

# Color Texture Segmentation by Decomposition of Gaussian Mixture Model

Jiří Grim, Petr Somol, Michal Haindl, and Pavel Pudil

Institute of Information Theory and Automation  
Academy of Sciences of the Czech Republic  
P.O. BOX 18, 18208 PRAGUE 8, Czech Republic  
{grim, somol, haindl, pudil}@utia.cas.cz  
<http://ro.utia.cas.cz/mem.html>

**Abstract.** Recently we have proposed Gaussian mixtures as a local statistical model to synthesize artificial textures. We describe the statistical dependence of pixels of a movable window by multivariate Gaussian mixture of product components. The mixture components correspond to different variants of image patches as they appear in the window. In this sense they can be used to identify different segments of the source color texture image. The segmentation can be obtained by means of Bayes formula provided that a proper decomposition of the estimated Gaussian mixture into sub-mixtures is available. In this paper the mixture model is decomposed by maximizing the mean probability of correct classification of pixels into segments in a way taking into account the assumed consistency of final segmentation.

## 1 Introduction

The concept of texture segmentation derives from a simple image decomposition suggested e.g. by different colors or by distinct edges. However, it is uneasy to specify the underlying classification problem when the image segments have to be identified by different textural properties. A unique solution of the texture segmentation problem is hardly achievable because a generally accepted definition of texture is also missing. The available texture based image segmentation methods usually consider some local texture properties to identify either boundaries or regions or to combine both approaches [12]. In view of the underlying high-dimensional problems the texture segmentation methods usually apply different feature extraction techniques or subspace approaches [7], [8], [9]. In most applications like image analysis or image database retrieval the segmentation algorithms should be computationally efficient. However, in some areas like medical imaging the computing time is less relevant [13].

In the present paper we propose a novel segmentation procedure based on a local statistical model of the texture properties. We assume that the image to be segmented is composed of different textures which can be characterized by some local statistical properties and simultaneously that the potential texture segments are sufficiently large and homogeneous to enable their identification.

Recently we have shown in a series of papers [3], [5], [6] that gray-scale textures can be described locally in terms of a joint probability density of gray-levels in a suitably chosen observation window. Unlike other approaches, no feature extraction technique has been applied to pixel variables of the window image patch. Consequently, the dimension of the estimated density may be very high, e.g. of order  $10^2 \div 10^3$ . By estimating the probability density in the form of Gaussian mixture of product components we succeeded to synthesize artificial textures by sequential prediction. A specific advantage of texture synthesis is the possibility to verify the quality of the estimated mixture model by comparing the original- and synthesized image visually [5], [6]. Motivated by successful experiments we have applied the local mixture model to statistical evaluation of color texture images with the aim to emphasize abnormalities or local defects [4].

The present application of the statistical model to color texture segmentation is based on decomposition of the underlying mixture density into sub-mixtures which correspond to different segments of the source texture image. Considering the framework of statistical classification we suggest a simple criterion in terms of probability of correct classification of pixels into the segments. The criterion is maximized by means of an iterative algorithm which is shown to converge monotonically in a finite number of steps.

In the following we first describe the local statistical texture model in the form of a multivariate Gaussian mixture of product components. In Sec. 3 we propose the basic texture segmentation algorithm and in Sec. 4 its topological modification. In Sec. 5 we illustrate the method by numerical examples from the Prague segmentation benchmark [11]. Finally in the Conclusion we summarize the method and discuss different computational aspects.

## 2 Gaussian Mixture Model

A digitized color texture image can be described by a matrix of vector variables where each pixel specifies the three RGB spectral values

$$\mathcal{Z} = [\mathbf{z}_{ij}]_{i=0}^I \text{ }_{j=0}^J, \quad \mathbf{z}_{ij} = (z_{ij1}, z_{ij2}, z_{ij3}) \in \mathcal{R}^3.$$

Here  $i, j$  correspond to row and column indices respectively. We assume that the statistical dependencies between pixels in a suitably chosen observation window do not depend on the window position or, in other words, that the local statistical properties of the texture are shift-invariant. Given a window centered at a position  $(i, j)$ , we denote

$$\mathbf{x}(i, j) = \mathbf{x} = (x_1, x_2, \dots, x_N)$$

the vector of spectral values of window defined context neighborhood in a fixed arrangement, i.e. for each pixel three spectral values. Then in each position the window interior (image patch)  $\mathbf{x}$  can be viewed as an observation of a random vector and therefore we can describe the statistical properties of the variables  $x_n$

in full generality by a joint probability density. For this purpose we approximate the unknown density function in the form of Gaussian mixture

$$P(\mathbf{x}) = \sum_{m \in \mathcal{M}} w_m F(\mathbf{x} | \boldsymbol{\mu}_m, \boldsymbol{\sigma}_m), \quad \mathbf{x} \in R^N. \quad (1)$$

Assuming conditional independence of variables we define the mixture components as products of univariate Gaussian densities [5], [6]:

$$F(\mathbf{x} | \boldsymbol{\mu}_m, \boldsymbol{\sigma}_m) = \prod_{n \in \mathcal{N}} f_n(x_n | \mu_{mn}, \sigma_{mn}), \quad (2)$$

$$f_n(x_n | \mu_{mn}, \sigma_{mn}) = \frac{1}{\sqrt{2\pi}\sigma_{mn}} \exp \left\{ -\frac{(x_n - \mu_{mn})^2}{2\sigma_{mn}^2} \right\}. \quad (3)$$

To simplify notation we denote  $\mathcal{M} = \{1, 2, \dots, M\}$  and  $\mathcal{N} = \{1, 2, \dots, N\}$  the index sets of components and variables respectively.

The standard way to estimate mixtures is to use the EM algorithm [10]. By using the “image patch” data set  $\mathcal{S}$  obtained by pixel-wise shifting the observation window through the original texture image

$$\mathcal{S} = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(K)}\}, \quad \mathbf{x}^{(k)} \in \mathcal{R}^N, \quad (K = |\mathcal{S}|), \quad (4)$$

we maximize the corresponding log-likelihood function

$$L = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{x} \in \mathcal{S}} \log \left[ \sum_{m \in \mathcal{M}} w_m F(\mathbf{x} | \boldsymbol{\mu}_m, \boldsymbol{\sigma}_m) \right] \quad (5)$$

by means of the well-known EM iteration equations [1]:

E-step:  $(m \in \mathcal{M}, n \in \mathcal{N}, \mathbf{x} \in \mathcal{S})$

$$q(m|\mathbf{x}) = \frac{w_m F(\mathbf{x} | \boldsymbol{\mu}_m, \boldsymbol{\sigma}_m)}{\sum_{j \in \mathcal{M}} w_j F(\mathbf{x} | \boldsymbol{\mu}_j, \boldsymbol{\sigma}_j)}, \quad m \in \mathcal{M} \quad (6)$$

M-step:

$$w'_m = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{x} \in \mathcal{S}} q(m|\mathbf{x}), \quad \mu'_{mn} = \frac{1}{\sum_{\mathbf{x} \in \mathcal{S}} q(m|\mathbf{x})} \sum_{\mathbf{x} \in \mathcal{S}} x_n q(m|\mathbf{x}), \quad (7)$$

$$(\sigma'_{mn})^2 = -(\mu'_{mn})^2 + \frac{1}{\sum_{\mathbf{x} \in \mathcal{S}} q(m|\mathbf{x})} \sum_{\mathbf{x} \in \mathcal{S}} x_n^2 q(m|\mathbf{x}). \quad (8)$$

Here the apostrophe denotes the new parameter values in each iteration.

The local mixture model  $P(\mathbf{x})$  provides fully general description of statistical dependencies of pixel variables in the observation window. For any given image patch  $\mathbf{x} \in \mathcal{S}$  we can compute the corresponding conditional weights  $q(m|\mathbf{x}), m \in \mathcal{M}$  which can be viewed as highly informative features describing the textural properties of the image patch  $\mathbf{x}$  in terms of its affinity with the component means  $\boldsymbol{\mu}_m$ . It is intuitively clear that the context information contained in  $q(m|\mathbf{x})$  increases with the window size but, simultaneously, the related textural properties become less local. The underlying density estimation problem also becomes more difficult with the increasing dimension  $N$ .

### 3 Segmentation Algorithm

By its nature the EM algorithm produces a set of mixture components which correspond to different local properties of the source texture image, as it can be seen in Fig. 1. The first row shows the color texture examples to be segmented [11] and the second row shows the corresponding mixture models. The component means  $\boldsymbol{\mu}_m$  (in the arrangement of the observation window) can be viewed as averaged (smoothed) representants of the typical image patch variants. All the three mixture densities have been estimated in the space of dimension  $N=1143$  without any feature extraction (window size:  $21 \times 21$  pixels with the corners cut away, i.e.  $N=381 \times 3=1143$ ). The number of mixture components has been chosen  $M_1 = 64$ ,  $M_2 = 59$  and  $M_3 = 64$  respectively.

The basic idea of segmentation is to unify similar texture pieces. As the mixture component means  $\boldsymbol{\mu}_m$  correspond to different variants of the image patches we assume that different parts of the texture can be characterized by aggregating the related mixture components, i.e. by decomposing the Gaussian mixture  $P(\boldsymbol{x})$  into sub-mixtures. In particular, let  $\mathfrak{S}$  be a partition of the index set  $\mathcal{M}$  into disjunct subsets  $\mathcal{M}_k \subset \mathcal{M}$ :

$$\mathfrak{S} = \{\mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_M\}, \quad \cup_{k \in \mathcal{M}} \mathcal{M}_k = \mathcal{M}, \quad \mathcal{M}_k \cap \mathcal{M}_j = \emptyset, k \neq j. \quad (9)$$

Then we can define the corresponding decomposition of the mixture density (1) into sub-mixtures:

$$P(\boldsymbol{x}) = \sum_{k \in \mathcal{M}} P_k(\boldsymbol{x}) = \sum_{k \in \mathcal{M}} \sum_{m \in \mathcal{M}_k} w_m F(\boldsymbol{x} | \boldsymbol{\mu}_m, \boldsymbol{\sigma}_m), \quad (10)$$

$$P_k(\boldsymbol{x}) = \sum_{m \in \mathcal{M}_k} w_m F(\boldsymbol{x} | \boldsymbol{\mu}_m, \boldsymbol{\sigma}_m). \quad (11)$$

We assume a sub-mixture  $P_k(\boldsymbol{x})$  corresponding to  $\mathcal{M}_k$  to be zero in case of empty subset  $\mathcal{M}_k = \emptyset$ . Given the sub-mixtures  $P_k(\boldsymbol{x})$  we can classify the image patches  $\boldsymbol{x} \in \mathcal{S}$  by means of Bayes formula. Let us recall that each vector  $\boldsymbol{x} = \boldsymbol{x}(i, j) \in \mathcal{S}$  uniquely corresponds to the position  $(i, j)$  of central pixel of the observation window and also for each  $\boldsymbol{x} \in \mathcal{S}$  we can evaluate the probabilities

$$p(k | \boldsymbol{x}) = \frac{P_k(\boldsymbol{x})}{P(\boldsymbol{x})} = \sum_{m \in \mathcal{M}_k} q(m | \boldsymbol{x}), \quad k \in \mathcal{M}. \quad (12)$$

Here  $p(k | \boldsymbol{x})$  is the probability that the central pixel of the image patch  $\boldsymbol{x}$  belongs to the texture segment  $\mathcal{S}_k$  which is characterized by the sub-mixture  $P_k(\boldsymbol{x})$ . Ignoring some border pixels, we can define a segmentation of the texture image as a partition  $\mathfrak{R}$  of the set  $\mathcal{S}$  into disjunct subsets  $\mathcal{S}_k \subset \mathcal{S}$ :

$$\mathfrak{R} = \{\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_M\}, \quad \cup_{k \in \mathcal{M}} \mathcal{S}_k = \mathcal{S}, \quad \mathcal{S}_k \cap \mathcal{S}_j = \emptyset, k \neq j. \quad (13)$$

In view of Eq. (12) the optimal subsets  $\mathcal{S}_k$  given  $\mathfrak{S}$  can be defined by means of the Bayes decision function <sup>1</sup>

$$d(\mathbf{x}|\mathfrak{S}) = \arg \max_{k \in \mathcal{M}} \{p(k|\mathbf{x})\} = \arg \max_{k \in \mathcal{M}} \left\{ \sum_{m \in \mathcal{M}_k} q(m|\mathbf{x}) \right\} \quad (14)$$

$$\mathcal{S}_k = \{\mathbf{x} \in \mathcal{S} : d(\mathbf{x}|\mathfrak{S}) = k\}, \quad k \in \mathcal{M} \quad (15)$$

where  $d(\mathbf{x}|\mathfrak{S})$  specifies the sub-mixture with the maximum a posteriori probability  $p(k|\mathbf{x})$  given  $\mathbf{x} \in \mathcal{S}$ .

In view of the available mixture model a natural way to measure the quality of the texture segmentation  $\mathfrak{R}$  is to compute the mean probability of correct pixel classification with respect to the given segments  $\mathcal{S}_k \in \mathfrak{R}$ :

$$Q(\mathfrak{S}, \mathfrak{R}) = \frac{1}{|\mathcal{S}|} \sum_{k \in \mathcal{M}} \sum_{\mathbf{x} \in \mathcal{S}_k} p(k|\mathbf{x}) = \frac{1}{|\mathcal{S}|} \sum_{k \in \mathcal{M}} \sum_{\mathbf{x} \in \mathcal{S}_k} \sum_{m \in \mathcal{M}_k} q(m|\mathbf{x}). \quad (16)$$

In the above criterion  $p(k|\mathbf{x})$  denotes the probability that the pixel  $(i, j)$  has been classified correctly, i.e. that  $\mathbf{x} = \mathbf{x}(i, j) \in \mathcal{S}_k$ . Given a segmentation  $\mathfrak{R}$  we define the optimal mixture decomposition  $\mathfrak{S}$  by Eqs.:

$$\varphi(m|\mathfrak{R}) = \arg \max_{k \in \mathcal{M}} \left\{ \sum_{\mathbf{x} \in \mathcal{S}_k} q(m|\mathbf{x}) \right\}, \quad (17)$$

$$\mathcal{M}_k = \{m \in \mathcal{M} : \varphi(m|\mathfrak{R}) = k\}, \quad k \in \mathcal{M}. \quad (18)$$

Here  $\varphi(m|\mathfrak{R})$  specifies the segment  $\mathcal{S}_k$  with the greatest ‘‘contribution’’ of the  $m$ -th mixture component. The criterion  $Q(\mathfrak{S}, \mathfrak{R})$  can be maximized by repeating the iterative steps (15), (18). In particular we prove:

**Theorem.** The iterative use of Eqs. (15) and (18) produces a nondecreasing sequence of values of the criterion  $Q(\mathfrak{S}, \mathfrak{R})$  converging in a finite number of steps to a finite limit.

**Proof.** Let us note first that any change of the mixture decomposition  $\mathfrak{S}$  can be viewed as a result of a sequence of elementary steps. In particular, let  $\mathfrak{S}^+$  be defined by an elementary change of  $\mathfrak{S}$  (cf. (17)):

$$\mathcal{M}_k^+ = \mathcal{M}_k \cup \{m_0\}, \quad \mathcal{M}_l^+ = \mathcal{M}_l \setminus \{m_0\}, \quad \mathcal{M}_j^+ = \mathcal{M}_j, \quad j \in \mathcal{M}, \quad j \neq k, l$$

with  $m_0$  satisfying the condition  $\varphi(m_0|\mathfrak{R}) = k$  (cf. (18)). Therefore the following inequality holds (cf. (17))

$$\sum_{\mathbf{x} \in \mathcal{S}_k} q(m_0|\mathbf{x}) \geq \sum_{\mathbf{x} \in \mathcal{S}_j} q(m_0|\mathbf{x}), \quad \forall j \in \mathcal{M} \quad (19)$$

and the corresponding change of the criterion  $Q$  (cf. (16)) is non-negative

<sup>1</sup> If the maximum is not unique we choose the smallest index  $k \in \mathcal{M}$  with the specified property.

$$Q(\mathfrak{S}^+, \mathfrak{R}) - Q(\mathfrak{S}, \mathfrak{R}) = \frac{1}{|\mathfrak{S}|} \left[ \sum_{\mathbf{x} \in \mathcal{S}_k} q(m_0 | \mathbf{x}) - \sum_{\mathbf{x} \in \mathcal{S}_l} q(m_0 | \mathbf{x}) \right] \geq 0. \quad (20)$$

Consequently, the inequality (20) is also valid for the resulting ‘‘cumulative’’ decomposition as defined by (18).

Analogously, let  $\mathfrak{R}^+$  is defined by an elementary change of texture segmentation  $\mathfrak{R}$ :

$$\mathcal{S}_k^+ = \mathcal{S}_k \cup \{\mathbf{x}_0\}, \quad \mathcal{S}_l^+ = \mathcal{S}_l \setminus \{\mathbf{x}_0\}, \quad \mathcal{S}_j^+ = \mathcal{S}_j, \quad j \neq k, l$$

with  $\mathbf{x}_0$  satisfying the condition  $d(\mathbf{x}_0 | \mathfrak{S}) = k$  (cf. (15)). Therefore the following inequality is satisfied (cf. (14)):

$$p(k | \mathbf{x}_0) \geq p(j | \mathbf{x}_0), \quad \forall j \in \mathcal{M}, \quad (21)$$

and the corresponding change of the criterion (16) is non-negative

$$Q(\mathfrak{S}, \mathfrak{R}^+) - Q(\mathfrak{S}, \mathfrak{R}) = \frac{1}{|\mathfrak{S}|} [p(k | \mathbf{x}_0) - p(l | \mathbf{x}_0)] \geq 0.$$

It can be seen that the last inequality is also valid for any complex change of the segmentation  $\mathfrak{R}$ .

Consequently, the iterative segmentation algorithm (15), (18) based on the local mixture model  $P(\mathbf{x})$  converges to a finite limit in a finite number of steps because the criterion  $Q(\mathfrak{S}, \mathfrak{R})$  is bounded above and the number of possible partitions  $\mathfrak{S}$  and  $\mathfrak{R}$  is finite.

## 4 Topologically Modified Segmentation

The ‘‘bottom up’’ segmentation algorithm from Sec. 3 starting with the finest partition  $\mathfrak{S}_0 : \mathcal{M}_k = \{k\}, k \in \mathcal{M}$  and maximizing the criterion  $Q(\mathfrak{S}, \mathfrak{R})$  converges in few iterations to a highly over-segmented texture. The tendency to over-segmentation is closely related to the high dimensionality of the estimated density  $P(\mathbf{x})$ . Even in case of a small size the observation window may contain several hundreds of pixels with three spectral values and the resulting dimension  $N$  is very high. In multidimensional spaces the mixture components are nearly non-overlapping and therefore there is usually one-to-one correspondence between the mixture components and the initial highly specific texture segments. The examples of the initial over-segmentation are shown in Fig. 1 (third row). Recall that in case of the mixture components from Fig. 1 the underlying densities have dimension  $N = 1143$ . In high-dimensional spaces the mixture components are well separated and almost non-overlapping and therefore the conditional probabilities  $q(m | \mathbf{x})$  have nearly binary properties.

In order to increase the sensitivity of the criterion (16) with respect to the topological properties of the texture we include a neighborhood into the underlying decision-making. We define the decision neighborhood  $\mathcal{D}(\mathbf{x}(i, j))$  of the vector  $\mathbf{x}(i, j) \in \mathcal{S}$  as a subset of vectors  $\mathbf{x} \in \mathcal{S}$  which correspond to the pixels

near the central point  $(i, j)$ . In particular, considering a square decision neighborhood with the corners cut away, we can write:

$$\mathcal{D}(\mathbf{x}(i, j)) = \{\mathbf{x}(k, l) \in \mathcal{S} : |i - k| + |j - l| < (2\rho - r)\}. \quad (22)$$

where  $\rho \geq 0$  is a window “radius” and  $r = \rho/2$  the corner size (cf. Fig. 1, second row). To simplify notation, the position  $(i, j)$  of the observation window will be omitted whenever tolerable.

By using the neighborhood  $\mathcal{D}(\mathbf{x})$  we can compute the mean probability of correct pixel classification in a more robust way by taking in account the neighbouring pixels:

$$Q(\mathfrak{S}, \mathfrak{R}) \approx \frac{1}{|\mathcal{S}|} \sum_{k \in \mathcal{M}} \sum_{\mathbf{x} \in \mathcal{S}_k} p(k|\mathcal{D}(\mathbf{x})) \quad (23)$$

Here the probability  $p(k|\mathcal{D}(\mathbf{x}))$  of correct classification of a pixel  $\mathbf{x} = \mathbf{x}(i, j) \in \mathcal{S}$  can be expressed in the form:

$$p(k|\mathcal{D}(\mathbf{x})) = \frac{P_k(\mathcal{D}(\mathbf{x}))}{P(\mathcal{D}(\mathbf{x}))} = \sum_{\mathbf{y} \in \mathcal{D}(\mathbf{x})} \frac{P_k(\mathbf{y})}{P(\mathcal{D}(\mathbf{x}))} = \sum_{\mathbf{y} \in \mathcal{D}(\mathbf{x})} \frac{P(\mathbf{y})}{P(\mathcal{D}(\mathbf{x}))} p(k|\mathbf{y}). \quad (24)$$

If we assume the term  $P(\mathbf{y})/P(\mathcal{D}(\mathbf{x}))$  to be approximately constant for all  $\mathbf{y} \in \mathcal{D}(\mathbf{x})$  then it can be replaced by a coefficient  $1/|\mathcal{D}(\mathbf{x})|$ . Denoting  $|\mathcal{D}(\mathbf{x})| = D_0$  we can write the criterion (23) in the following more suitable form:

$$Q_{\mathcal{D}}(\mathfrak{S}, \mathfrak{R}) = \frac{1}{|\mathcal{S}|} \sum_{k \in \mathcal{M}} \sum_{\mathbf{x} \in \mathcal{S}_k} \frac{1}{D_0} \sum_{\mathbf{y} \in \mathcal{D}(\mathbf{x})} p(k|\mathbf{y}) \quad (25)$$

If we modify definition (15) of the segmentation  $\mathfrak{R}$  given the decomposition  $\mathfrak{S}$

$$d_{\mathcal{D}}(\mathbf{x}|\mathfrak{S}) = \arg \max_{k \in \mathcal{M}} \left\{ \sum_{m \in \mathcal{M}_k} \sum_{\mathbf{y} \in \mathcal{D}(\mathbf{x})} q(m|\mathbf{y}) \right\}. \quad (26)$$

$$\mathcal{S}_k = \{\mathbf{x} \in \mathcal{S} : d_{\mathcal{D}}(\mathbf{x}|\mathfrak{S}) = k\}, \quad k \in \mathcal{M}, \quad (27)$$

and the definition (18) of mixture decomposition  $\mathfrak{S}$  given the segmentation  $\mathfrak{R}$ :

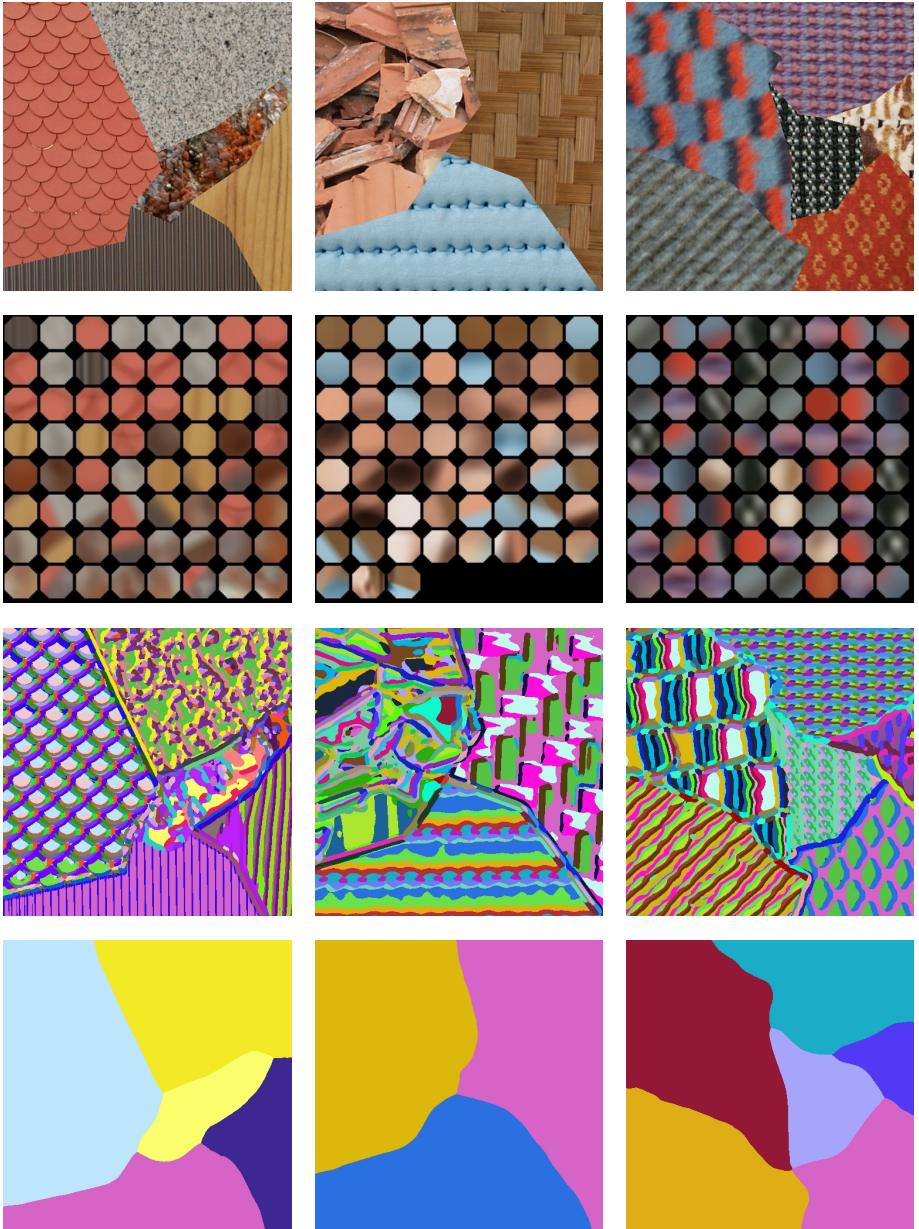
$$\varphi_{\mathcal{D}}(m|\mathfrak{R}) = \arg \max_{k \in \mathcal{M}} \left\{ \sum_{\mathbf{x} \in \mathcal{S}_k} \sum_{\mathbf{y} \in \mathcal{D}(\mathbf{x})} q(m|\mathbf{y}) \right\}, \quad (28)$$

$$\mathcal{M}_k = \{m \in \mathcal{M} : \varphi_{\mathcal{D}}(m|\mathfrak{R}) = k\}, \quad k \in \mathcal{M} \quad (29)$$

then the criterion (25) is again maximized by iterating the steps (27) and (29). The proof of this assertion is analogous to that of Sec. 3.

## 5 Computational Experiments

The practical implementation of the above algorithm starts with the highly over-segmented texture from Sec. 3 which corresponds to the neighborhood radius



**Fig. 1.** Texture segmentation experiments. Each column corresponds to one example taken from [11]. The texture image is in the first row, the second row shows the component means  $\mu_m$  of the respective Gaussian mixture model and the third and fourth row show the initial and final segmentation respectively. The resulting segmentation is essentially correct despite the strong inhomogeneity of the texture segments. There are only minor “rounding” errors on the segment boundaries.



$\rho = 0$ , i.e. to  $|D(\mathbf{x})| = 1$ . Then the topologically modified segmentation algorithm is repeatedly started again with an increased neighborhood  $D(\mathbf{x})$  and iterated until convergence. The repeated application of the segmentation algorithm represents a hierarchical scheme which can be stopped e.g. when the number of segments does not change for several repetitions or by choosing an ad hoc maximum size of the final decision neighborhood.

In the experiments we have applied the proposed segmentation algorithm to the Prague texture segmentation benchmark which is known to contain very difficult examples. The three test images shown in the first row of Fig. 1 are available at the address [11]. For each texture image we have computed the local statistical model in the form of Gaussian mixture for the window size  $21 \times 21$  (dimension  $N=1143$ , cf. Sec. 3 for details). The second row of Fig. 1 shows the corresponding component means for the three mixtures - as discussed in Sec. 3. The initial “over-segmented” images in the third row illustrate the “discriminative power” of the models and finally the fourth row shows the resulting segmentations. In the considered examples 1-3 the segmentation algorithm has been stopped for the size of decision neighborhood  $\rho = 28$ ,  $\rho = 33$  and  $\rho = 24$  respectively. The results of segmentation are rather convincing. It can be seen that, despite strong inhomogeneity, all texture segments have been correctly identified, possibly except for some minor “rounding” errors on the segment boundaries.

In the present form the proposed method is rather demanding (several hours) mainly because of the time-consuming mixture estimation. However, the resulting segmentation can be obtained more quickly when the size of the decision neighborhood can be specified in advance.

## 6 Conclusion

In the present paper we propose a color texture segmentation algorithm based on statistical model of local texture properties. We describe the statistical dependencies between the spectral pixel values in a suitably chosen observation window by a multivariate Gaussian mixture with product components. We estimate the mixture parameters by means of EM algorithm from color image patch data obtained by pixelwise shifting the observation window through the original color texture image. No feature extraction- or dimensionality reduction technique is applied to the spectral pixel variables. As the mixture components correspond to different “averaged” variants of the image patches we identify each texture segment by a corresponding sub-mixture by using Bayes formula. We propose a simple segmentation criterion in terms of probability of correct pixel classification into segments and an iterative algorithm to maximize the proposed criterion. Simultaneously, we prove that the convergence of the algorithm is monotonic in a finite number of steps.

In the present form the proposed texture segmentation method is time-consuming and therefore hardly applicable on-line. On the other hand it may be useful, e.g. for careful off-line evaluation of medical images.

**Acknowledgement.** This research was supported by the EC project no. FP6-507752 MUSCLE, by the grant No. 1ET400750407 of the Grant Agency of the Academy of Sciences CR and partially by the projects MŠMT 1M0572 DAR and 2C06019 ZIMOLEZ.

## References

1. Grim, J., Kittler, J., Pudil, P., Somol, P.: Multiple classifier fusion in probabilistic neural networks. *Pattern Analysis & Applications*, 7, 5 (2002) 221-233.
2. Grim, J., Haindl, M.: A discrete mixtures color texture model. In: *Texture 2002. The 2nd International Workshop on Texture Analysis and Synthesis*, Copenhagen 2002. (Chantler M. ed.). Heriot-Watt Univ., Glasgow, (2002) 59-62
3. Grim, J., Haindl, M.: Texture Modelling by Discrete Distribution Mixtures. *Computational Statistics and Data Analysis*, 3-4, 41 (2003) 603-615
4. Grim, J., Somol, P., Haindl, M., Pudil, P.: A statistical approach to local evaluation of a single texture image. In: *Proceedings of the 16-th Annual Symposium PRASA 2005*. (Nicolls F. ed.). University of Cape Town, (2005) 171-176
5. Haindl, M., Grim, J., Somol, P., Pudil, P., Kudo, M.: A Gaussian mixture-based color texture model. In: *Proceedings of the 17th IAPR International Conference on Pattern Recognition*. IEEE, Los Alamitos, (2004) 177-180
6. Haindl, M., Grim, J., Pudil, P., Kudo, M.: A Hybrid BTF Model Based on Gaussian Mixtures. In: *Texture 2005. Proceedings of the 4th International Workshop on Texture Analysis*. (Chantler M., Drbohlav O. eds.). IEEE, Los Alamitos, (2005) 95-100
7. Heidemann, G., Ritter, H.: A neural 3-D object recognition architecture using optimized Gabor filters. In: *Proceedings of the 13th IAPR International Conference on Pattern Recognition*. Vol. IV, 70-74, Los Alamitos CA, IAPR, IEEE Computer Society Press, 1996
8. Ridder, D. de, Kittler, J., Lemmers, O., Duin, R.P.W.: The adaptive sub-space map for texture segmentation. In: *Proceedings of the 15th IAPR International Conference on Pattern Recognition*, Eds. A. Sanfeliu et al., 216-220, IAPR, IEEE Computer Society Press, 2000
9. Ridder, D. de, Kittler, J., Duin, R.P.W.: Probabilistic PCA and ICA subspace mixture models for image segmentation. In: *Proceedings of the British Machine Vision Conference*, Eds. M. Mirmehdi et al., (2000) 112-121
10. McLachlan, G.J. and Peel, D.: *Finite Mixture Models*, John Wiley & Sons, New York, (2000)
11. *Prague texture segmentation benchmark (2004)*: <http://mosaic.utia.cas.cz>
12. Reed, T.R., du Buf, J.M.H.: A review of recent texture segmentation and feature extraction techniques. *CVGIP-Image Understanding*, 57 (1993) 359-372
13. Zwiggelaar, R., Planiol, P., Mart, R., Blot, L., Denton, E.R.E., Rubin, C.M.E.: EM Texture Segmentation of Mammographic Images. In: *Proceedings of the 6-th International Workshop on Digital Mammography*, Germany, (2002) 541-543