

A Hierarchical Texture Model for Unsupervised Segmentation of Remotely Sensed Images

Giuseppe Scarpa^{1,2}, Michal Haindl², and Josiane Zerubia¹

¹ ARIANA Research Group, INRIA/I3S, Sophia Antipolis, France

² Pattern Recognition Dep., ÚTIA, Academy of Sciences, Prague, Czech Republic
giscarpa@unina.it, haindl@utia.cas.cz, zerubia@sophia.inria.fr

Abstract. In this work a novel texture model particularly suited for unsupervised image segmentation is proposed. Any texture is represented at region level by means of a finite-state hierarchical model resulting from the superposition of several Markov chains, each associated with a different spatial direction. Corresponding to such a modeling, an optimization scheme, referred to as Texture Fragmentation and Reconstruction (TFR) algorithm, has been introduced.

The TFR addresses the model estimation problem in two sequential layers: the former “fragmentation” step allows to find the terminal states of the model, while the latter “reconstruction” step is aimed at estimating the relationships among the states which provide the optimal hierarchical structure to associate with the model. The latter step is based on a probabilistic measure, i.e, the region gain, which accounts for both the region scale and the inter-region interaction.

The proposed segmentation algorithm was tested on a segmentation benchmark and applied to high resolution remote-sensing forest images as well.

Keywords: Segmentation, texture model, Markov chain, remote sensing, forest classification.

1 Introduction

Image segmentation [1,2,3,4] is a low-level processing which is of critical importance for many applications in several domains, like medical imaging, remote sensing, source coding, and so on. Although segmentation has been widely studied in the last decades in many cases it remains still open, as for textured images, where the spatial interactions may cover long ranges asking for high order complex modeling. In this work we focus on a remote sensing application, which is the segmentation of forest images [5,6] that represents a basic step for land cover classification and monitoring.

There are a large number of approaches to segmentation, but due to space limitations, here we confine ourselves to reviewing only those that have been tested using the same benchmarking system [7] as we use, and which therefore serve as points of comparison. In [8] image blocks are modeled by means of local

Gauss Markov Random Fields (GMRF) and the segmentation is performed in the parameter space by assuming an underlying Gaussian Mixture. Similar to the previous, but with an auto-regressive 3-D model (AR3D) in place of the Gauss MRF, is the method presented in [3]. In [9] an approach, namely the JSEG, is presented where segmentation is achieved in two steps: a color quantization followed by a processing of the label map which accounts for spatial interaction. Another method taken in consideration is the segmentation algorithm underlying the content-based image retrieval system *Blobworld* [1]. Here a Gaussian Mixture model is assumed in a feature space, where contrast, anisotropy and polarity are the salient texture descriptors, and the EM algorithm carries out the clustering. Finally, the algorithm presented in [10] (EDISON) combines a region-based approach with a contour-based one, hence balancing the global evidence which characterizes a region-based model with the local information typically dominant in the contour modeling.

In this work we present a method based on a hierarchical finite-state probabilistic texture modeling. The model is coupled with an optimization scheme, namely the Texture Fragmentation and Reconstruction (TFR) algorithm, which first estimates the states at the finest level (fragmentation), and then relates them hierarchically (reconstruction) as to provide the desired hierarchical segmentation.

In order to assess the accuracy of the proposed method, we have used the Prague Texture Segmentation Data Generator Benchmark [7] where all the algorithms mentioned above were tested as well. Furthermore, we provide a few results obtained by the TFR in the case of high resolution remotely sensed images portraying wooded areas.

2 Hierarchical Texture Model

In this work we present a *hierarchical, discrete* and *region-based* probabilistic model for texture representation, which is particularly suited for *unsupervised* image segmentation. In order to apply the model, an early processing is then needed to provide a discrete image that roughly represents the original data. In general this processing may be any known pixel-wise texture feature extraction followed by a clustering, but in practice we reduce it to a simple color-based segmentation, since the textural information will be handled in the discrete space. Obviously, this first operation is associated with an information loss which reduces the description capability of the model. However, while this could be a rather serious limit in a synthesis framework, it is not that critical in an analysis problem like segmentation, and especially in an unsupervised setting where robustness, rather than precision, is quite often the most relevant issue.

To introduce the model, let us consider the example in Fig.1, where a textile pattern (a) is associated with some graphical representations. Imagine first a simple 3-level discrete approximation of the data (say, the color-states *blue*, *black* and *red*), and consider its partition in uniform connected regions. A Region Adjacency Graph (RAG) representation of this partition is shown in (b).

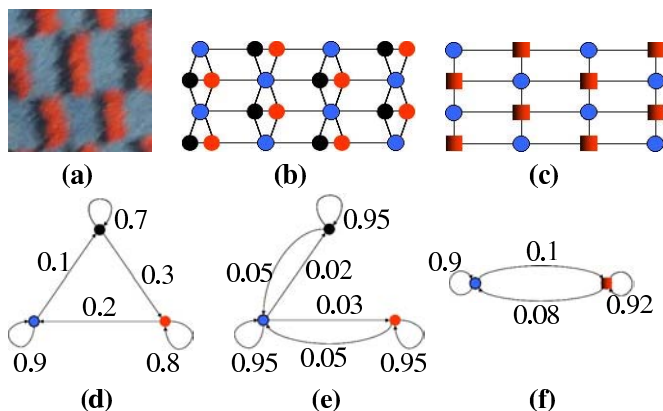


Fig. 1. Hierarchical texture model. Textile pattern (a); H-RAG: 3- and 2-state RAG, (b) and (c) respectively; 3-state chain models for east and south directions, (d) and (e) respectively; and 2-state chain for east direction (f).

Likewise, in case of a 2-state partition (for example, let *black* and *red* collapse in a single state) we would get a RAG like that depicted in (c). Notice that, by merging state *black* with *red* without involving the *blue* one, we established a clear relationship between the two graphs, which form together a Hierarchical RAG (H-RAG) [11]. In this toy example the H-RAG has only two layers because we have considered only two nested partitions, but in practice it has usually more layers as we start from much finer segmentations.

Now, let us observe how the textural properties are reflected in the adjacency graphs (b) and (c) as cyclic occurrence (strictly periodic in the specific example) of subgraphs of three and two nodes, respectively for (b) and (c). Such phenomenon can be synthetically represented for any given spatial direction by means of state diagrams, as in (d) and (e) for directions east and south respectively, when three color states are considered (b), and in (f) for east direction if we have only two states (c). As well as the RAGs, and for the same reasons, these diagrams are hierarchically related for any given direction, (see for example (d) and (f)). The example also clearly shows that, for a fixed periodical texture component, the coarser the scale of the RAG representation, the lesser the order at which it is revealed on the graph. In other words, the multiscale representation allow us to represent simultaneously both micro- and macro-textural features with the same (low) order but in different layers of the hierarchical model.

As can be seen, the compact representation (d)-(f) not only accounts for the adjacency among states but also for their directionality (mutual positioning) and relevance, through the specification of transition probabilities (TP) on a pixel-by-pixel step basis. Approximated TPs are indicated on the graphs just to give an idea of their relationship with the visual appearance of the texture. In particular, observe that intra-region TPs account for the shape of the texture components. As an example, consider the blue patches that regularly occur in the sample. Due to their rectangular shape, the associated intra-region TP in

the vertical direction (e) is larger than the horizontal one (d). The remaining, inter-region, TPs accounts instead for the spatial context, that is, the relative occurrence and positioning of the neighboring regions.

More precisely our texture model refers to the graphical representations introduced above and is basically a simultaneous hierarchical finite-state Markov model that for a given texture is completely defined by the triple $(\Omega, \mathcal{T}, \mathcal{P})$, where Ω is the set of states of the finest, but discrete, version of the texture, \mathcal{T} is a tree structure representing the hierarchical relationships among the states¹ and, finally, $\mathcal{P} = \{\mathbf{P}_\omega\}_{\omega \in \Omega}$ is the set of TP matrices (TPMs) for the terminal states. TPMs are given by

$$\mathbf{P}_\omega(\omega', j) = \frac{|\mathcal{S}_{\omega \rightarrow j \rightarrow \omega'}|}{|\mathcal{S}_\omega|} \quad \forall \omega' \in \Omega, \quad 1 \leq j \leq 8, \quad (1)$$

where \mathcal{S}_ω is the set of pixels with state ω and $\mathcal{S}_{\omega \rightarrow j \rightarrow \omega'}$ is the restriction of \mathcal{S}_ω to its sites whose neighbor in position (direction) j belongs to state ω' . While the TPMs defined above describe globally a texture, a single connected region element n of a given state ω has itself an own TPM, \mathbf{P}_ω^n , computed through the same formula but restricted to the region $\mathcal{S}_\omega^n \subseteq \mathcal{S}_\omega$.

Observe that at coarser level representations the states are completely defined by combination of related offspring states according to the given structure \mathcal{T} , which means that their TPMs are derived by simple weighted averages. Moreover, notice that in general a color may occur in a texture according to different configurations, hence increasing the number of states which do not necessarily represent different colors.

3 Texture Fragmentation and Reconstruction (TFR) Algorithm

Let us consider now the application of the above modeling in the particular case of unsupervised segmentation. The image to be segmented is then a composition of an unknown number of different textures whose corresponding models are unknown as well and need to be estimated during the process of texture identification. The model fitting consists in estimating the states (with related TPMs) at the finest scale and the hierarchical tree which univocally defines each intermediate state.

The determination of the number of textures of a given image, classically referred to as *cluster validation problem*, is strictly related to the spatial scale (hence to the hierarchical structure) at which we are interpreting the image. When the scale is not fixed somehow, the cluster validation becomes an ill-posed problem. To give an example, the same texture of Fig.1 may be interpreted as a composition of three different textures if we refer to a finer scale.

As a consequence we aim at solving this problem simultaneously with the estimation of the internal structures, according to the model defined above. In

¹ Hence, the states of Ω are associated with the terminal nodes, while the root represents the whole image.

practice, this means that we fit the image with only one hierarchical model which (when correctly derived) includes as non-overlapped substructures the marginal models associated with the single textures. Then, by specifying a spatial scale, we automatically get the proper pruning of the structure which provides us with the marginal models and the associated image partition.

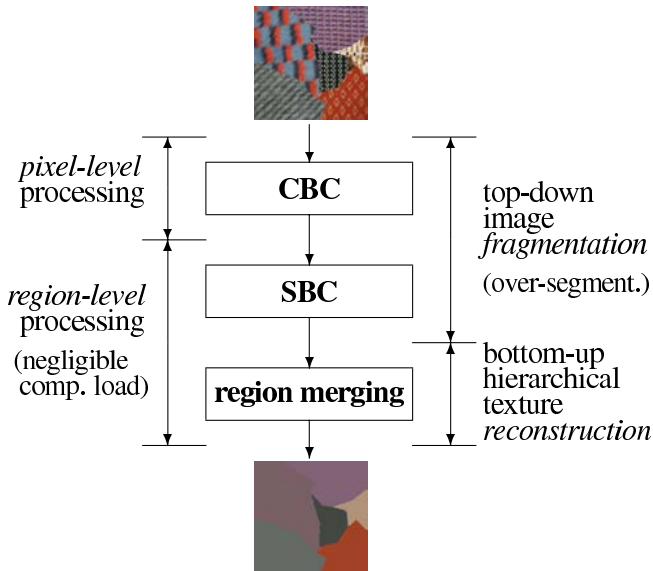


Fig. 2. TFR algorithm flow chart

In order to estimate this overall model we implemented the optimization scheme shown in Fig.2, namely the Texture Fragmentation and Reconstruction (TFR) algorithm, which first extracts a proper number of terminal states through the top-down fragmentation step, composed of blocks CBC (Color-Based Clustering) and SBC (Spatial-Based Clustering), and then relates them by means of a recursive bottom-up merging step, as to reconstruct the whole hierarchical structure.

The estimation of the states is performed in two steps, the former (CBC) dealing with color information, hence working at the pixel level, the latter (SBC) focused on the spatial information at the region level in the TPM space. In principle, CBC may be any color quantization process, but in our implementation we preferred the use of the TS-MRF (tree-structured Markov random field) segmentation algorithm [2], since it avoids the generation of punctiform regions (which are not reliably characterized in terms of TPM) due to regularization of the MRF. Furthermore, the tree-structured formulation ensures a quick processing and allows to balance the energy among the discrete color states.

Once the color segmentation has been obtained, we switch to a region-based representation, by taking connected regions with uniform color as basic elements

characterized by TPMs. Since the color of a region only partially defines its state², the SBC applies to each set of elements with common color, as to split it in subgroups which are homogeneous also in terms of TPM, that is providing the desired states. The split is realized by means of a k -means algorithm [12] applied in the feature space resulting from a PCA (Principal Component Analysis) [12] on the TPM space. The PCA was necessary because of the large dimensionality of the full feature space w.r.t. the number of elements which does not allow a reliable characterization.

The steps described above, CBC and SBC, realize the “fragmentation” whose goal is the estimation of the terminal states of the hierarchical model. The “reconstruction”, that is the estimation of the hierarchy structure, is realized by means of the region (or state) merging process, which is nothing but a sequential binary combination of the states driven by a specific parameter, namely the *region gain* which accounts for the mutual spatial relationships among the corresponding regions. Indeed the merging selection process is not symmetric, as the gain is a measure of the scale of the region weighted by an additional term which quantifies the attraction operated by the other regions (candidates for the merging). The scale factor allows to always privilege the merging of small regions so that the final hierarchy is such that micro-textural features are represented at the bottom, while the macro ones will appear at upper levels, and finally inter-texture mergings will be placed at the top of the structure, in order to keep separate the marginal sub-models corresponding to the different textures.

In this work we compare two different region gains. The former, already proposed in [13], is defined as

$$\mathcal{G}^i \triangleq \frac{p(s \in \mathcal{R}_i)}{\max_{j \neq i} p(r \in \mathcal{R}_j | s \in \mathcal{R}_i)} = p(s \in \mathcal{R}_i) \cdot \frac{1}{p(r \notin \mathcal{R}_i | s \in \mathcal{R}_i)} \cdot \frac{p(r \notin \mathcal{R}_i | s \in \mathcal{R}_i)}{\max_{j \neq i} p(r \in \mathcal{R}_j | s \in \mathcal{R}_i)}$$

where \mathcal{R}_i is the region of interest, s is an image site and r is any of the eight neighbors of s . The first two factors represent the scale, since one is proportional to the area of the region and the other quantifies its compactness. The third term accounts for the relative occurrence of the nearest neighbour region (context).

The latter, introduced here, is a modification of the former where the contextual term has been reinforced by means of the Kullback-Leibler Divergence (KLD), $D(q_i \| q_j)$, between the region spatial distributions, that is

$$\log \mathcal{G}_{\text{KLD}}^i \triangleq \min_{j \neq i} \left\{ \log \frac{p(s \in \mathcal{R}_i)}{p(r \in \mathcal{R}_j | s \in \mathcal{R}_i)} + D(q_i \| q_j) \right\}, \quad (2)$$

where q_i and q_j are normals (see details about KLD for Gaussians in [14]).

² More states may correspond to the same color, because either it appears in different configurations in a texture or it occurs in different textures.

4 Application to the Prague Benchmark and Numerical Evaluation

The proposed algorithm, that is the TFR or the TFR+ (when the gain includes the KLD term), is compared with other algorithms which were tested on the same benchmark system [7] and are briefly recalled in the introduction. The system provides a comparison w.r.t. a large number of indicators, some of which are region-based, some others are pixel-wise accuracy indicators, and a few of them give a measure of consistency. A complete description of all the parameters can be found on the system website [7].

Table 1. Up arrows indicate that larger values of the parameters are better; down arrows, the opposite. Benchmark criteria: CS, correct segmentation; OS, over-segmentation; US, under-segmentation; ME, missed error; NE, noise error; O, omission error; C, commission error; CA, class accuracy; CO, recall - correct assignment; CC, precision - object accuracy; I., type I error; II., type II error; EA, mean class accuracy estimate; MS, mapping score; RM, root mean square proportion estimation error; CI, comparison index; GCE (LCE), Global (Local) Consistency Error.

	Benchmark – Colour						
	TFR+	TFR	AR3D	GMRF	JSEG	Blobworld	EDISON
↑ CS	51.25	46.13	37.42	31.93	27.47	21.01	12.68
↓ OS	5.84	2.37	59.53	53.27	38.62	7.33	86.91
↓ US	7.16	23.99	8.86	11.24	5.04	9.30	0.00
↓ ME	31.64	26.70	12.54	14.97	35.00	59.55	2.48
↓ NE	31.38	25.23	13.14	16.91	35.50	61.68	4.68
↓ O	23.60	27.00	35.19	36.49	38.19	43.96	68.45
↓ C	22.42	26.47	11.85	12.18	13.35	31.38	0.86
↑ CA	67.45	61.32	59.46	57.91	55.29	46.23	31.19
↑ CO	76.40	73.00	64.81	63.51	61.81	56.04	31.55
↑ CC	81.12	68.91	91.79	89.26	87.70	73.62	98.09
↓ I.	23.60	27.00	35.19	36.49	38.19	43.96	68.45
↓ II.	4.09	8.56	3.39	3.14	3.66	6.72	0.24
↑ EA	75.80	68.62	69.60	68.41	66.74	58.37	41.29
↑ MS	65.19	59.76	58.89	57.42	55.14	40.36	31.13
↓ RM	6.87	7.57	4.66	4.56	4.62	7.52	3.09
↑ CI	77.21	69.73	73.15	71.80	70.27	61.31	50.29
↓ GCE	20.35	15.52	12.13	16.03	18.45	31.16	3.55
↓ LCE	14.36	12.03	6.69	7.31	11.64	23.19	3.44

For the sake of brevity we do not show here the segmentation maps, which can be found on the benchmark web site [7] as well, but just the numerical results summarized in Tab.1. The interpretation of these indicators may seem quite ambiguous since (of course) no algorithm outperforms uniformly all the other ones. However it can be easily recognized that the two versions of TFR seem to outperform the other ones w.r.t. many indicators, with TFR+ being generally better than TFR. In particular, the main drawback of the reference methods is the tendency to over-segment while, on the contrary, only the TFR has a tendency to under-segment. In this regard, the best trade-off is reached by the TFR+, which outperforms TFR and can be considered as the best one.

5 Application to Remotely Sensed Data

Finally, in this section an application of the proposed method to remote-sensing data is presented. We worked on high resolution (50cm) aerial images covering wooded areas, which match well with the proposed modeling since they present different relevant texture patterns with acceptable stationarity. Such images are courtesy of the “French Forest Inventory” (IFN).

We present two experiments. The former, see Fig.3, refers to an area composed of several classes of trees plus no tree lands and shadows. Since we have no ground-truth related to these data, we build up the latter experiment where a mosaic image was obtained which is composed of four square subimages, see Fig.4. Three of them represent different quasi stationary tree textures, while the last one (bottom-left) is a mixing of an urban class and one of the other (bottom-right) tree textures.

We experimented only the case of TFR+, since it has been shown to be better than TFR in the previous section. Also no comparative algorithms have yet been tested on these data, and eventually we can only make conjectures about the performances of TFR+. A comparison with another method currently under development could be made later.

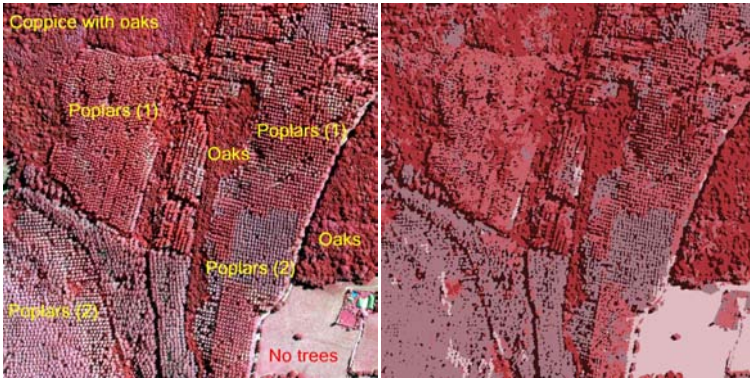


Fig. 3. Left: Forest image, south of Bourgogne, France. ©IFN. Right: Segmentation map obtained by the TFR+ algorithm (5 classes: two kinds of poplars, oaks, no trees, and shadows).

The 1024×1024 forest image and the associated 5-class TFR+’s segmentation are shown in Fig.3. One class represents just the shadows, one is associated with low vegetation areas, the remaining three classes correspond to different tree patterns. The segmentation seems to be quite promising according to a visual inspection. Indeed, in order to obtain such good result, a slight modification of the TFR+ algorithm was necessary. In fact, the proposed optimization schemes (meaning both TFR and TFR+) are sensitive to the presence of continuous regions, like background colors, because these are typically large and, hence, work

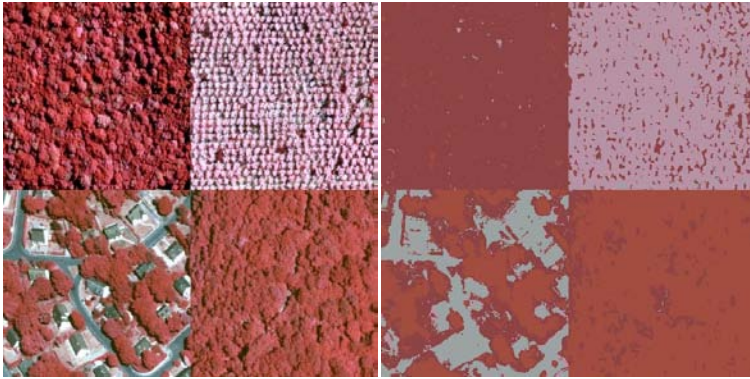


Fig. 4. Left: Mosaic of different kinds of remotely sensed forest patterns, south of Bourgogne ©IFN. Right: Segmentation map obtained by the TFR+ (4 classes).

as collectors of other regions. This becomes a critical problem when different textures have the same background color and share a long contour, where we can find many of such regions which cross the border and, therefore, link the textures forcing a merging. Unfortunately this was the case of the shadow regions present in the image (see Fig.3, left hand side). For this reason we decided to simply detect the background regions (just the shadows, in this case) after the CBC step, and ignore them in the subsequent steps (SBC and region merging).

Instead, in the latter experiment such modification was not necessary. The results are encouraging in this case as well. In particular, from the segmentation shown in Fig.4, we can see that the three different tree patterns have been detected with satisfactory precision. As for the mixed urban-tree area (bottom-left), the urban elements are assigned with a fourth class, while the trees are largely assigned with the correct tree class (that at bottom-right).

6 Conclusion

In this work we have presented a novel texture model which is particularly suited for the task of image segmentation in an unsupervised framework. The model aims at describing each texture at multiple scales through a region-based hierarchical approach which allows a very simple, but effective, segmentation scheme (the TFR) which processes color and spatial information in two independent steps, as to obtain an image decomposition in texture states. Finally a region merging procedure allows us to properly relate the states hierarchically, and single out the textured regions.

Numerical results proved the superior performance of the proposed method w.r.t. to other algorithms on the Prague benchmark data. Encouraging results have been obtained as well on satellite images. Future research will be focused on the replacement of k -means at SBC layer with a more effective clustering method.

Acknowledgments. This work was carried out during the tenure of an ERCIM fellowship (Scarpa’s postdoc), and supported by EU MUSCLE project (e-team: shape modelling), FP6-507752, and partially by the project 1ET400750407. The authors would also like to thank the “French Forest Inventory” (IFN) for providing the remotely sensed data covering the forest areas.

References

1. Carson, C., Thomas, M., Belongie, S., Hellerstein, J.M., Malik, J.: Blobworld: A system for region based image indexing and retrieval. In: 3th ICVIS, Amsterdam, The Netherlands, pp. 509–516. Springer, Heidelberg (1999)
2. D’Elia, C., Poggi, G., Scarpa, G.: A Tree-Structured Markov random field model for Bayesian image segmentation. *IEEE Transactions on Image Processing* 12(10), 1259–1273 (2003)
3. Haindl, M., Mikeš, S.: Colour texture segmentation using modelling approach. In: Singh, S., Singh, M., Apte, C., Perner, P. (eds.) ICAPR 2005. LNCS, vol. 3687, pp. 484–491. Springer, Heidelberg (2005)
4. Poggi, G., Scarpa, G., Zerubia, J.: Supervised segmentation of remote-sensing images based on a tree-structured MRF model. *IEEE Transactions on Geoscience and Remote Sensing* 43(8), 1901–1911 (2005)
5. Beaulieu, J.-M., Touzi, R.: Segmentation of textured polarimetric SAR scenes by likelihood approximation. *IEEE Transactions on Geoscience and Remote Sensing* 42(10), 2063–2072 (2004)
6. S.Hese. Segmentation of forest stands in very high resolution stereo data. In Proc. IGARSS’01, Sydney (AUS), July 2001, vol.4, pp.1654–1656 (2001)
7. Mikeš, S., Haindl, M.: Prague texture segmentation data generator and benchmark. *ERCIM News* 64, 67–68 (2006) <http://mosaic.utia.cas.cz>.
8. Haindl, M., Mikeš, S.: Model-based texture segmentation. In: Campilho, A., Kamel, M. (eds.) ICIAR 2004. LNCS, vol. 3212, pp. 306–313. Springer, Heidelberg (2004)
9. Deng, Y., Manjunath, B.S.: Unsupervised Segmentation of Color-Texture Regions in Images and Video. *IEEE Transactions on Pattern Analysis Machine Intelligence* 23(8), 800–810 (2001)
10. Christoudias, C.M., Georgescu, B., Meer, P.: Synergism in Low Level Vision. In: proc. 16th ICPR, August 2002, vol.4, pp.150–155, Los Alamitos (2002)
11. Fischer, B., Thies, C.J., Guld, M.O., Lehmann, T.M.: Content-based image retrieval by matching hierarchical attributed region adjacency graphs. In: Proc. SPIE, vol. 5370, pp. 598–606, San Diego, CA (USA) (2004)
12. Duda, R.O., Hart, P.E., Stork, D.G.: *Pattern Classification*, 2nd edn. Wiley, Chichester (2000)
13. Scarpa, G., Haindl, M.: Unsupervised Texture Segmentation by Spectral-Spatial-Independent Clustering. In: Proc. 18th ICPR, Hong Kong (China), August 2006, vol.2, pp.151–154 (2006)
14. Penny, W.D.: Kullback-Leibler divergences of normal, Gamma, Dirichlet and Wishart densities. Technical report, Wellcome Department of Imaging Neuroscience, University College London (UK) (2001)