A Gaussian Mixture-Based Colour Texture Model

M. Haindl, J. Grim, P. Somol, P. Pudil
Institute of Information Theory and Automation
Academy of Sciences CR, Prague, Czech Republic
{haindl,grim,somol,pudil}@utia.cas.cz

M. Kudo
Mineichi Kudo
Graduate School of Engineering,
Hokkaido University, Sapporo, Japan
mine@main.eng.hokudai.ac.jp

Abstract

A new method of colour texture modelling based on Gaussian distribution mixtures is discussed. We estimate the local statistical properties of the monospectral version of the target texture in the form of a Gaussian mixture of product components. The synthesized texture is obtained by means of a step-wise prediction of the texture image. In order to achieve a realistic colour texture image and to avoid possible loss of high-frequency details we use optimally chosen pieces of the original colour source texture in the synthesis phase. In this sense the proposed texture modelling method can be viewed as a statistically controlled sampling. By using multispectral or mutually registered BTF texture pieces the method can be easily extended also for these textures.

1. Introduction

Virtual or augmented reality systems in computer graphics applications (e.g., computer games, CAD systems) require object surfaces covered with realistic nature-like colour textures to enhance realism in virtual scenes. The objects can be covered by digitized 3D natural textures, however, this technique is inconvenient because of extremal memory demands, visible discontinuities and several other drawbacks. Textures can be alternatively synthesized artificially. The related methods may be divided primarily into intelligent sampling- and model-based methods. Intelligent sampling approaches [3], [6], [5], [15], [18], [4], [19] rely on sophisticated sampling from real texture measurements while the model-based techniques [1], [2], [7], [10], [12], [14], [17], [20] describe texture data by using multidimensional mathematical models and their synthesis is based on the estimated model parameters only.

There are several texture modelling approaches published [16], [12], [14] and some survey articles are also available [10], [11]. Most published texture models are restricted only to monospectral textures, for few models developed for multispectral (mostly colour) textures refer [1], [12], [14]. We introduced in our previous paper [12] a fast multiresolution Markov random field (MRF) based model and the simultaneous causal autoregressive random field model [13]. Although the former method avoids the time consuming Markov chain Monte Carlo simulation, it requires several approximations. The later method is very efficient for colour texture modelling not only because it does not suffer with some problems of alternative options (see [10], [11] for details) but it is also easy to analyze as well as to synthesize. Last but not least it is still flexible enough to imitate a large set of natural and artificial textures. However model-based methods are usually extremely compressed approximations of real measurements and as such they sometimes compromise visual realism more than prevailing intelligent sampling type of methods.

We have proposed recently a novel approach to texture modelling based on approximating local statistical properties of textures within a small observation window by means of distribution mixtures [8], [9]. The method largely benefits from the advantageous computational properties of product mixtures. The mixture parameters can be estimated by the means of the EM algorithm and, on the other hand, the texture synthesis is easily achievable via computing conditional probabilities. We have assumed rather general local statistical model by considering the mixture components to be defined as products of univariate discrete probability distributions in the original space. In our experiments we have succeeded to synthesize realistic colour textures with strong periodicity which are notoriously difficult for most of the alternative model-based approaches. However, in many cases the mixture based synthesis failed completely with the resulting noise field. It appears that for less periodical textures the available learning data sets are not large enough to enable a reliable estimation of the high-dimensional discrete distribution mixture. The problem cannot be solved simply by increasing the original source texture image because of exceedingly time-consuming computation.

In the present paper we propose texture modelling by using mixtures of multivariate Gaussian densities with diagonal covariance matrices. In this way the number of the estimated mixture parameters reduces considerably and the numerical stability of the synthesis procedure correspondingly improves. The Gaussian mixture model is estimated from the monospectral projection of the original colour texture image. The estimated component means correspond to the typical variants of the texture pieces occurring in the observation window, however, they are monospectral and smoothed. In order to obtain the colour texture with all high frequency details the component means are replaced by the most similar pieces of source colour texture in the synthesis phase.

2. Normal Mixture Model

We assume a mixture model to represent texture frequency (i.e. to control sampling process) but not to represent any spectral information. Hence it is enough to use simpler monospectral normal mixture model. For the sake of our method let us consider first a digitized gray-scale texture image Y on a finite rectangular $I \times J$ lattice with the row and columns indices $i = 1, 2, \ldots, I;\ j = 1, 2, \ldots, J$, respectively. We assume that each pixel of the image is described by a grey level taking on K possible values, i.e., $Y_{i,j} \in \mathcal{K}$ where $\mathcal{K} = \{1, 2, \ldots, K\}$ is the set of distinguished grey levels (often $|\mathcal{K}| = 256$).

The concept of texture intuitively implies some degree of homogeneity. In other words we may assume that the local statistical properties of a texture as observed, e.g., within a small moving window should be invariant with respect to the window position. In this sense we can describe the statistical properties of interior pixels of the moving window by a joint probability distribution. Denoting by $\mathrm{x} = (x_1, x_2, \ldots, x_N) \in \mathcal{X},\ \ \mathcal{X} = \mathcal{K}^N$ the vector of interior pixels of the observation window we assume the joint probability distribution $P(\mathrm{x})$ in the form of a normal mixture

$$\begin{aligned} P(\mathrm{x}) &= \sum_{m\in\mathcal{M}} w_m F(\mathrm{x}|\mu_m,\sigma_m), \quad \mathrm{x}\in\mathcal{X} \ , \qquad (1) \\ \mathcal{M} &= \{1,2,\ldots,M\}, \quad \mathcal{N} = \{1,2,\ldots,N\} \end{aligned}$$

where w_m are probability weights, $\mathcal{M}, \mathcal{N}$ the index sets and the mixture components are defined as products of univariate Gaussian densities

$$\begin{aligned} F(\mathrm{x}|\mu_m,\sigma_m) &= \prod_{n\in\mathcal{N}} f_n(x_n|\mu_{mn},\sigma_{mn}) \ , \qquad (2) \\ f_n(x_n|\mu_{mn},\sigma_{mn}) &= \frac{1}{\sqrt{2\pi}\sigma_{mn}} \exp\{-\frac{(x_n-\mu_{mn})^2}{2\sigma_{mn}^2}\} \ , \end{aligned}$$

i.e., the components are multivariate Gaussian densities with diagonal covariance matrices.

Obviously, assuming the Gaussian densities (2), we ignore the discrete nature of the variables x_n (typically $x_n \in \{0, 1, \ldots, 255\}$). On the other hand we need only 2 parameters to specify the density $f_n(x_n|\mu_{mn}, \sigma_{mn})$ in contrast to 255 parameters to be specified in case of a general discrete distribution $f_n(x_n|m)$ as used in [9].

The maximum-likelihood estimates of the parameters $w_m, \mu_{mn}, \sigma_{mn}$ can be computed by the means of the EM algorithm [9]. In particular we use a data set $\mathcal{S}$ obtained by pixel-wise shifting the observation window within the original texture image

$$\mathcal{S} = \{\mathrm{x}^{(1)}, \ldots, \mathrm{x}^{(K)}\}, \ \ \mathrm{x}^{(k)} \in \mathcal{X}. \qquad (3)$$

The corresponding log-likelihood function

$$L = \frac{1}{|\mathcal{S}|} \sum_{x\in\mathcal{S}} \log\,[\sum_{m\in\mathcal{M}} w_m F(\mathrm{x}|\mu_m,\sigma_m)] \qquad (4)$$

is maximized by the EM algorithm $(m \in \mathcal{M}, n \in \mathcal{N}, \mathrm{x} \in \mathcal{S})$

$$\begin{aligned} q(m|\mathrm{x}) &= \frac{w_m F(\mathrm{x}|\mu_m,\sigma_m)}{\sum_{j\in\mathcal{M}} w_j F(\mathrm{x}|\mu_j,\sigma_j)} \ , \qquad (5) \\ w_m^{'} &= \frac{1}{|\mathcal{S}|} \sum_{x\in\mathcal{S}} q(m|\mathrm{x}) \ , \qquad (6) \\ \mu_m^{'} &= \frac{1}{\sum_{x\in\mathcal{S}} q(m|\mathrm{x})} \sum_{x\in\mathcal{S}} x_n q(m|\mathrm{x}) \ , \qquad (7) \\ (\sigma_m^{'})^2 &= -(\mu_m^{'})^2 + \frac{\sum_{x\in\mathcal{S}} x_n^2 q(m|\mathrm{x})}{\sum_{x\in\mathcal{S}} q(m|\mathrm{x})} \ . \qquad (8) \end{aligned}$$

Here the apostrophe denotes the new parameter values.

Let us remark that in practical experiments the resulting parameters μ_m are visually well interpretable. They correspond to the typical (smoothed) variants of the monospectral texture pieces occurring in the observation window. 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 is relatively large ($|\mathcal{S}| \approx 10^4 - 10^5$). Nevertheless the window should always be kept reasonably small and the sample size as large as possible.

3. Texture Synthesis

The statistical description of the local texture properties naturally suggests the possibility of texture synthesis by local prediction. We assume that in a general situation, at a given position of the observation window, some part of the synthesized texture is already specified. If we denote

$$\begin{aligned} \mathrm{x}_{\mathcal{C}} &= (x_{n_1}, x_{n_2}, \ldots, x_{n_l}) \in \mathcal{X}_{\mathcal{C}}, \ \ \mathcal{X}_{\mathcal{C}} = \mathcal{K}^{|\mathcal{C}|} \\ \mathcal{C} &= \{n_1, n_2, \ldots, n_l\} \subset \mathcal{N} \end{aligned}$$

the subvector of all previously specified pixels of the observation window, then the statistical properties of the remaining unspecified pixel variables $x_n, n \in (\mathcal{N} \setminus \mathcal{C})$, can be described by the conditional densities

$$p_{n|\mathcal{C}}(x_n|\mathrm{x}_\mathcal{C}) = \frac{P_{n,\mathcal{C}}(x_n, \mathrm{x}_\mathcal{C})}{P_\mathcal{C}(\mathrm{x}_\mathcal{C})} = \sum_{m \in \mathcal{M}} W_m(\mathrm{x}_\mathcal{C}) f_n(x_n|\mu_{mn}, \sigma_{mn}) \ , \quad (9)$$

$$W_m(\mathrm{x}_\mathcal{C}) = \frac{w_m F(\mathrm{x}_\mathcal{C}|\mu_m, \sigma_m)}{\sum_{j \in \mathcal{M}} w_j F(\mathrm{x}_\mathcal{C}|\mu_j, \sigma_j)} \ , \quad (10)$$

$$F(\mathrm{x}_\mathcal{C}|\mu_m, \sigma_m) = \prod_{n \in \mathcal{C}} f_n(x_n|\mu_{mn}, \sigma_{mn}) \ . \quad (11)$$

Let us recall that the simple plug-in formula (9) is formally enabled by a simple evaluation of the marginal densities $P_{n,\mathcal{C}}(x_n, \mathrm{x}_\mathcal{C})$ and $P_\mathcal{C}(\mathrm{x}_\mathcal{C})$. [1] For a fixed position of the observation window the formula (9) can be applied to all the unspecified pixels $n \in \mathcal{N} \setminus \mathcal{C}$. The unknown grey-levels x_n can be synthesized by random sampling from the conditional density $p_{n|C}(x_n|\mathrm{x}_C)$ or by computing the conditional expectation

$$E\{x_n\} = \int x_n p_{n|\mathcal{C}}(x_n|\mathrm{x}_\mathcal{C}) dx_n = \sum_{m \in \mathcal{M}} W_m(\mathrm{x}_\mathcal{C}) \mu_{mn} \ .$$

In our experiments we have used random sampling from the weights $W_m(\mathrm{x}_\mathcal{C})$ and then substituted the chosen parameters μ_m. In this way the synthesized part of the observation window is actually covered by the corresponding component means μ_{mn} of the randomly chosen component. In the next step, after synthesis of all interior pixels, the observation window is shifted to a new position and the conditional distribution (9) has to be computed for a new subset of the specified pixels x_C.

We have used a step-wise left-to-right and top-to-down shifting of the observation window. We have obtained the best results for the shift of about half window-size. Surprisingly the synthesized texture image has been more stable and more realistic than in the case of pixel-wise shifting. This effect is probably due to a reduced dimension of the underlying marginal densities $P_{n,\mathcal{C}}(x_n, \mathrm{x}_\mathcal{C})$ and $P_\mathcal{C}(\mathrm{x}_\mathcal{C})$. It appears that the limiting quality of the estimated distribution mixture $P(\mathrm{x})$ is less relevant in lower-dimensional marginals.

4. Statistically Optimized Sampling

The texture synthesis procedure as described in Sec. 3 has been applied to different texture images with satisfactory results (cf. Fig. 1 - column 2). Nevertheless, the synthesized textures are monospectral and moreover they appear to be too “smooth” or “filtered” with the high frequency details missing. For this reason we have replaced in the synthesis phase the mean vectors μ_m of the mixture components by the most similar colour “centroides” from the source image, i.e. by real pieces of the original colour texture.

For each vector μ_m we have found the corresponding monospectral centroid c_m by minimizing the Euclidean distance $\|\mathrm{x} - \mu_m\|$ over all possible positions of the observation window x in the monospectral version of the source image:

$$\mathrm{c}_m = (c_1, c_2, \ldots, c_N) = \arg\min_{\mathrm{x}} \{\|\mathrm{x} - \mu_m\|\} \ . \quad (12)$$

Given the position of the optimal centroid c_m, we have chosen the corresponding piece of the colour texture for the synthesis phase.

Let us recall that in this way the texture synthesized by means of the estimated normal mixture is used only as a “draft layout” for the final texture image which is entirely composed of small pieces of the original colour texture. (The last step includes also some smoothing of the boundaries between the neighbouring centroides.) The resulting synthesized texture is actually a mosaic of the original colour texture pieces controlled by the separately synthesized draft layout. The last version of the modelling method has been applied to different textures. In the experiments we have obtained very realistic texture images (cf. Fig. 1 - column 3).

It can be seen that in the final version the texture synthesis is much similar to standard sampling methods. By using the estimated component means μ_m we choose from the original texture image a fixed set of colour “centroides” playing the role of tiles. The synthesis based on a local monospectral prediction controls the tilling in a statistically optimal way by using statistically optimized set of colour tiles.

5. Conclusions

We proposed the Gaussian distribution mixtures based colour texture model. This novel model can be either used to directly synthesize colour textures or to control sophisticated sampling from original measurement data. This last option presented in the paper combines advantages of both basic texture modelling approaches. It allows high visual quality of synthetic textures while requires to store only small patches of the original measurements or even only Gaussian mixtures parameters in the direct modelling version.

An important aspect of the proposed approach is its possible extension to multispectral or mutually registered BTF texture images. In the estimation phase we use a grey-scale

1 The discussion of asymptotic properties of the conditional probability density $p_{n|\mathcal{C}}(x_n|\mathrm{x}_\mathcal{C})$ is beyond the scope of the present paper.

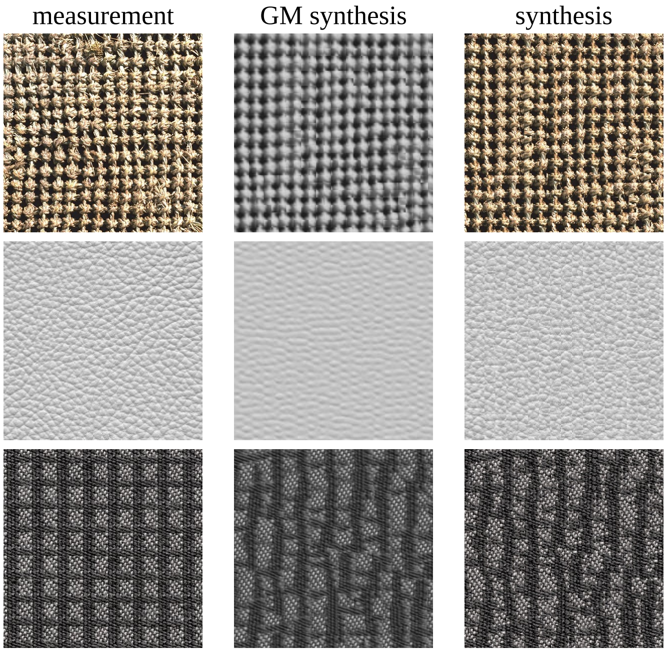

Figure 1. Rattan, leather and carpet examples and their synthetic results.

projection of a multispectral texture image and, in the final phase we use the optimal centroides in a multispectral or BTF version.

Acknowledgements

This research was supported by the EC projects no. IST-2001-34744 RealReflect, FP6-507752 MUSCLE, by the grant No. 402/03/1310 of the Czech Grant Agency and partially by the grant MŠMT No. ME567 MIXMODE.

References

[1] J. Bennett and A. Khotanzad. Multispectral random field models for synthesis and analysis of color images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(3):327–332, March 1998.

[2] J. Bennett and A. Khotanzad. Maximum likelihood estimation methods for multispectral random field image models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21(3):537–543, 1999.

[3] J. De Bonet. Multiresolution sampling procedure for analysis and synthesis of textured images. In *Proc. SIGGRAPH 97*, pages 361–368. ACM, 1997.

[4] J. Dong and M. Chantler. Capture and synthesis of 3d surface texture. In *Texture 2002*, volume 1, pages 41–45. Heriot-Watt University, 2002.

[5] A. A. Efros and W. T. Freeman. Image quilting for texture synthesis and transfer. In E. Fiume, editor, *SIGGRAPH 2001, Computer Graphics Proceedings*, pages 341–346. ACM Press / ACM SIGGRAPH, 2001.

[6] A. A. Efros and T. K. Leung. Texture synthesis by non-parametric sampling. In *Proc. Int. Conf. on Computer Vision (2)*, pages 1033–1038, 1999.

[7] G. Gimelfarb. *Image Textures and Gibbs Random Fields*. Kluwer Academic Publishers, 1999.

[8] J. Grim and M. Haindl. A discrete mixtures colour texture model. In M. Chantler, editor, *Texture 2002. The 2nd International Workshop on Texture Analysis and Synthesis*, pages 59–62, Glasgow, June 2002. Heriot-Watt University.

[9] J. Grim and M. Haindl. Texture modelling by discrete distribution mixtures. *Computational Statistics Data Analysis*, 41(3-4):603–615, January 2003.

[10] M. Haindl. Texture synthesis. *CWI Quarterly*, 4(4):305–331, December 1991.

[11] M. Haindl. Texture modelling. In B. Sanchez, J. M. Pineda, J. Wolfmann, Z. Bellahse, and F. Ferri, editors, *Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics*, volume VII, pages 634–639, Orlando, USA, July 2000. International Institute of Informatics and Systemics.

[12] M. Haindl and V. Havlíček. Multiresolution colour texture synthesis. In K. Dobrovodský, editor, *Proceedings of the 7th International Workshop on Robotics in Alpe-Adria-Danube Region*, pages 297–302, Bratislava, June 1998. ASCO Art.

[13] M. Haindl and V. Havlíček. A multiscale colour texture model. In R. Kasturi, D. Laurendeau, and C. Suen, editors, *Proceedings of the 16th International Conference on Pattern Recognition*, pages 255–258, Los Alamitos, August 2002. IEEE Computer Society.

[14] M. Haindl and V. Havlíček. A multiresolution causal colour texture model. In F. J. Ferri, J. M. Inesta, A. Amin, and P. Pudil, editors, *Advances in Pattern Recognition, Lecture Notes in Computer Science 1876*, chapter 1, pages 114 –122. Springer-Verlag, Berlin, August 2000.

[15] D. Heeger and J. Bergen. Pyramid based texture analysis/synthesis. In *Proc. SIGGRAPH 95*, pages 229–238. ACM, 1995.

[16] R. Kashyap. Analysis and synthesis of image patterns by spatial interaction models. In L. Kanal and A.Rosenfeld, editors, *Progress in Pattern Recognition 1*, North-Holland, 1981. Elsevier.

[17] R. Paget and I. D. Longstaff. Texture synthesis via a noncausal nonparametric multiscale markov random field. *IEEE Trans. on Image Processing*, 7(8):925–932, 1998.

[18] Y. Xu, B. Guo, and H. Shum. Chaos mosaic: Fast and memory efficient texture synthesis. Technical Report MSR-TR-2000-32, Redmont, 2000.

[19] S. Zelinka and M. Garland. Towards real-time texture synthesis with the jump map. In *13th European Workshop on Rendering*, 2002.

[20] S. Zhu, X. Liu, and Y. Wu. Exploring texture ensembles by efficient markov chain monte carlo - toward a “trichromacy” theory of texture. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(6):554–569, June 2000.