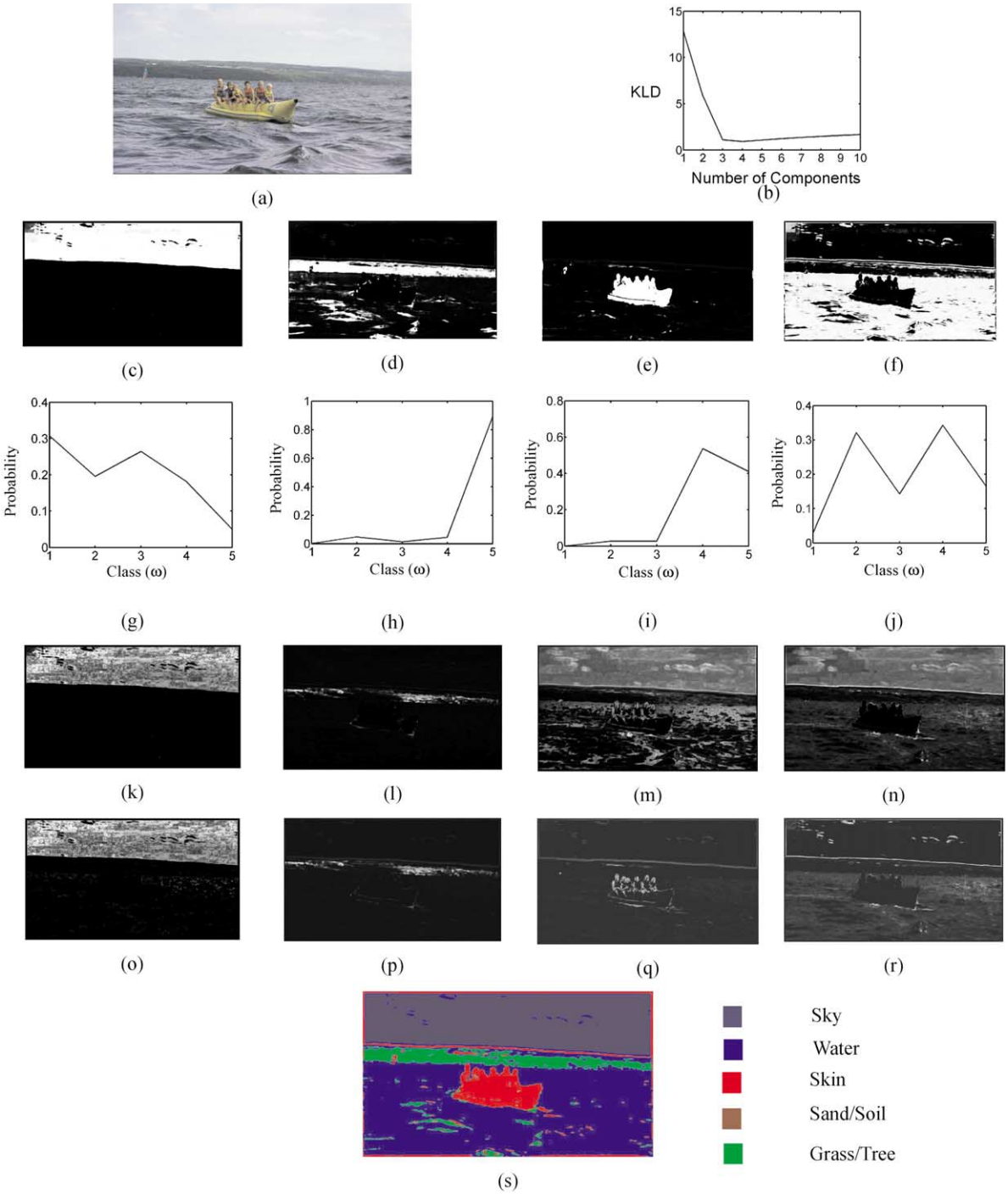

Fig. 9. (a) Input color image. (b) KLD-based model selection favors four components. (c)–(f) Cluster Probability Maps for each of the four clusters. (g)–(j) Corresponding Class posterior distributions. (k)–(n) Class Density Maps for sky, grass/tree, skin, and water, respectively. (o)–(r) Corresponding Constrained Density Maps. (s) MAP Discriminative result showing the semantic classification of various image regions.

the texture measure is not a point estimate, as emphasized above. Use of improved combinations of the eigenvalues of the second moment matrix to avoid this problem has been left as a future work. In the training data, the features of the sky and water regions are distributed close to each other and classification just based on the training data would be quite misleading. Reasonably good classification results for these regions were obtained using our approach (Fig. 8(b) and (c)), because the statistics of these two regions in the test image could reinforce their respective class associations, generating good Constrained Density Maps.

It can be noted that the above MAP-based classification scheme assigns each pixel in the test image to one of the classes in the recognition vocabulary. But some of the image pixels may not belong to any of the classes in the vocabulary, e.g. in the given test image, pixels pertaining

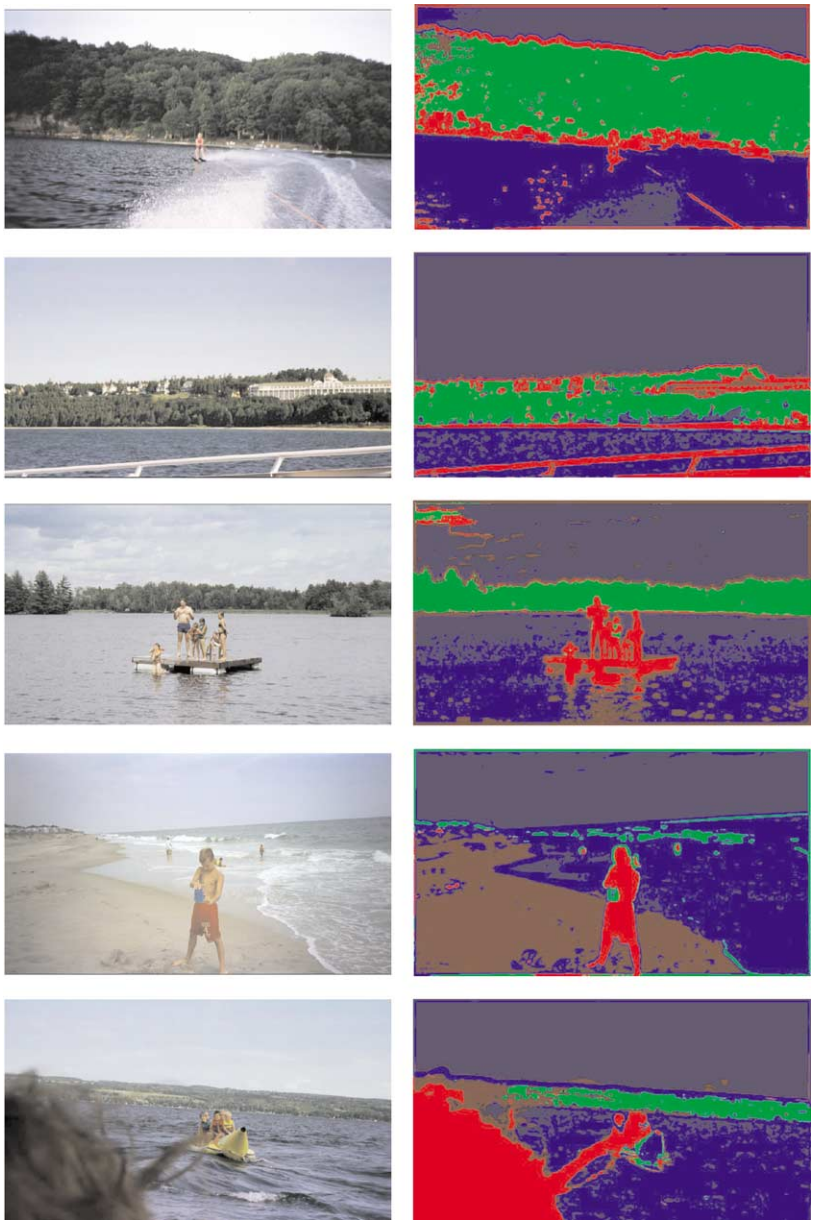

Fig. 10. MAP-based discriminative classification results on images from the test set. Left column: input color images. Right column: image regions classified in one of the semantic classes in the recognition vocabulary.

to the red wall and associated steel gate, chairs etc. In such cases, the MAP scheme assigns those pixels to the best possible classes in the vocabulary. Other minor artifacts in the classified regions are due to the above reason. In the future, we plan to expand the recognition vocabulary to contain more classes. In Fig. 8(e), the concrete floor has been classified as 'sand' because there is no perceptual difference between the two. Skin and sand regions have been partially misclassified due to extreme overlap in the low-level features. Thus, if the statistics of the newly observed image also supports the hypothesis obtained from the Class Density Maps, there is little one can do except either enhancing the feature set, or using some kind of high-level contextual cues. This example motivates the use of scene context to discriminate between regions that are almost impossible to disambiguate using just low level features. Since there was no exclusive class for the 'red wall' in the vocabulary, it has been classified in the semantically nearest class 'skin'.

However, it should be noted that MAP-based classification was used purely for evaluation purposes. The main result of the proposed generative approach is the Constrained Class Density Map for each class, shown in Fig. 6(f)–(j), which is to be further used in the iterative scene context generation scheme.

All intermediate results of the proposed technique are shown on another image given in Fig. 9(a). The KLD-based model selection (Fig. 9(b)) favors four components for soft

clustering. The Cluster Probability Maps for each of the four clusters are displayed in Fig. 9(c)–(f). Fig. 9(g)–(j) show the posterior distribution for each of the Cluster Probability Maps given in the above row. It can be noted that the Bayesian scheme assigns each cluster to its semantically most coherent class. The original Class Density Maps corresponding to classes sky, grass/tree, skin, and water are shown in Fig. 9(k)–(n), respectively, and the corresponding Constrained Density Maps are shown in Fig. 9(o)–(r). The spurious effects of using full generative model are more visible in Fig. 9(m) and (n). In Fig. 9(m), for the class skin, several water and sky regions have been assigned high density. Similarly, in Fig. 9(n), sky regions have been given high density for the class water. Their Constrained Class Density Maps in Fig. 9(q) and (r), respectively, show considerable improvement over the original Class Density Maps, as the false regions have shrunk significantly. The discriminative map of the input image showing the semantic classification of various image regions is given in Fig. 9(s). The color code used for representing different classes has been displayed next to the classified image.

Some more discriminative classification results on a few images from the test set are shown in Fig. 10. Left column shows the original input images while the right column shows the corresponding images with various regions classified as one of the five classes in the recognition vocabulary. It is worth noting once again that any area in an image that does not belong to any of the five classes is assigned to the semantically closest class among five since we have not used a 'reject' class in the present formulation.

6. Conclusions

We have proposed and successfully demonstrated the use of an observation-constrained generative approach for the probabilistic classification of image regions. A probabilistic approach towards clustering and classification leads to a useful technique, which is capable of refining the classification results obtained using the generative models. The proposed scheme is robust to errors in clustering. A KLD-based fast component selection procedure has been proposed for natural scene images. In the future, we intend to work towards evolving efficient schemes to generate distribution over scene hypothesis using the Constrained Class Density Maps. The proposed probabilistic classification forms the first step of a promising feedback framework to iteratively refine the scene context as well as the region hypothesis.

Acknowledgements

We would like to thank Amit Singhal and Zhao Hui Sun from Eastman Kodak Company, Rochester, and Daniel Huber and Goksel Dedeoglu from Carnegie Mellon University, Pittsburgh for their useful suggestions.

References

[1] C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, New York, 1995.
[2] N.W. Campbell, W.P.J. Mackeown, B.T. Thomas, T. Troscianko, The automatic classification of outdoor images, International Conference on Engineering Applications of Neural Networks (1996) 339–342.
[3] C. Carson, S. Belongie, H. Grennspan, J. Malik, Region-based image querying, CVPR'97, Workshop on Content-based Access of Image and Video Libraries, 1997.
[4] A. Dempster, N. Laird, D. Rubin, Maximum likelihood from incomplete data via the EM algorithm, Journal of the Royal Statistical Society, Series B 39 (1977) 1–38.
[5] C. Farley, A.E. Raftery, How many clusters? Which clustering method? Answers via model-based cluster analysis, Technical Report, No. 329, Department of Statistics, University of Washington, 1998.
[6] M.H. Hansen, B. Yu, Model selection and the principle of minimum description length, JASA 96 (1998) 746–774.
[7] D. Heckerman, D. Chickering, A comparison of scientific and engineering criteria for Bayesian model selection, Technical Report MSR-TR-96-12, Microsoft Research, 1996.
[8] S. Konishi, A.L. Yuille, Statistical cues for domain specific image segmentation with performance analysis, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition 1 (2000) 125–132.
[9] S. Kumar, A.C. Loui, M. Hebert, Probabilistic classification of image regions using an observation-constrained generative approach, The ECCV Workshop on Generative Model Based Vision, Copenhagen (2002) 91–99.
[10] B.S. Manjunath, W.Y. Ma, Texture features for browsing and retrieval of image data, IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (1996) 837–842.
[11] J. Mao, A.K. Jain, Texture classification and segmentation using multiresolution simultaneous autoregressive models, Pattern Recognition 25 (1992) 173–188.
[12] T. Rikert, M. Jones, P. Viola, A cluster-based statistical model for object detection, Proceedings of the IEEE International Conference on Computer Vision (1999) 1046–1053.
[13] J. Rissanen, Modeling by shortest data description, Automatica 14 (1978) 465–471.
[14] H. Schneiderman, A statistical approach to 3D object detection applied to faces and cars, PhD Thesis, Carnegie Mellon University, 2000.
[15] H. Schneiderman, T. Kanade, A statistical method for 3D object detection applied to faces and cars, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2000) 746–751.
[16] N. Sochen, R. Kimmel, R. Malladi, A general framework for low level vision, IEEE Transactions on Image Processing 7 (1998) 310–318.
[17] M. Storring, H. Andersen, E. Granum, Skin colour detection under changing lighting condition, in: H. Araujo, J. Dias (Eds.), Seventh Symposium on Intelligent Robotics Systems, 1999, pp. 187–195.
[18] T.M. Strat, Natural Object Recognition, Springer, Berlin, 1992.
[19] M.A. Tanner, Tools for Statistical Inference, Third ed., Springer, Berlin, 1996.
[20] A. Torralba, P. Sinha, Statistical context priming for object detection, Proceedings of the International Conference on Computer Vision, ICCV'01 (2001).
[21] A. Vailaya, Semantic classification in image databases, PhD thesis, Michigan State University, 2000.