Probabilistic Discrete Mixtures Colour Texture Models

Michal Haindl, Vojtěch Havlíček, and Jiří Grim

Institute of Information Theory and Automation
of the ASCR, Prague, Czech Republic

Abstract. A new generative multispectral texture model based on discrete distribution mixtures is introduced. Statistical texture properties are represented by a discrete distribution mixture of product components. A natural colour or multispectral texture is spectrally factorized and discrete mixtures models are learned and used to synthesize single orthogonal monospectral components. Texture synthesis is based on easy computation of arbitrary conditional distributions from the model. Finally single synthesized monospectral texture components are transformed into the required synthetic colour texture. This model can easily serve for texture segmentation, retrieval or to model single factors in complex Bidirectional Texture Function (BTF) space models. The advantages and weak points of the presented approach are demonstrated on several colour texture applications.

Keywords: Discrete distribution mixtures, EM algorithm, texture modeling.

1 Introduction

Realistic texture models are crucial for various image recognition or modeling applications such as image classification, segmentation, content based image retrieval (CBIR), image enhancement or restoration. Virtual or augmented reality systems require object surfaces covered with realistic nature-like colour textures to enhance realism in virtual scenes. Such textures can be either digitized natural textures or textures synthesized from an appropriate mathematical model. However digitized 3D textures are far less convenient alternative, because of extremal system memory demands especially for the most advanced Bidirectional texture function (BTF) representation [1] and several other serious drawbacks.

Texture synthesis methods may be divided primarily into intelligent sampling and model-based methods. Intelligent sampling approaches [2], [3], [4], [5],[6] rely on sophisticated sampling from real texture measurements while the model-based techniques [7],[8],[9], [10], [11],[12], [13], [14] describe texture data using multidimensional mathematical models and their synthesis is based on the estimated model parameters only. There are several texture modeling approaches published [7], [11], [12] and some survey articles are also available [10], [15].

In the present paper we propose texture modeling by multivariate discrete mixtures (DM) with components defined as products of univariate discrete probability distributions. Each of the univariate component-specific distributions is defined simply by a vector of probabilities without any constraint. The full generality of the univariate distributions in the components is one of the strong arguments for the proposed mixture model. Another motivation is the computational simplicity of the product mixtures. In particular, the texture synthesis is actually enabled by an easy evaluation of arbitrary

J. Ruiz-Shulcloper and W.G. Kropatsch (Eds.): CIARP 2008, LNCS 5197, pp. 675–682, 2008.

marginals and simple computation of the conditional probability distributions. In the application part we demonstrate advantages and weak points of the proposed method on several colour textured images.

2 Distribution Mixture Model

Regular or near-regular textures are very difficult to synthesize using statistical models such as Markov random field (MRF) model family, however they occur frequently in images as visual representation of man-made structures. The presented model is well suited for them and as such it can serve as appropriate submodel for more complex BTF models among others. Modeling general static colour texture images requires three-dimensional models or to accept some spectral information loss using a set of factorized less-dimensional 2D probabilistic models. The factorization alternative accepted in this paper is attractive because it allows using simpler 2D data models with less parameters (one third in the three-spectral case of colour MRF textures).

A digitized texture image Y is assumed to be defined on a finite rectangular $N \times O \times d$ lattice I, $r = \{r_1, r_2, r_3\} \in I$ denotes a pixel multiindex with the row, columns and spectral indices, respectively. The notation $\bullet$ has the meaning of all possible values of the corresponding index.

Supposing now uncorrelated monospectral textures after the PCA based decorrelation step of our algorithm, we assume that each pixel of the image is described by a grey level taking on K possible values, i.e., $Y_r \in \mathcal{K}, \quad \forall r \in I, \quad \mathcal{K} = \{1, 2, \ldots, K\}$, where $\mathcal{K}$ is the set of distinguished grey levels (often $|\mathcal{K}| = 256$). In this sense a monospectral component of the original texture image can be viewed as a vector $Y_{\bullet,\bullet,r_3} \in \mathcal{K}^{NO}$, in some chosen pixel ordering. To simplify notation we will neglect further on the spectral component in the multiindices r, s because single submodels describe only decorrelated mono-spectral components of the original multi-spectral texture. Let us suppose that the natural homogeneous texture image represents a realization of a random vector with a probability distribution $P(Y_{\bullet,\bullet,r_3})$ and that the properties of the texture can be fully characterized by statistical dependencies on a subfield, i.e., by a marginal probability distribution of grey levels on pixels within the scope of a window centered around the location r and specified by the index set I_r. $I_r = \{r + s : |r_1 - s_1| \leq \alpha \wedge |r_2 - s_2| \leq \beta\} \subset I$ where α, β are some chosen constants and $|.|$ is the absolute value. If we denote $Y_{\{r\}}$ the corresponding subvector of $Y_{\bullet,\bullet,r_3}$ containing all Y_s such that $s \in I_r$, $Y_{\{r\}} = [Y_s \; \forall s \in I_r]$, $\eta = card\{I_r\}$ and $P(Y_{\{r\}})$ the corresponding marginal distribution of $P(Y)$ then the marginal probability distribution on the "generating" window I_r is assumed to be invariant with respect to arbitrary shifting within the original image, i.e., $P(Y_{\{r\}}) = P(Y_{\{s\}})$, $\quad \forall s, r \in I, s \neq r$.

Thus, e.g., for a rectangular window of the size $\eta = 20 \times 20$ pixels we have to estimate a 400-dimensional probability distribution $P(Y_{\{r\}})$. The marginal distribution $P(Y_{\{r\}})$ is assumed to contain sufficient information to synthesize the modeled texture. The distribution $P(Y_{\{r\}})$ is assumed to be discrete with factorizing components $P(Y_{\{r\}} \,|\, m)$ in the form:

$$P(Y_{\{r\}}) = \sum_{m \in \mathcal{M}} p(m)\, P(Y_{\{r\}} \,|\, m) = \sum_{m \in \mathcal{M}} p(m) \prod_{s \in I_r} p_s(Y_s \,|\, m) \; . \tag{1}$$

$Y_{\{r\}} \in \mathcal{K}^\eta$ $\mathcal{M} = \{1, 2, \ldots, M\}$ where $p(m)$ are probability weights. $p_s(Y_s \,|\, m)$ are univariate discrete (component-specific) probability distributions. It can be seen that, by Eq. (1) the variables $\{Y_s : \forall s \in I_r\}$ are conditionally independent with respect to the index variable m. From the theoretical point of view, this assumption is not restrictive. It can be easily verified that, in discrete case $Y_{\{r\}} \in \mathcal{K}^\eta$, the class of finite mixtures (1) is complete in the sense that any discrete probability distribution on $\mathcal{K}^\eta$ can be expressed in the form (1) for M sufficiently large. The parameters of the mixture model (1) are probabilistic component weights $p(m)$ and the univariate discrete distributions of grey levels simply defined by a vector of probabilities:

$$p_n(\cdot \,|\, m) = (p_n(1 \,|\, m), p_n(2 \,|\, m), \ldots, p_n(K \,|\, m)) \ . \quad (2)$$

The total number of mixture (1) parameters is thus $card\{M\}(1 + \eta K)$ - confined to the appropriate norming conditions. Note that the form of the univariate discrete distributions (2) is fully general without any constraint. In contrast to different parametric models (e.g., normal) the K-dimensional vector $p_n(\cdot \,|\, m)$ can describe arbitrary discrete distribution. This fact is one of the main arguments for the choice of the discrete mixture model (1). Another strong motivation for the conditional independence model (1) is a simple switch-over to any marginal distribution by deleting superfluous terms in the products $P(Y_{\{r\}} \,|\, m)$.

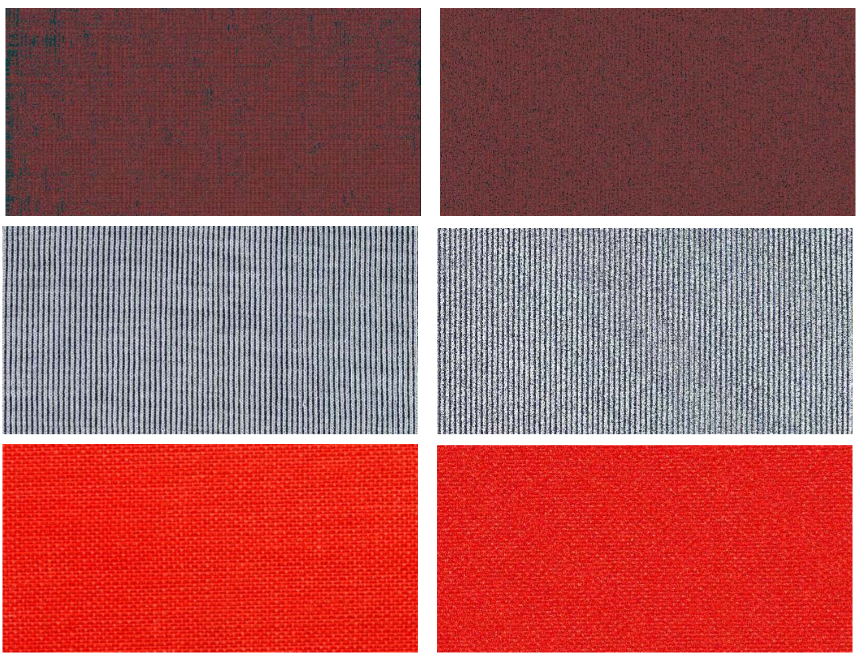

Fig. 1. Natural and synthetic (right) textile textures

3 EM Algorithm

The underlying structural model of conditional independence is identified from a data set $\mathcal{S}$ obtained by step-wise shifting the contextual window I_r within the original texture image, i.e., for each location r one realization of $Y_{\{r\}}$.

$$\mathcal{S} = \{Y_{\{r\}} \; \forall r \in I, \, I_r \subset I\} \quad Y_{\{r\}} \in \mathcal{K}^\eta \; . \tag{3}$$

The unknown parameters of the approximating mixture can be estimated by means of the iterative EM (Expectation Maximization) algorithm [16], [17]. In order to estimate the unknown distributions $p_n(\cdot \,|\, m)$ and the component weights $p(m)$ we maximize the likelihood function corresponding to (3)

$$L = \frac{1}{|\mathcal{S}|} \sum_{Y_{\{r\}} \in \mathcal{S}} \log \, [\sum_{m \in \mathcal{M}} P(Y_{\{r\}} \,|\, m) \, p(m)] \tag{4}$$

by means of the EM algorithm. The related iteration equations can be expressed as follows:

$$q^{(t)}(m | \, Y_{\{r\}}) = \frac{P^{(t)}(Y_{\{r\}} \,|\, m) \, p^{(t)}(m)}{\sum_{j \in \mathcal{M}} P^{(t)}(Y_{\{r\}} \,|\, j) \, p^{(t)}(j)}, \tag{5}$$

$$p^{(t+1)}(m) = \frac{1}{|\mathcal{S}|} \sum_{Y_{\{r\}} \in \mathcal{S}} q^{(t)}(m \,| \; Y_{\{r\}}), \tag{6}$$

$$p_n^{(t+1)}(\xi \,|\, m) = \frac{1}{|\mathcal{S}| p^{(t+1)}(m)} \sum_{Y_{\{r\}} \in \mathcal{S}} \delta(\xi, Y_n) \, q^{(t)}(m \,|\, Y_{\{r\}}), \quad \xi \in \mathcal{K}. \tag{7}$$

The mixture parameters are initialized by random numbers. The iteration process is stopped when the criterion increments are sufficiently small. Note that a larger number of grey levels increases the memory requirements but not necessarily the computing time (see (7)). It is well known (cf. [17]) that the iteration scheme (5) – (7) has the following monotonic property: $L^{(t+1)} \geq L^{(t)}, \quad t = 0, 1, 2, \ldots$ which implies the convergence of the sequence $\{L^{(t)}\}_0^\infty$ to a stationary point of EM algorithm (local extremum or a saddle point of L). It should be noted that the properties of the ML estimates obtained by means of the data set $\mathcal{S}$ may be negatively influenced by the fact that the observations in $\mathcal{S}$ are not independent.

4 Texture Synthesis

Let I_r be a fixed position of the generating window. If $Y_{\{\rho\}} \subset Y_{\{r\}}$ is a subvector of all pixels previously specified within this window and $\rho \subset I_r$ the corresponding index subset, then the statistical properties of the remaining unspecified variables are fully described by the corresponding conditional distribution. In view of the advantageous properties of our mixture model we can easily compute any univariate conditional distribution $p_{n \,|\, \rho}$:

$$p_{n \,|\, \rho}(Y_n \,|\, Y_{\{\rho\}}) = \sum_{m=1}^{M} W_m(Y_{\{\rho\}}) \, p_n(Y_n \,|\, m) \;\; , \tag{8}$$

where $W_m(Y_{\{\{\rho\}})$ are the a posteriori component weights corresponding to the given subvector $Y_{\{\rho\}}$:

$$W_m(Y_{\{\rho\}}) = \frac{p(m)\, P_\rho(Y_{\{\rho\}} \,|\, m)}{\sum_{j=1}^{M} p(j)\, P_\rho(Y_{\{\rho\}} \,|\, j)} \ , \tag{9}$$

$$P_\rho(Y_{\{\rho\}} \,|\, m) = \prod_{n \in \rho} p_n(Y_n \,|\, m) \ .$$

The grey level y_n can be randomly generated by means of the conditional distribution $p_{n|C}(y_n|Y_{\{\rho\}})$ whereby Eqs. (8) can be applied to all the unspecified variables $n = \eta - \ card\{\rho\}$ given a fixed position of the generating field. Simultaneously, each newly generated grey level y_n can be used to upgrade the conditional weights $W_m(Y_{\{\rho\}})$. In the next step, the generating field is shifted to a new position and the conditional distribution (8) has to be computed for a new subset of the specified pixels in ρ. In our experiments we have used a regular left-to-right and top-to-down shifting of the generating window.

Single mixture models (1) synthesize single decorrelated monospectral components and the resulting synthesized colour texture is obtained from the set of synthesized monospectral images inverting the decorrelation process.

5 Experimental Results

The implementation of EM algorithm is simple but there are some well known computational problems, e.g., the proper choice of the number of components, the existence of local maxima of the likelihood function an the related problem of a proper choice of the initial parameter values. The above difficulties are less relevant if the sample size is sufficiently large. In our case the dimension of the estimated distribution is not too high ($N \approx 10^1 - 10^2$) and the number of the training data vectors relatively large ($|\mathcal{S}| \approx 10^4 - 10^5$). The number of grey levels to be distinguished is very high (usually $|\mathcal{K}| = 256$) and therefore the estimated distribution becomes considerably complex. Moreover, the shifting window seems to produce rather "flat" probability distributions,

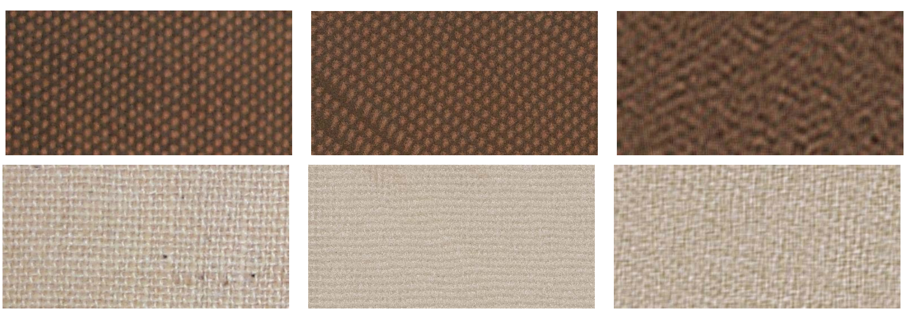

Fig. 2. Natural (left) and synthetic (DM middle) carpet (upper row) and jute (bottom row) textures compared with their synthetic (right column) alternatives generated using the Gaussian Markov random field model

Fig. 3. Natural and synthetic (DM middle, GMRF synthesis right) rattan textures

especially in case of homogeneous structures. For these reasons the generating window should always be kept reasonably small and the sample size as large as possible. The application of the model on deterministic periodic textures (e.g., chessboard, stripes) is capable to reproduce original textures exactly and the EM algorithm converges quickly from arbitrary starting conditions.

The examples Figs. 1 - 3 exemplify properties of the DM model on natural colour textures. The carpet texture on Fig. 2 represents relatively regular texture which is notoriously difficult for some alternative texture models like for example Gaussian Markov random field (GMRF) models (Fig. 2 - top right) but the presented model produced very good synthesis result (Fig. 2 - top middle). Similarly the jute example (Fig. 2 - bottom) or the textile textures (Fig. 1) demonstrate its good performance. The DM model for the rattan Fig. 3 - left expressed major periodic features and although it failed to specify rattan details, it is still much superior to its GMRF alternative (Fig. 3 - top right). Finally the last example on Fig. 3-bottom demonstrates clear failure of our model (the most informative monospectral component presented only), there is a strong tendency to cover a large portion of the synthesized field by a texture resembling "white noise". It appears that, because of a high dimensionality of the underlying space, the estimated mixture distribution has properties resembling widely separated components. This observation relates to the well known experience that high dimensional spaces can be viewed as "sparse". The isolated peaks of the estimated mixture seem to be able to reflect only the basic rattan structure. Consequently, in most cases the neighborhood of the synthesized pixel is untypical from the point of view of the estimated mixture and therefore the corresponding conditional distribution (8) is very flat or nearly uniform. In such a case the synthesis produces a texture resembling white noise. Theoretically this modeling result could be improved if we would have larger trainee image to train a DM model parameters but on the other hand, the sufficient size of the trainee image could already be prohibitive from the computational point of view. All DM models used the contextual window size 21×21 pixels, the training set of $|\mathcal{S}| = 262144$ vectors. The chosen distribution mixture model included $M = 40$ components and about 10 iterations of EM algorithm were needed to achieve a reasonable convergence. The computation was rather time-consuming it took several hours in total on HP workstations. The time needed for texture synthesis is comparable with one iteration step of the EM algorithm. Resulting textures can be further slightly improved by iterating the synthesis procedure or by our probabilistic synthesis strategy described elsewhere. Similarly as all other known texture models also our DM model has its strong as well as weak sides. While the presented model can realistically synthesize natural or man-made textures with strong periodicities, which are notoriously difficult for most alternative approaches, its major weakness is lesser robustness than the Markovian models family. A DM model has strong tendency either to produce high quality synthetic texture or to completely fail with resulting noise field. Markovian models in these cases demonstrate clear effort to grasp at lest some of the difficult texture features (Fig. 3 - bottom right). The computationally most efficient Markovian models are much faster than the presented model, but general Markovian models which require Markov chain Monte Carlo methods for their analysis as well as synthesis are comparable.

6 Conclusion

The proposed DM model is the only statistical model capable to synthesize regular or near-regular colour textures. DM representation can be simultaneously used also in any texture based recognition task such as classification, segmentation, image retrieval, etc. Moreover, the DM model can serve as the underlying factor model for more complex BTF space models. The application of EM algorithm to texture modeling has some specific features. Generally the dimension of the sample space is relatively high ($N = 200 - 400$) and the corresponding sample size appears to be insufficient. Moreover, the data vectors obtained by shifting the window are not independent as it is assumed in the likelihood criterion. For these and other reasons the estimation of the texture model in the form of discrete product mixture is a difficult task. Our extensive DM models simulations suggest that very often the model requires a large training data set and powerful computing resources to successfully reproduce any given natural texture. While the computational complexity is going to be less important in near future, the requirement for large learning data set can be restrictive in some texture modeling applications.

Acknowledgements

This research was supported by the projects no. GAAV 1ET400750407, GAČR 102/08/0593, 102/07/1594 and partially by the MŠMT grants 1M0572 DAR, 2C06019.

References

1. Haindl, M., Filip, J.: Extreme compression and modeling of bidirectional texture function. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(10), 1859–1865 (2007)
2. Efros, A.A., Leung, T.K.: Texture synthesis by non-parametric sampling. In: Proc. Int. Conf. on Computer Vision (2), Corfu, Greece, pp. 1033–1038 (1999)
3. Efros, A.A., Freeman, W.T.: Image quilting for texture synthesis and transfer. In: Fiume, E. (ed.) ACM SIGGRAPH 2001, pp. 341–346. ACM Press, New York (2001)

4. Heeger, D., Bergen, J.: Pyramid based texture analysis/synthesis. In: ACM SIGGRAPH 1995, pp. 229–238. ACM Press, New York (1995)
5. Dong, J., Chantler, M.: Capture and synthesis of 3d surface texture. In: Texture 2002, vol. 1, pp. 41–45. Heriot-Watt University (2002)
6. Zelinka, S., Garland, M.: Towards real-time texture synthesis with the jump map. In: 13th European Workshop on Rendering, pp. 99–104 (2002)
7. Bennett, J., Khotanzad, A.: Multispectral random field models for synthesis and analysis of color images. IEEE Trans. on Pattern Anal. and Machine Int. 20(3), 327–332 (1998)
8. Bennett, J., Khotanzad, A.: Maximum likelihood estimation methods for multispectral random field image models. IEEE Transactions on Pattern Analysis and Machine Intelligence 21(6), 537–543 (1999)
9. Gimelfarb, G.: Image Textures and Gibbs Random Fields. Kluwer Academic Pub., Dordrecht (1999)
10. Haindl, M.: Texture synthesis. CWI Quarterly 4(4), 305–331 (1991)
11. Haindl, M., Havlíček, V.: Multiresolution colour texture synthesis. In: Dobrovodský, K. (ed.) Proceedings of the 7th International Workshop on Robotics in Alpe-Adria-Danube Region, Bratislava, ASCO Art, pp. 297–302 (1998)
12. Haindl, M., Havlíček, V.: A multiresolution causal colour texture model. In: Amin, A., Pudil, P., Ferri, F.J., Iñesta, J.M. (eds.) SPR 2000 and SSPR 2000. LNCS, vol. 1876, pp. 114–122. Springer, Heidelberg (2000)
13. Paget, R., Longstaff, I.D.: Texture synthesis via a noncausal nonparametric multiscale markov random field. IEEE Trans. on Image Processing 7(8), 925–932 (1998)
14. Zhu, S., Liu, X., Wu, Y.: Exploring texture ensembles by efficient markov chain monte carlo - toward a "trichromacy" theory of texture. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(6), 554–569 (2000)
15. Haindl, M.: Texture modelling. In: Sanchez, B., Pineda, J.M., Wolfmann, J., Bellahse, Z., Ferri, F. (eds.) Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics, Orlando, USA, vol. VII, pp. 634–639. International Institute of Informatics and Systemics (2000)
16. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the em algorithm. Journal of the Royal Statistical Society, B B 39, 1–38 (1977)
17. Schlesinger, M.: Relation between learning and self-learning in pattern recognition (in Russian). Kibernetika (2), 81–88 (1968)
18. Liang, L., Liu, C., Xu, Y.Q., Guo, B., Shum, H.Y.: Real-time texture synthesis by patch-based sampling. ACM Transactions on Graphics (TOG) 20(3), 127–150 (1968)
19. Liu, F., Picard, R.: Periodicity, directionality, and randomness: Wold features for image modeling and retrieval. IEEE Transactions on Pattern Analysis and Machine Intelligence 18(7), 722–733 (1996)
20. Portilla, J., Simoncelli, E.: A parametric texture model based on joint statistics of complex wavelet coefficients. International Journal of Computer Vision 40(1), 49–71 (2000)
21. Tong, X., Zhang, J., Liu, L., Wang, X., Guo, B., Shum, H.Y.: Synthesis of bidirectional texture functions on arbitrary surfaces. ACM Transactions on Graphics (TOG) 21(3), 665–672 (2002)