# Mixture Models and the Segmentation of Multimodal Textures

Roberto Manduchi
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109
manduchi@jpl.nasa.gov

**Abstract**

*A problem with using mixture-of-Gaussian models for unsupervised texture segmentation is that a "multimodal" texture (such as can often be encountered in natural images) cannot be well represented by a single Gaussian cluster. We propose a divide-and-conquer method that groups together Gaussian clusters (estimated via Expectation Maximization) into homogeneous texture classes. This method allows to successfully segment even rather complex textures, as demonstrated by experimental tests on natural images.*

## 1 Introduction

Algorithms for image segmentation can be roughly divided into two categories: those that use statistical models for describing the behavior of visual features, and those that only require some measure of "similarity" between features [9][11]. Recent graph cutting techniques [10] are an instance of the latter. These algorithms partition a graph describing the interrelation between image pixels by minimizing a suitable functional of the related "affinity matrix". The entry of the affinity matrix at position $(m, n)$ is a combination of the difference in appearance between the $m$-th and the $n$-th pixels and of their distance in the image plane. Thus, these approaches seamlessly integrate spatial and appearance coherence in a elegant and general framework. Unfortunately, handling relational graphs built from all the pixels in an image is very challenging in terms of memory and computational power even for moderate size images (e.g., $200 \times 200$ pixels), therefore heavy image subsampling is in order.

Statistical techniques stand on the other side of the spectrum[1]. They assume that image features obey a probabilistic model, and approach segmentation as a general clustering problem, drawing on classical results of pattern recognition. Bayesian approaches maximize the probability that a point $x$ characterized by the image feature $\mathbf{c}(x)$ belongs to the cluster $j$, i.e., $P(j|\mathbf{c}(x))$. Interdependence among nearby pixels is taken into account, for example, by means of Markov Random Field models. An advantage of statistical techniques is that the final segmentation is "soft", being expressed in terms of posterior probabilities. This facilitates integration with other visual features and/or with *a priori*, "supervised" information [6].

This paper proposes a simple statistical parametric technique for texture segmentation. The statistical description of textures has received much attention in recent years. Texture features $\mathbf{c}(x)$ are typically extracted from the output of a set of scaled/oriented filters, which are supposed to capture local salient information in the neighborhood of each image point. Several non-parametric techniques can be found in the literature for estimating the marginal densities $p(\mathbf{c}(x))$ in the case of homogeneous textures [8][2][5]. Parametric mixture models are the framework of choice for segmentation. These models assume that a feature $\mathbf{c}$ is generated by one of $N$ possible processes ("components"). The probability density function of $\mathbf{c}$ can thus be expressed by a *mixture distribution*

$$p(\mathbf{c}) = \sum_{j=1}^{N} P(j)p(\mathbf{c}|j) \qquad (1)$$

where $p(\mathbf{c}|j)$ is the conditional likelihood of the feature $\mathbf{c}$ generated by the component $j$ and $P(j)$ is the prior probability of the component $j$ (called *mixing parameter*). The posterior probabilities $P(j|\mathbf{c}(x))$ are derived straightforwardly from the mixture model using Bayes' rule, and are used for the final segmentation. Note that each component of the model corresponds to exactly one image segment[2].

[1] Statistical and graph-theoretic techniques are not necessarily disjoint. For example, one may use knowledge about class statistics to design more effective distance metrics for use in the affinity matrix.

[2] In the context of this paper, image segments are not neces-

Mixture models owe their popularity in part to the existence of an efficient technique (the Expectation-Maximization algorithm) for the maximum likelihood parameters estimation [7]. In its simplest formulation, the EM algorithm relies on two hypotheses: 1) a suitable model for the conditional likelihoods is known, and 2) the observed samples are statistically independent. Neither of these hypotheses is verified in typical textures. The problem of sample independence[3] is fairly well understood; extensions of the EM algorithm that use MRF modeling of the class label distribution have been proposed [15][14]. In this paper we tackle the first problem, the determination of a statistical model for feature generation within each texture class, originating our argument from the observation that simple Gaussian models are inadequate to describe "multimodal" textures, such as can be often encountered in practice.

Mixture of Gaussians are the most common instance of mixture models, one reason being that Gaussian conditional likelihoods allow for the E- and M-steps of the EM algorithm to be solved in closed form [7]. Each Gaussian cluster represents a "mode" of the mixture distribution. Malik *et al.* [3] call the cluster centers "textons" and use them for compact texture representation (via vector quantization). Our main point here is that it is often necessary to use more than one Gaussian cluster to represent an homogeneous texture feature distribution. For example, consider the image of Figure 1(a), composed by the juxtaposition[4] of a Brodatz texture and of the same texture rotated by 45°. In this simple experiment, we used a bank of Gabor filters at four orientations to extract texture features. It is easy to convince oneself that the feature distribution in each texture patch is bimodal, due to the two presence of two principal orientations. Therefore, a 2-components mixture-of-Gaussians model fails to represent the whole scene giving, for example, the incorrect segmentation of Figure 1(b) (note that, due to the symmetry of the distributions in orientation space, there are other possible stationary points the algorithm may converge to, including the "correct" one).

To deal with multimodal textures like the one in Figure 1(a), we propose an unsupervised divide-and-conquer strategy. First, extract a suitable number of mixture components using the EM algorithm; then, group together those clusters which are likely to belong to the same texture. For example, in Figure 1(c) we show the EM segmentation using six Gaussian components. By suitably grouping these components into two sets, we obtain the correct segmentation of Figure 1(d). In this case, each texture is described by a mixture of three Gaussians.

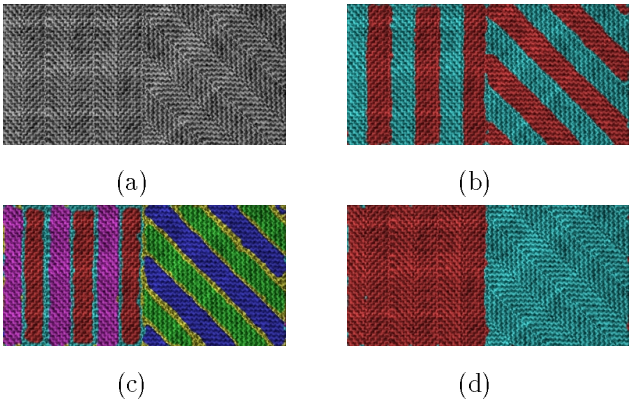



Figure 1: (a): Original image. (b) Incorrect segmentation using a mixture of two Gaussians. (c) Segmentation with six Gaussian clusters. (d) Segmentation into two texture classes, each one of which is represented by a mixture of three Gaussian clusters.

How can we estimate the correct assignments cluster–texture? Our algorithm determines a cost function of cluster grouping that keeps *spatial coherence* into account. A simple, non-iterative technique allows us to determine the cluster groupings that minimize such function, and the final Bayesian assignment is performed based on the new combined posterior distribution. Results on natural textured scene show the effectiveness of the proposed method.

## 2 Multimodal texture segmentation

### 2.1 Problem statement

Our strategy for segmenting multimodal textures is based on "grouping together" some of the components of a given mixture model. More precisely, consider a partition $\{I_1, \ldots, I_{\hat{N}}\}$ of the discrete set $I = \{1, \ldots, N\}$. Let

$$\hat{p}(\mathbf{c}|k) = \frac{1}{\sum_{i \in I_k} P(i)} \sum_{i \in I_k} P(i) p(\mathbf{c}|i), \qquad (2)$$

$$\hat{P}(k) = \sum_{i \in I_k} P(i)$$

sarily (and usually are not) connected.

[3] There are actually two kinds of dependency, one concerning the underlying class label distribution, and the other concerning the feature distribution within each class[15].

[4] Of course, one may argue that four homogeneous textures can be seen in the scene, depending on the scale of the *observation window* used.

We can rewrite (1) as

$$p(\mathbf{c}) = \sum_{k=1}^{\hat{N}} \hat{P}(k)\hat{p}(\mathbf{c}|k) \tag{3}$$

The index $k$ in (3) labels the different *texture classes* in the scene; the index $i$ in (2) enumerates the clusters within each texture class. A feature $\mathbf{c}$ is assigned to the texture $k$ that maximizes $\hat{P}(k)\hat{p}(\mathbf{c}|k) = \sum_{i \in I_k} P(i)p(\mathbf{c}|i)$. It is important to note that, in general, the set of pixels that are assigned to a class $k$ by means of (3) *is not* the union of the sets of pixels assigned to the classes $\{i \in I_k\}$: grouping together clusters determines new Bayesian assignments that are not trivially derived from the original ones.

As anticipated in the Introduction, we will determine the groupings in (3) by exploiting the spatial coherence of the class assignment function. More precisely, we observe that the posterior probabilities $P_k(i_1|\mathbf{c}(x))$ and $P_k(i_2|\mathbf{c}(x))$ for two clusters $i_1$ and $i_2$ belonging to the same texture $k$ are typically *spatially correlated*. They can assume high values ($\leq 1$) only in image areas corresponding to the same texture; for homogeneous textures, it is reasonable to assume that, within a "window of observability" of suitable scale, we will normally find both pixels assigned to cluster $i_1$ and pixels assigned to cluster $i_2$. This notion is exploited in the context of the recently proposed *maximum descriptiveness* criterion [6] for grouping "redundant" clusters in a mixture model. We first discuss the maximum descriptiveness criterion, referring the reader to [6] for more details. We then show its application in the context of this work.

## 2.2 Model descriptiveness

Consider a mixture model with density $p(\mathbf{c})$ expressed by (1). The *descriptiveness* $D$ of the model [6] is defined by

$$D = \sum_{j=1}^{N} D_j \ , \ D_j = \int p(\mathbf{c}|j)P(j|\mathbf{c})d\mathbf{c} \tag{4}$$

where the posterior probabilities $P(j|\mathbf{c})$ are derived from (1) using Bayes' rule: $P(j|\mathbf{c}) = P(j)p(\mathbf{c}|j)/p(\mathbf{c})$. Let us examine each term of the sum in (4). The $j$-th cluster "describes" each feature $\mathbf{c}$ by means of the conditional likelihood $p(\mathbf{c}|j)$. The posterior probability $P(j|\mathbf{c})$ specifies in a "soft" fashion which features are actually assigned by the model to the $j$-th cluster. Thus, the integrals in the sum determine how well each cluster describes the features that are assigned to it. It is easily seen that models with "hard" assignment rules have the highest descriptiveness (which can only be less than or equal to $N$). Models with highly overlapping densities $p(\mathbf{c}|j)$ have smaller descriptiveness for the same number of classes. The lowest value of the descriptiveness ($D$=1) is achieved when all of the conditional likelihoods are identical.

A very useful property of the descriptiveness is that it can be easily estimated: a simple application of Bayes' rule proves the following identity:

$$D_j = \frac{E\left[P(j|\mathbf{c})^2\right]}{P(j)} \tag{5}$$

where $E[\cdot]$ is the expectation computed with respect to the density $p(\mathbf{c})$. The numerator of each term (5) can thus be estimated by simply averaging $P(j|\mathbf{c}(x))^2$ over the image. The denumerator is estimated by averaging $P(j|\mathbf{c}(x))$ over the image.

For our purposes, the descriptiveness of a model is not used by itself; it is its *variation* when two or more clusters are grouped together which is of interest to us. Suppose that a new model is generated by grouping two clusters (say, clusters $i$ and $j$) into a new "super-cluster" $i \cup j$ according to the following rules:

$$\begin{aligned} &P(i \cup j) = P(i) + P(j) \\ &P(i \cup j\ |\mathbf{c}) = P(i|\mathbf{c}) + P(j|\mathbf{c}) \\ &p(\mathbf{c}|i \cup j) = p(\mathbf{c}|i)\,\tfrac{P(i)}{P(i)+P(j)} + p(\mathbf{c}|j)\,\tfrac{P(j)}{P(i)+P(j)} \end{aligned} \tag{6}$$

Note that the conditional likelihood defined in the last row of (6) is such that the density $p(\mathbf{c})$ defined by the model does not change: our grouping operation (which is equivalent to (3)) is purely formal. However, the model descriptiveness $D$ will change (in general) as an effect of cluster grouping. Indeed, it can be shown that the model descriptiveness $D$ may only decrease or remain unchanged when two or more clusters are grouped together. The descriptiveness decreases the most when clusters with well-separated conditional distributions are grouped together, while highly overlapping distributions can be grouped with little descriptiveness loss.

To decide which clusters should be grouped together into a super-cluster as by (6) (or (3)) in order to reduce the number of texture classes, we may look at the corresponding model descriptiveness decrement $\Delta D$. The intuition is that clusters which are highly overlapping in feature space (small $\Delta D$) are the "safest" choice for grouping. Thus, we should choose the cluster grouping scheme that yields the smallest value of $\Delta D$. We will call this strategy the *maximum descriptiveness* criterion. A fast sub-optimal technique for minimizing the descriptiveness loss over cluster groupings has been proposed in [6]. This al-

gorithm greedily groups two clusters at a time, each time minimizing $\Delta D$.

There is an interesting interpretation of the descriptiveness which will be useful for our work. Suppose we are grouping together two clusters of indices $i$ and $j$. Then, from (5) and (6) we have that

$$\Delta D = D_i \frac{P(j)}{P(i)+P(j)} + D_j \frac{P(i)}{P(i)+P(j)} - \frac{2E\left[P(i|\mathbf{c})P(j|\mathbf{c})\right]}{P(i)+P(j)} \tag{7}$$

The last term in the sum above is the cross-correlation between the two distributions, normalized with respect to the average of the corresponding priors. Thus, for given cluster descriptiveness $D_i$,$D_j$ and prior probabilities $P(i)$,$P(j)$, the two clusters will determine a large $\Delta D$ when grouped together if the two corresponding distributions are uncorrelated. Since these distributions are actually a function of the spatial position $x$ of the features $\mathbf{c}(x)$, we may use the signal processing definition of cross-correlation as a function of the displacement $X$:

$$C_{ij}(X) = E\left[P(i|\mathbf{c}(x))P(j|\mathbf{c}(x+X))\right] \tag{8}$$

and rewrite the last term of (7) as $-\frac{2C_{ij}(0)}{P(i)+P(j)}$.

### 2.2.1 Comparison with a criterion based on mutual information

Equation (4) may be rewritten as follows:

$$D = \sum_{j=1}^{N} \int \frac{p(\mathbf{c},j)^2}{p(\mathbf{c})P(j)} d\mathbf{c} = E\left[\frac{p(\mathbf{c},j)}{p(\mathbf{c})P(j)}\right] \tag{9}$$

where now the expectation is computed over the joint density $p(\mathbf{c},j)$. Equation (9) suggests another criterion for cluster grouping, based on the maximization of the following functional:

$$\hat{D} = E\left[\log \frac{p(\mathbf{c},j)}{p(\mathbf{c})P(j)}\right] \tag{10}$$

Here $\hat{D}$ is the Kullback-Leibler (K-L) divergence between the joint density $p(\mathbf{c},j)$ and the product of the marginal density $p(\mathbf{c})$ and of the mass distribution $P(j)$, which can be considered a generalized form of mutual information. Thus, $\hat{D}$ represents the expected dependence between the observed data and the underlying generative model. Being a K-L divergence, $\hat{D}$ is always non-negative; it is shown in the Appendix that $\hat{D}$ never increases when two clusters are grouped together. Hence, we may choose to group together those clusters which yield the smallest decrement of $\hat{D}$. An intuitive justification of such a criterion is provided by the following observation. Let us rewrite equation (10) as

$$\hat{D} = H(P(j)) - E[H(P(j|\mathbf{c}))] \tag{11}$$

where now the expectation is computed over the density $p(\mathbf{c})$, and $H(\cdot)$ is the entropy operator. Maximizing $\hat{D}$ corresponds to minimizing $E[H(P(j|\mathbf{c}))]$ (which represents the mean "softness" of posterior assignment), while at the same time maximizing the entropy of the distribution of the priors. Thus, our criterion favors models characterized by "hard" assignments and homogeneous prior distribution.

Experimental tests have shown that the results using this mutual information criterion are often less convincing from a "perceptual" point of view than those obtained with the descriptiveness criterion described in Section 2.2. Indeed, depending on the "temperature" of the original clustering algorithm, the entropy of the prior distribution may dominate the sum in (11), in which case this criterion simply tries to make the distribution of the priors as uniform as possible.

## 2.3 Cluster-texture assignment

Our goal is to find a criterion that tells us when two clusters belong to the same texture, so that we can group them together as in (3). The maximum descriptiveness criterion described in the previous section is not helpful if applied directly to the posterior probabilities $P(j|\mathbf{c})$: two different clusters belonging to the same texture class may be well separated in feature space, as in the case of Figure 1. Instead, we propose to apply the same criterion to the *spatially filtered* versions of the posterior probabilities.

The intuition behind this strategy is the following. As observed earlier, we expect that the posterior distributions for different clusters belonging to the same texture should be spatially correlated. By spatially smoothing these distributions, we expect that a point that was assigned with high probability to just one cluster will now be softly assigned to a number of clusters belonging to the same texture. Cluster grouping is then determined by applying the maximum descriptiveness algorithm to the smoothed posterior distributions. Note that this procedure is used only to find the correspondence cluster-texture: the final segmentation is operated using the model (3), i.e., based on non-filtered distributions (see Figure 2).

We now give a more thorough justification of our method. Let $g(x)$ be an isotropic Gaussian kernel of suitable scale $\sigma$, normalized to unit area. Let $\bar{P}(j|x) = \int P(j|\mathbf{c}(t))g(x-t)\,dt$ be the filtered version of the posterior distribution $P(j|\mathbf{c}(x))$ (we dropped

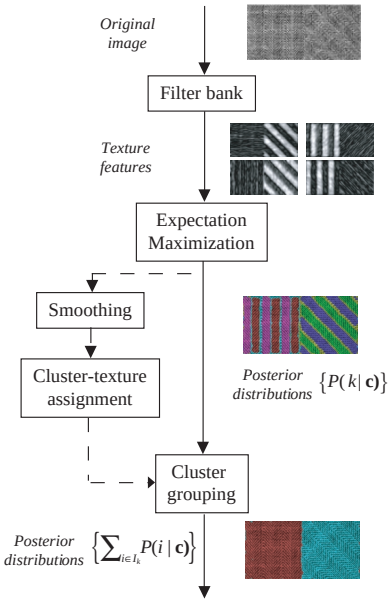



Figure 2: Scheme of our strategy for cluster grouping. The images in the scheme refer to Figure 1.

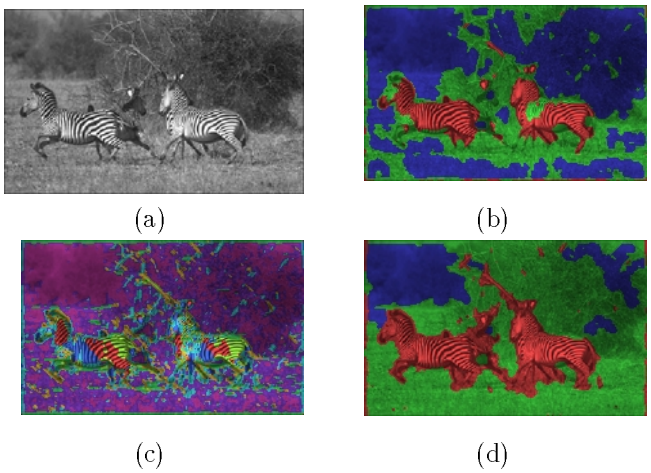



Figure 3: (a): "Zebras" image. (b): Segmentation using three clusters. (c) Segmentation using eight clusters. (d): Segmentation into thee texture classes by cluster grouping.

the dependency on $\mathbf{c}$ because now $\bar{P}(j|x)$ is a function of a whole ensemble of features in a neighborhood of $x$). Since $g(x)$ has unit area, it is easily proved that $\bar{P}(j|x)$ for $1 \leq j \leq N$ is still a mass distribution for each $x$. Furthermore, $\bar{P}(j) = E\left[\bar{P}(j|x)\right] = P(j)$.

Now, consider the cross-correlation function

$$\bar{C}_{ij}(X) = E\left[\bar{P}(i|x)\bar{P}(j|(x+X)\right] \tag{12}$$

It is easy to prove that

$$\bar{C}_{ij}(X) = \int C_{ij}(x)\bar{g}(X-x)dx \tag{13}$$

where $C_{ij}(x)$ was defined in (8) and $\bar{g}(x) = \int g(t)g(t-x)dx$ (note that $\bar{g}(x)$ is a unit-area Gaussian kernel with standard deviation $\bar{\sigma} = \sigma/2$). Therefore, $\bar{C}_{ij}(0)$ is a weighted average of the values of the cross–correlation between the $i$-th and the $j$-th posterior distributions within a neighborhood of radius proportional to $\sigma/2$ (which we will call the *observation window*).

Now consider the decrement of descriptiveness $\Delta\bar{D}$ consequent to grouping two clusters $i$ and $j$ after spatial smoothing:

$$\Delta\bar{D} = \bar{D}_i\frac{P(j)}{P(i)+P(j)} + \bar{D}_j\frac{P(i)}{P(i)+P(j)} - \frac{2\bar{C}_{ij}(0)}{P(i)+P(j)} \tag{14}$$

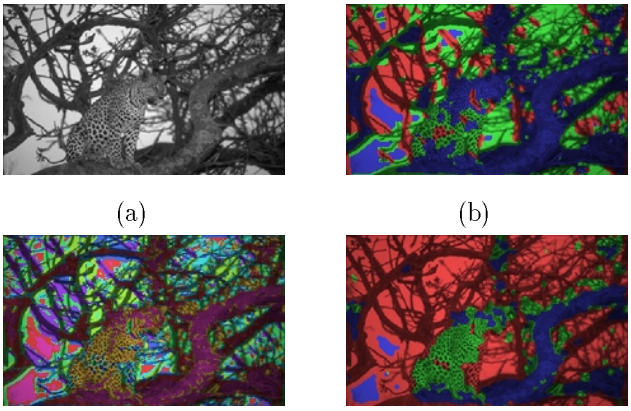
From (14) we maintain that, for given $P(i|\mathbf{c}(x))$, $P(j|\mathbf{c}(x))$ and priors $P(i)$,$P(j)$, the value $\Delta\bar{D}$ depends on the degree of local spatial correlation between the two posterior distributions. Thus, the maximum descriptiveness algorithm applied to the smoothed distributions will correctly determine which cluster posterior distributions best correlate, and will group them together into common texture classes. An instance of application of the proposed algorithm is shown in Figure 1(d); more examples are described in the next section.

### 2.4 Experiments

We present here the segmentation results using our method on three real-world textured images: the "Zebras" image (Figure 3(a)), the "Cheeta" image (Figure 4(a)) and the "Pebbles" image (Figure 5(a)).

The vectors formed by the magnitude of the output of complex Gabor filters at three scales and four orientations have been used as texture features. The Gaussian filter used to smooth the posterior distributions for cluster-texture assignment had standard deviation $\sigma = 40$, seven times larger than the standard deviation of the Gaussian envelope of the largest Gabor filter used. In both cases, we started with a mixture model composed by eight Gaussian clusters. This number has been chosen arbitrarily; validation mechanisms for selecting a "suitable" number of clusters can be found in the literature [12].

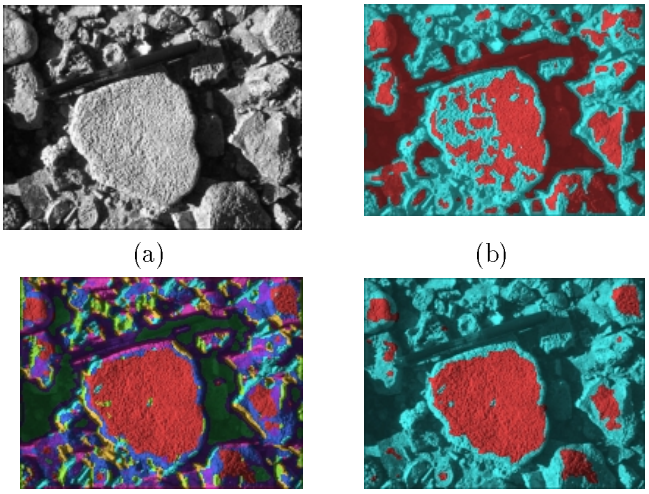
The EM algorithm was bootstraped with initial parameter values determined by a previous K-means clustering, and was stopped after twenty iterations.



(a) (b)

(c) (d)

Figure 4: (a): "Cheeta" image. (b): Segmentation using three clusters. (c) Segmentation using eight clusters. (d): Segmentation into thee texture classes by cluster grouping.



(a) (b)

(c) (d)

Figure 5: (a): "Pebbles" image. ((b): Segmentation using two clusters. (c) Segmentation using eight clusters. (d): Segmentation into two texture classes by cluster grouping.

In passing, we notice that increasing the number of clusters reduces the risk of missing global minima in the EM iterations. A simple post-processing technique [15] was used to enforce spatial coherence on the resulting multimodal posterior distributions. This algorithm is in essence a "soft" version of Besag's Iterated Conditional Modes [1]; its relation to the mean field theory is discussed in [16].

The segmentations relative to the original clusterings into eight clusters are shown in Figures 3(c), 4(c) and 5(c). After cluster grouping, we obtain the segmentations of Figures 3(d) and 4(d) (three texture classes), and 5(d) (two texture classes). For comparison, the direct EM segmentation into the same number of classes is shown in Figures 3(b), 4(b) and 5(b). In the case of the "Zebras" image, our algorithm succesfully segmented the striped regions corresponding to the zebras (5 clusters), and allocated one texture class (2 clusters) to the grass and the large bush. Direct EM clustering fails to segment the zebras into one class due to the large variance in orientation and scale corresponding to the distincitve stripes. In the case of the "Cheeta" images, we notice that the shapes of the foreground branch and of the cheeta have been identified (although not perfectly). Several small areas around the larger tree branch are misclassified due to their similarity with the polka-dot texture on the cheeta's skin. More remarkably, the cluttered background has been segmented almost completely into just one class, the union of six distinct clusters. In the case of the"Pebbles" image, one texture class (1 cluster) has been allocated to the characteristic surface of some flat rocks in the scene. Note that the "background" texture class contains clusters corresponding to dark and bright areas as well as to edge areas.

In terms of implementation complexity, we observe that the bulk of the computation is due to the EM iterations (for which, however, acceleration method exist [7]). The determination of the cluster-texture assignments takes a proportionally negligible time, using the greedy maximum descriptiveness strategy of [6].

## 3 Conclusions and discussion

We presented a divide-and-conquer strategy for texture segmentation. The behavior of the texture features in the scene is first modeled by a number of Gaussian clusters, estimated via Expectation Maximization. Then, selected cluster sets are grouped together to form texture classes. Spatial correlation of the posterior cluster distributions is at the basis of our cluster grouping criterion. Despite its simplicity, this algorithm can model even complex and multimodal distributions, typical of natural outdoor images.

It is instructive to compare our method with other statistic-based techniques which perform clustering on parameter vectors obtained by local statistical analysis. Indeed, some of the earliest filter–based segmentation algorithms [13][4] estimate the local variance of the analysis filter outputs (by performing squaring followed by spatial smoothing) and use these values for

segmentation. More recent variations compute local histograms of the filter outputs. Such approaches are haunted by the problem of selecting an appropriate scale of the analysis window, be it the standard deviation of the smoother or the size of the region used for computing local histograms. The larger the analysis window, the more accurate the local statistics, but the coarser the resolution of the final segmentation. Our methods works directly on the texture features, not on their local statistics. Of course, we need to select a scale for the "observation window", but this value does not affect the resolution of the final class assignment, which is performed using unsmoothed posterior distributions.

A drawback of our technique is that clusters are grouped by the "hard" scheme of (2). This can cause problems if the same cluster appears in two or more texture classes. A more general grouping solution, which is the object of current research, would define the mixture model (3) with

$$\begin{aligned} \hat{P}(k) &= \textstyle\sum_{j=1}^{N} \alpha_{kj} P(j) \\ \hat{p}(\mathbf{c}|k) &= \frac{\sum_{j=1}^{N} \alpha_{kj} P(j) p(\mathbf{c}|j)}{\sum_{j=1}^{N} \alpha_{kj} P(j)} \end{aligned} \tag{15}$$

where $\alpha_{kj} \geq 0$ and $\sum_{k=1}^{\hat{N}} \alpha_{kj} = 1$. The parameters $\{\alpha_{kj}\}$ should be chosen so as to maximize the resulting model descriptiveness.

## Appendix

We will prove that term $\hat{D}$ in (10) can never increase if two clusters $(i, j)$ are grouped together as in (6). We have that

$$\begin{aligned} &\Delta\hat{D} = \\ &\textstyle\int p(\mathbf{c}) P(i|\mathbf{c}) \log \frac{P(i|\mathbf{c})}{P(i)} d\mathbf{c} + \int p(\mathbf{c}) P(j|\mathbf{c}) \log \frac{P(j|\mathbf{c})}{P(j)} d\mathbf{c} - \\ &\textstyle\int p(\mathbf{c}) \left(P(i|\mathbf{c}) + P(j|\mathbf{c})\right) \log \frac{P(i|\mathbf{c})+P(j|\mathbf{c})}{P(i)+P(j)} d\mathbf{c} = \\ &\textstyle\int p(\mathbf{c}) P(i|\mathbf{c}) \log \frac{P(i|\mathbf{c})(P(i)+P(j))}{P(i)(P(i|\mathbf{c})+P(j|\mathbf{c}))} d\mathbf{c} + \\ &\textstyle\int p(\mathbf{c}) P(j|\mathbf{c}) \log \frac{P(j|\mathbf{c})(P(i)+P(j))}{P(j)(P(i|\mathbf{c})+P(j|\mathbf{c}))} d\mathbf{c} = \\ &\textstyle P(i) \int p(\mathbf{c}|i) \log \frac{p(\mathbf{c}|i)}{p(\mathbf{c}|i \cup j)} d\mathbf{c} + P(j) \int p(\mathbf{c}|j) \log \frac{p(\mathbf{c}|j)}{p(\mathbf{c}|i \cup j)} d\mathbf{c} \end{aligned} \tag{16}$$

Thus, $\Delta\hat{D}$ is a linear combination with non-negative coefficients of two Kullback-Leibler divergences, and therefore it is always non-negative.

## Acknowledgments

The work described was funded by the TMOD Technology Program and performed at the Jet Propulsion Laboratory, California Institute of Technology under contract with the National Aeronautics and Space Administration. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

## References

[1] J. Besag. On the statistical analysis of dirty pictures. *J. R. Statist. Soc. B*, 48(3):259–302, 1986.

[2] J.S. De Bonet and P. Viola. Texture recognition using a non-parametric multi-scale statistical model. *Proc. IEEE CVPR*, 641–647, Santa Barbara, June 1998.

[3] J. Malik, S. Belongie, J. Shi, T. Leung. Textons, contours and regions: cue integration in image segmentation. *Proc. ICCV*, 918–925, Kerkyra, 1999.

[4] J. Malik and P. Perona. Preattentive texture discrimination with early vision mechanisms. *Journ. Optical Society of America*, 7(5):923–932, May 1990.

[5] R. Manduchi, J. Portilla. Independent component analysis of textures. *Proc. ICCV*, 1054–1060, Kerkyra, 1999.

[6] R. Manduchi. Bayesian fusion of texture and color segmentations. *Proc. ICCV*, 956–962, Kerkyra, 1999.

[7] G.J. McLachlan, T. Krishnan. *The EM algorithm and extensions.* John Wiley and Sons, 1997.

[8] K. Popat and R. Picard. Cluster-based probability model and its application to image and texture processing. *IEEE Trans. Image Proc.*, 6(2):268–284, February 1997.

[9] Y. Rubner and C. Tomasi. A metric for distributions with applications to image databases. *Proc. 6th ICCV*, 59–66, Bombay, 1998.

[10] J. Shy and J. Malik. Normalized cuts and image segmentation. *Proc. IEEE CVPR*, 731–737, Santa Barbara, 1997.

[11] J. Shy and J. Malik. Self–inducing relational distance and its application to image segmentation. *Proc. ECCV*, 538–543, 1998.

[12] P. Smyth. Clustering using Monte Carlo cross–validation. *Proc. Int. Conf. on Knowl. Disc. and Data Min.*, 126–133, 1996.

[13] M. Unser and M. Eden. Multiresolution feature extraction and selection for texture segmentation. *IEEE Trans. Patt. Anal. Mach. Intell.*, 2(7), 717–728, 1989.

[14] Y. Weiss and E.H. Adelson. A unified mixture framework for motion segmentation: incorporating spatial coherence and estimating the number of models. *Proc. IEEE CVPR*, 321–326, 1996.

[15] J. Zhang, J.W. Modestino, and D.A. Langan. Maximum-likelihood parameter estimation for unsupervised stochastic model-based image segmentation. *IEEE Trans. Image Proc.*, 3(4), 404–420, July 1994.

[16] J. Zhang and J.W. Modestino. The mean–field theory in EM procedures for Markov random fields. *Proc. IEEE ISIT*, Budapest, 1991.