# Color Texture Segmentation by Decomposition of Gaussian Mixture Model

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

**Institute of Information Theory and Automation**
**Academy of Sciences of the Czech Republic, Prague**

**Department of Pattern Recognition**
**http://www.utia.cas.cz/RO**

Conference CIARP, Cancun, Mexico 2006

# Outline


1. Local Statistical Texture Model
   - Gaussian Mixture Model
   - EM Algorithm for Gaussian Mixtures
2. Model Estimation
   - Computational Experiment
   - Texture Examples
   - Theoretical Aspects
3. Texture Segmentation by Mixture Decomposition
   - Segmentation Principle
   - Iterative Segmentation Algorithm
4. Topological Segmentation
   - Topological Segmentation Principle
   - Topologically Modified Segmentation Algorithm
   - Segmentation Examples
5. Conclusions

# Local Statistical Gaussian Mixture Model

**Digitized color texture image:**

$$\mathcal{Z} = [\boldsymbol{z}_{ij}]_{i=1}^{I}\,{}_{j=1}^{J}, \quad \boldsymbol{z}_{ij} = (z_{ij1}, z_{ij2}, z_{ij3}) \in \mathcal{R}^3 \;\approx\; \text{RGB spectral values}$$

**Assumption:**

local statistical properties of spectral pixel values are specific for different parts of texture image

**window interior (patch):** $\boldsymbol{x} = (x_1, x_2, \ldots, x_N) \in \mathcal{X}, \quad \mathcal{X} = \mathcal{R}^N$

**Method:**

approximation of the joint multivariate probability density $P(\boldsymbol{x})$ by normal mixture of product components:

$$P(\boldsymbol{x}) = \sum_{m \in \mathcal{M}} w_m F(\boldsymbol{x}|\boldsymbol{\mu}_m, \boldsymbol{\sigma}_m) = \sum_{m \in \mathcal{M}} w_m \prod_{n \in \mathcal{N}} f_n(x_n|\mu_{mn}, \sigma_{mn})$$

$$\mathcal{M} = \{1, \ldots, M\}, \quad \mathcal{N} = \{1, \ldots, N\} \;\approx\; \text{index sets}$$

# EM Algorithm for Normal Product Mixtures

**dat set:** $\mathcal{S} = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(K)}\}$ $\approx$ by shifting observation window

**components:**

$$F(\boldsymbol{x}|\boldsymbol{\mu}_m, \boldsymbol{\sigma}_m) = \prod_{n \in \mathcal{N}} \frac{1}{\sqrt{(2\pi)}\sigma_{mn}} \exp\left\{ -\frac{(x_n - \mu_{mn})^2}{2\sigma_{mn}^2} \right\}$$

**log-likelihood criterion:**

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

**EM Algorithm:**

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

$$w_m' = \frac{1}{|\mathcal{S}|} \sum_{\boldsymbol{x} \in \mathcal{S}} q(m|\boldsymbol{x}), \qquad \mu_{mn}' = \frac{1}{\sum_{\boldsymbol{x} \in \mathcal{S}} q(m|\boldsymbol{x})} \sum_{\boldsymbol{x} \in \mathcal{S}} x_n q(m|\boldsymbol{x})$$

$$(\sigma_{mn}')^2 = -(\mu_{mn}')^2 + \frac{1}{\sum_{\boldsymbol{x} \in \mathcal{S}} q(m|\boldsymbol{x})} \sum_{\boldsymbol{x} \in \mathcal{S}} x_n^2 q(m|\boldsymbol{x}), \ \ n \in \mathcal{N}$$

# Model Estimation - Computational Experiment

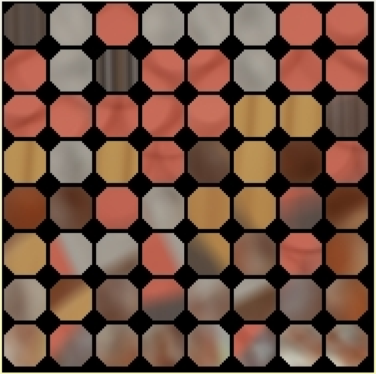
- source texture images of size 512x512 pixels ($|\mathcal{S}| \approx 250000$)
- context information contained in $q(m|\boldsymbol{x})$ increases with window size but, simultaneously, the related textural properties become less local
- dimension of the window data vector $\boldsymbol{x}$: $N = 1143 = 3x381$ (for window size 21x21 pixels with cut-off corners)
- no feature extraction or dimensionality reduction technique applied to pixel variables
- number of mixture components: $M_1 = 64$, $M_2 = 59$ and $M_3 = 64$
- EM algorithm: random initialization, 10 - 20 iterations
- (!) image patches obtained by shifting the window are overlapping and therefore the corresponding data vectors are not independent
- $\Rightarrow$ data set $\mathcal{S}$ is less representative (bad sampling properties)

ŪTIA

# Prague Segmentation Benchmark - Example 1

**original image**

**component means**

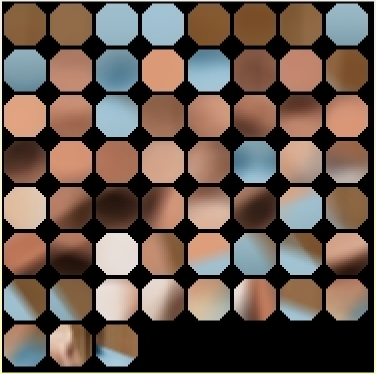


- window size: 21x21 pixels with cut-off corners
- dimension of data vector $\boldsymbol{x}$: $N = 1143$
- number of mixture components: $M = 64$

# Prague Segmentation Benchmark - Example 2

**original image**

**component means**

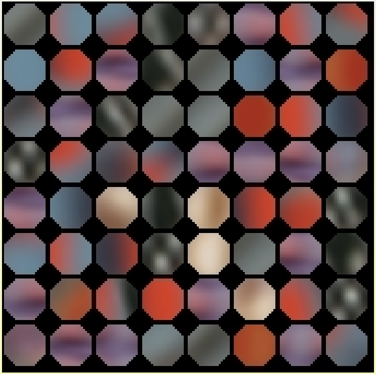


- window size: 21x21 pixels with cut-off corners
- dimension of data vector $\boldsymbol{x}$: $N = 1143$
- number of mixture components: $M = 59$

# Prague Segmentation Benchmark - Example 3

**original image**

**component means**



- window size: 21x21 pixels with cut-off corners
- dimension of data vector $\boldsymbol{x}$: $N = 1143$
- number of mixture components: $M = 64$

# Theoretical Aspects of Mixture Model Application

- unlike other fields (e.g. texture modelling) the estimated Gaussian mixture $P(\boldsymbol{x})$ is applied to the "training" data set $\mathcal{S}$ again
- limited representativeness of the set $\mathcal{S}$ is less relevant since $P(\boldsymbol{x})$ is not applied to the data not contained in $\mathcal{S}$
- log-likelihood criterion optimally "fits" the estimated mixture $P(\boldsymbol{x})$ to the data set $\mathcal{S}$ (risk of "over-fitting" is less relevant)
- $\Rightarrow$ mixture component means (in window arrangement) correspond to different variants of patches occurring in the shifted window
- informativity of the estimated mixture model can be verified visually by successful texture synthesis References

**Hypothesis:**

Different parts (segments) of texture image can be characterized by specific subsets of mixture components, i.e. by decomposing the Gaussian mixture $P(\boldsymbol{x})$ into corresponding sub-mixtures.

# Segmentation Principle

**partition of the index set $\mathcal{M}$ into disjunct subsets $\mathcal{M}_k$:**

$$\Im = \{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_M\}, \quad \cup_{k\in\mathcal{M}}\mathcal{M}_k = \mathcal{M}, \quad \mathcal{M}_k \cap \mathcal{M}_j = \emptyset, k \neq j$$

**mixture decomposition into corresponding sub-mixtures:**

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

**classification of central pixel by window neighborhood vector $\boldsymbol{x}$:**

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

**partition of the set $\mathcal{S}$ into subsets $\mathcal{S}_k$:** (ties arbitrarily decided)

$$\mathcal{S}_k = \{\boldsymbol{x} \in \mathcal{S} : p(k|\boldsymbol{x}) \geq p(j|\boldsymbol{x}), \forall j \in \mathcal{M}\}, \; k \in \mathcal{M}$$

$$\Re = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_M\}, \quad \cup_{k\in\mathcal{M}}\mathcal{S}_k = \mathcal{S}, \quad \mathcal{S}_k \cap \mathcal{S}_j = \emptyset, k \neq j.$$

# Iterative Segmentation Algorithm

**Criterion:** mean probability of correct pixel classification

$$Q(\Im, \Re) = \frac{1}{|\mathcal{S}|} \sum_{k \in \mathcal{M}} \sum_{\boldsymbol{x} \in \mathcal{S}_k} p(k|\boldsymbol{x}) = \frac{1}{|\mathcal{S}|} \sum_{k \in \mathcal{M}} \sum_{\boldsymbol{x} \in \mathcal{S}_k} \sum_{m \in \mathcal{M}_k} q(m|\boldsymbol{x})$$

**Algorithm 1:** (initial partition of $\Im_0 : \mathcal{M}_k = \{k\}, k \in \mathcal{M}$)

STEP 1: define subsets $\mathcal{S}_k, k \in \mathcal{M}$ given the partition $\Im$ of $\mathcal{M}$:

$$\mathcal{S}_k = \{\boldsymbol{x} \in \mathcal{S} : \sum_{m \in \mathcal{M}_k} q(m|\boldsymbol{x}) \geq \sum_{m \in \mathcal{M}_j} q(m|\boldsymbol{x}), \forall j \in \mathcal{M}\}$$

STEP 2: define subsets $\mathcal{M}_k, k \in \mathcal{M}$ given the partition $\Re$ of $\mathcal{S}$:

$$\mathcal{M}_k = \{m \in \mathcal{M} : \sum_{\boldsymbol{x} \in \mathcal{S}_k} q(m|\boldsymbol{x}) \geq \sum_{\boldsymbol{x} \in \mathcal{S}_j} q(m|\boldsymbol{x}), \forall j \in \mathcal{M}\}$$

**Remark:** Algorithm converges monotonically in a finite number of steps.

# Topological Principle

(!) mixture components in high dimensions are nearly non-overlapping:

**Problem of over-segmentation:**

The "bottom up" Algorithm 1 maximizing the criterion $Q(\Im, \Re)$ converges in few iterations to a highly over-segmented texture.

**Idea:** robust pixel classification by using $\epsilon$-neighborhood $\mathcal{D}_\epsilon(\boldsymbol{x})$

$$\mathcal{D}_\epsilon(\boldsymbol{x}) = \mathcal{D}_\epsilon(\boldsymbol{x}(i,j)) = \{\boldsymbol{x}(k,l) \in \mathcal{S} : (i-k)^2 + (j-l)^2 < \epsilon^2\}$$

**Criterion:** setting $D_\epsilon = |\mathcal{D}_\epsilon(\boldsymbol{x})|$ we can write

$$Q(\Im, \Re) \approx \frac{1}{|\mathcal{S}|} \sum_{k \in \mathcal{M}} \sum_{\boldsymbol{x} \in \mathcal{S}_k} p(k|\mathcal{D}_\epsilon(\boldsymbol{x})) \doteq \frac{1}{|\mathcal{S}|} \sum_{k \in \mathcal{M}} \sum_{\boldsymbol{x} \in \mathcal{S}_k} \frac{1}{D_\epsilon} \sum_{\boldsymbol{y} \in \mathcal{D}_\epsilon(\boldsymbol{x})} p(k|\boldsymbol{y})$$

since

$$p(k|\mathcal{D}_\epsilon(\boldsymbol{x})) = \sum_{\boldsymbol{y} \in \mathcal{D}_\epsilon(\boldsymbol{x})} \frac{P_k(\boldsymbol{y})}{P(\mathcal{D}_\epsilon(\boldsymbol{x}))} = \sum_{\boldsymbol{y} \in \mathcal{D}_\epsilon(\boldsymbol{x})} \frac{P(\boldsymbol{y})}{P(\mathcal{D}_\epsilon(\boldsymbol{x}))} p(k|\boldsymbol{y}) \doteq \frac{1}{D_\epsilon} \sum_{\boldsymbol{y} \in \mathcal{D}_\epsilon(\boldsymbol{x})} p(k|\boldsymbol{y})$$

# Topologically Modified Segmentation Algorithm

**Criterion:** mean conditional probability of correct pixel classification

$$Q_\epsilon(\Im, \Re) = \frac{1}{|\mathcal{S}|} \sum_{k \in \mathcal{M}} \sum_{x \in \mathcal{S}_k} \frac{1}{D_\epsilon} \sum_{\mathbf{y} \in \mathcal{D}_\epsilon(\mathbf{x})} \sum_{m \in \mathcal{M}_k} q(m|\mathbf{y})$$

**Algorithm 2:** (initial partition of $\Im$, decision neighborhood radius $\epsilon$)

STEP 1: define subsets $\mathcal{S}_k, k \in \mathcal{M}$ given the partition $\Im$ of $\mathcal{M}$:

$$\mathcal{S}_k = \{\mathbf{x} \in \mathcal{S} : \sum_{\mathbf{y} \in \mathcal{D}_\epsilon(\mathbf{x})} \sum_{m \in \mathcal{M}_k} q(m|\mathbf{y}) \geq \sum_{\mathbf{y} \in \mathcal{D}_\epsilon(\mathbf{x})} \sum_{m \in \mathcal{M}_j} q(m|\mathbf{y}), \forall j \in \mathcal{M}\}$$

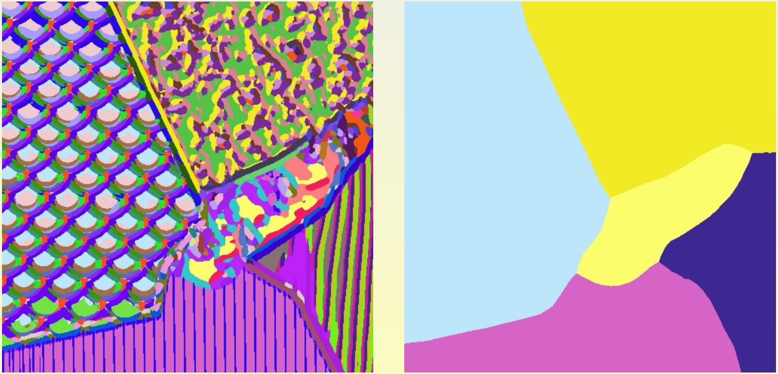
STEP 2: define sub-sets $\mathcal{M}_k, k \in \mathcal{M}$ given the partition $\Re$ of $\mathcal{S}$:

$$\mathcal{M}_k = \{m \in \mathcal{M} : \sum_{x \in \mathcal{S}_k} \sum_{\mathbf{y} \in \mathcal{D}_\epsilon(\mathbf{x})} q(m|\mathbf{y}) \geq \sum_{x \in \mathcal{S}_k} \sum_{\mathbf{y} \in \mathcal{D}_\epsilon(\mathbf{x})} q(j|\mathbf{y}), \forall j \in \mathcal{M}\}$$

**Remark:** After convergence the Algorithm 2 continues with an increased decision neighborhood radius $\epsilon$.

# Prague Segmentation Benchmark - Example 1

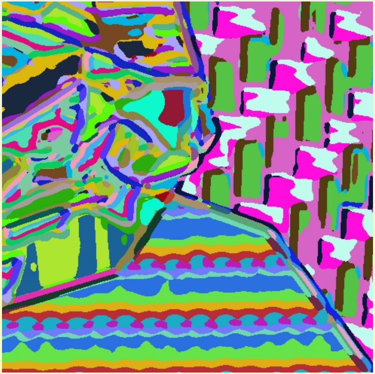
**initial over-segmentation** **final segments**



segmentation stopped for decision neighborhood: $\rho = 28$

# Prague Segmentation Benchmark - Example 2

**initial over-segmentation**

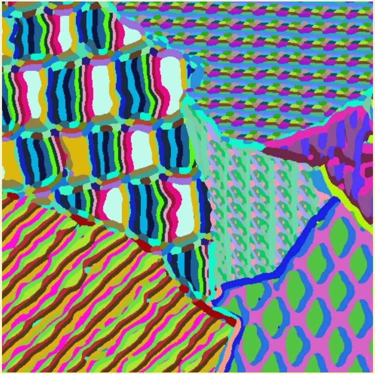


**final segments**

segmentation stopped for decision neighborhood: $\rho = 33$

# Prague Segmentation Benchmark - Example 3

**initial over-segmentation**



**final segments**

segmentation stopped for decision neighborhood: $\rho = 24$

# Conclusions

## Color Texture Segmentation by Using Local Statistical Model

- local texture properties described by Gaussian product mixture
- mixture parameters estimated from image patch data obtained by pixelwise shifting a suitably chosen observation window
- no feature extraction or dimensionality reduction technique is applied to the spectral pixel variables
- mixture component means (in window arrangement) correspond to different variants of patches occurring in the shifted window
- texture segments can be identified by corresponding sub-mixtures
- simple segmentation criterion in terms of probability of correct pixel classification is applied
- the proposed iterative algorithm is shown to maximize the segmentation criterion monotonically in a finite number of steps
- topological version of the algorithm is controlled by decision neighborhood radius

# Literatura 1/3


- 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
- Grim, J., Haindl, M.: Texture Modelling by Discrete Distribution Mixtures. Comp. Statist. and Data Analysis, 3-4, **41** (2003) 603-615
- 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
- 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


Return

# Literatura 2/3


- 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
- 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
- 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
- *Prague texture segmentation benchmark (2004):* htpp://mosaic.utia.cas.cz

# Literatura 3/3


- 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
- Zwiggelaar, R., Planiol, P., Martí, 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
- McLachlan, G.J. and Peel, D.: *Finite Mixture Models*, John Wiley & Sons, New York, (2000)
- Schlesinger, M.I. (1968): Relation between learning and self-learning in pattern recognition." (in Russian), *Kibernetika*, (Kiev), No. 2, 81-88.
- Titterington, D.M., Smith, A.F.M., & Makov, U.E. (1985): *Statistical analysis of finite mixture distributions.* New York: John Wiley & Sons, 1985.