# Texture Editing Using Frequency Swap Strategy

Michal Haindl and Vojtěch Havlíček

Institute of Information Theory and Automation
of the ASCR, Prague, Czech Republic
{haindl,havlicek}@utia.cz

**Abstract.** A fully automatic colour texture editing method is proposed, which allows to synthesise and enlarge an artificial texture sharing anticipated properties from its parent textures. The edited colour texture maintains its original colour spectrum while its frequency is modified according to one or more target template textures. Edited texture is synthesised using a fast recursive model-based algorithm. The algorithm starts with edited and target colour texture samples decomposed into a multi-resolution grid using the Gaussian-Laplacian pyramid. Each band pass colour factors are independently modelled by their dedicated 3D causal autoregressive random field models (CAR). We estimate an optimal contextual neighbourhood and parameters for each of the CAR submodel. The synthesised multi-resolution Laplacian pyramid of the edited colour texture is replaced by the synthesised template texture Laplacian pyramid. Finally the modified texture pyramid is collapsed into the required fine resolution colour texture. The primary benefit of these multigrid texture editing models is their ability to produce realistic novel textures with required visual properties capable of enhancing realism in various texture application areas.

## 1 Introduction

Image editing remains a complex user-directed task, often requiring proficiency in design, colour spaces, computer interaction and file management. Editing provides the scene designer with tools which enable to control virtual scene objects, geometric surfaces, illumination and objects faces appearance in the form of their corresponding textures. Image editing software is often characterised [1] by a seemingly endless array of toolbars, filters, transformations and layers. Although some recent attempts [2,3,4,5,6,7,8] have been made to automate this process, automatic integration of user preferences still remains an open problem in the context of texture editing [9,10].

The primary contribution of our method is a simple intuitive and fully automatic tool for the scene designer to modify objects surface appearance by controlled texture modifications. Contrary to some other texture editing approaches such as the procedural textures, the edited texture visual appearance predictably corresponds to the anticipated projection.

Authentic and photo realistic appearance of natural materials covering surfaces of virtual objects in virtual or augmented reality rendered scenes requires nature-like colour textures covering visualised scene objects. Such textures can

X. Jiang and N. Petkov (Eds.): CAIP 2009, LNCS 5702, pp. 1146–1153, 2009.

be either digitised natural textures or textures synthesised from an appropriate mathematical model. The former simplistic option suffers among others from extreme memory requirements for storage of a large number of digitised cross-sectioned slices through different material samples. Synthetic textures are more flexible, extremely compressed (few parameters have to be stored only), they may be evaluated directly in procedural form and can be designed to meet certain constraints or to secure some desirable properties (e.g., smooth periodicity, no visible discontinuities, etc.). The underlying mathematical models have besides presented texture editing also many other applications (e.g., image restoration, image and video compression, classification, segmentation, etc.).

Several monospectral texture modelling approaches were published, e.g., [11,12], among them also few colour models, e.g., [13,14,15,16] and some survey articles are available [17,18] as well. [13] introduced a fast multiresolution Markov random field based method. Although this method avoids the time consuming Markov chain Monte Carlo simulation so typical for applications of Markov models it still requires several simplifying approximations. Several alternative Markovian colour texture models such as the simultaneous 2D causal autoregressive random fields (2D CAR) [16], 2D Gaussian Markov models (2D GMRF) [19], or 3D CAR [20] were introduced as well and later generalised also for Bidirectional Texture Function (BTF) [21,22,23,24] or dynamic textures [25]. These models are appropriate for colour texture synthesis not only because they do not suffer from some problems of alternative options (see [17,18] for details) but they are also easy to analyze as well as to synthesise and last but not least they are still flexible enough to imitate a large set of natural and artificial textures.

## 2 Markovian Texture Model

We assume to have two colour textures $Y_\alpha, Y_\delta$ which can be represented using a Markovian random field model (MRF). The texture $Y_\alpha$ is the input texture which will be modified according to a target template texture $Y_\delta$. The edited colour texture maintains most of its original colour spectrum but changes its frequency to resemble the template texture $Y_\delta$. Single frequency factors are modelled using the exceptionally fast 3D wide-sense Markov causal autoregressive random field model (3D CAR). Let the digitised colour texture $Y$ is indexed on a finite rectangular three-dimensional $N \times M \times d$ underlying lattice $I$, where $N \times M$ is the image size and $d$ is the number of spectral bands (i.e., $d = 3$ for usual colour textures). Let us denote a simplified multiindices $r, s$ to have two components $r = [r_1, r_2], s = [s_1, s_2]$. The first component is row and the second one is column index, respectively.

### 2.1 Frequency Factorisation

The analyzed colour texture image is decomposed into a multi-resolution grid using Laplacian pyramid and the intermediary Gaussian pyramid. The benefit of the multigrid approach is the replacement of a large neighbourhood CAR model with a set of several simpler CAR models which are easy to estimate and synthesise. Each resolution data are independently modelled by their dedicated CAR. Each one generates a single spatial frequency band of the texture. The Gaussian pyramid $\ddot{Y}_{\nu}^{(k)}$ is a sequence of images in which each one is a low-pass down-sampled version of its predecessor where the weighting function (FIR generating kernel) is chosen subject to the following constraints:

$$w_s = \hat{w}_{s_1} \hat{w}_{s_2}, \qquad \sum_i \hat{w}_i = 1 \quad , \qquad \hat{w}_i = \hat{w}_{-i} \quad , \qquad \hat{w}_0 = 2\hat{w}_1 \;\; (\zeta = 1)$$

and $\nu \in \{\alpha, \delta\}$. The solution of the above constraints for the reduction factor 3 $(2\zeta + 1)$ is $\hat{w}_0 = 0.5, \hat{w}_1 = 0.25$ and the FIR equation is now

$$\ddot{Y}_{r,\nu}^{(k)} = \sum_{i,j=-\zeta}^{\zeta} \hat{w}_i \hat{w}_j \ddot{Y}_{2r+(i,j),\nu}^{(k-1)} \quad . \tag{1}$$

The Gaussian pyramid for a reduction factor $n$ is

$$\ddot{Y}_{r,\nu}^{(k)} = \downarrow_r^n (\ddot{Y}_{\nu}^{(k-1)} \otimes w) \qquad k = 1, 2, \ldots \quad , \tag{2}$$

where $\ddot{Y}_{\nu}^{(0)} = Y_{\nu}$ , $\downarrow^n$ denotes down-sampling with reduction factor $n$ and $\otimes$ is the convolution operation.

The Laplacian pyramid $\dot{Y}_{r,\nu}^{(k)}$ contains band-pass components and provides a good approximation to the Laplacian of the Gaussian kernel. It can be constructed by differencing single Gaussian pyramid layers:

$$\dot{Y}_{r,\nu}^{(k)} = \ddot{Y}_{r,\nu}^{(k)} - \uparrow_r^n (\ddot{Y}_{\nu}^{(k+1)}) \qquad k = 0, 1, \ldots \quad , \tag{3}$$

where $\uparrow^n$ is the up-sampling with an expanding factor $n$. Single orthogonal multispectral components are thus decomposed into a multi-resolution grid and each resolution data are independently modelled by their dedicated independent Gaussian noise driven autoregressive random field model as follows.

### 2.2 3D CAR Texture Model

Single frequency factors are modelled using the causal autoregressive random field (3D CAR) model [20] which is a family of random variables with a joint probability density on the set of all possible realisations $Y$ of the $M \times N \times d$ lattice $I$, subject to the following condition:

$$p(Y \mid \gamma, \Sigma^{-1}) = \frac{|\Sigma^{-1}|^{\frac{(MN-1)}{2}}}{(2\pi)^{\frac{d(MN-1)}{2}}} \exp\left\{ -\frac{1}{2} tr\{\Sigma^{-1} \begin{pmatrix} -I \\ \gamma^T \end{pmatrix}^T \tilde{V}_{MN-1} \begin{pmatrix} -I \\ \gamma^T \end{pmatrix}\} \right\} \; ,$$

where the following notation is used

$$\tilde{V}_{r-1} = \begin{pmatrix} \tilde{V}_{yy(r-1)} & \tilde{V}_{xy(r-1)}^T \\ \tilde{V}_{xy(r-1)} & \tilde{V}_{xx(r-1)} \end{pmatrix} \; , \qquad \tilde{V}_{yy(r-1)} = \sum_{k=1}^{r-1} Y_k Y_k^T \; ,$$

$$\tilde{V}_{xy(r-1)} = \sum_{k=1}^{r-1} X_k Y_k^T \; , \qquad \tilde{V}_{xx(r-1)} = \sum_{k=1}^{r-1} X_k X_k^T \; .$$

The 3D CAR model can be expressed as a stationary causal uncorrelated noise driven 3D autoregressive process:

$$Y_r = \gamma X_r + e_r \ , \tag{4}$$

where $\gamma$ is the $d \times d\eta$ parameter matrix $\gamma = [A_1, \ldots, A_\eta]$ , $\eta = card(I_r^c)$ , $I_r^c$ is a causal neighbourhood, $e_r$ is a Gaussian white noise vector with zero mean and a constant but unknown covariance matrix $\Sigma$ (estimated by (7)) and $X_r$ is a corresponding vector of $Y_{r-s}$ (design vector).

**Parameter Estimation.** The selection of an appropriate CAR model support is important to obtain good results in modelling of a given random field. If the contextual neighbourhood is too small it cannot capture all details of the random field. Inclusion of the unnecessary neighbours on the other hand add to the computational burden and can potentially degrade the performance of the model as an additional source of noise. The optimal Bayesian decision rule for minimising the average probability of decision error chooses the maximum posterior probability model, i.e., a model $M_i$ corresponding to $\max_j\{p(M_j|Y^{(r-1)})\}$ where $Y^{(r-1)}$ denotes the known process history $Y^{(r-1)} = \{Y_{r-1}, Y_{r-2}, \ldots, Y_1\}$ . The most probable CAR model given past data $Y^{(r-1)}$, the normal-Wishart parameter prior and the uniform model prior is the model $M_i$ for which $i = \arg\max_j\{D_{j(r-1)}\}$

$$\begin{aligned} D_{j(r-1)} = {} & \frac{d^2\eta}{2} \ln \pi \sum_{i=1}^{d} \left[ \ln \Gamma\left(\frac{\beta(r) - d\eta + d + 2 - i}{2}\right) - \ln \Gamma\left(\frac{\beta(0) - d\eta + d + 2 - i}{2}\right) \right] \\ & - \frac{d}{2} \ln |V_{xx(r-1)}| - \frac{\beta(r) - d\eta + d + 1}{2} \ln |\lambda_{(r-1)}| \end{aligned}$$

where $\beta(r) = \beta(0) + r - 1$ , $\beta(0) > 1$ , and

$$\lambda_{(r)} = V_{yy(r)} - V_{xy(r)}^T V_{xx(r)}^{-1} V_{xy(r)} \ . \tag{5}$$

Parameter estimation of a CAR model using the maximum likelihood, the least square or Bayesian methods can be found analytically. The Bayesian parameter estimations of the causal AR model with the normal-Wishart parameter prior which maximise the posterior density are:

$$\hat{\gamma}_{r-1}^T = V_{xx(r-1)}^{-1} V_{xy(r-1)} \tag{6}$$

and

$$\hat{\Sigma}_{r-1} = \frac{\lambda_{(r-1)}}{\beta(r)} \ , \tag{7}$$

where $V_{uz(r-1)} = \tilde{V}_{uz(r-1)} + V_{uz(0)}$ and matrices $V_{uz(0)}$ are the corresponding matrices from the normal-Wishart parameter prior. The estimates (5), (6),(7) can be also evaluated recursively if necessary.

**Model Synthesis.** The CAR model synthesis is very simple and a 3D causal CAR random field can be directly generated from the model equation (4) using a multivariate Gaussian generator. Single CAR models synthesise spatial frequency bands of the texture.

### 2.3 Laplacian Pyramid Swap

The synthesised Laplacian pyramid layers from the target texture target template texture $\dot{Y}_\delta$ are used instead of the corresponding input texture Laplacian pyramid layers ($\dot{Y}_\alpha$), i.e.

$$\dot{Y}^{(k)}_{r,\alpha} = \dot{Y}^{(k)}_{r,\delta} \qquad \forall k \ . \tag{8}$$

The input texture $Y_\alpha$ Laplacian pyramid layers ($\dot{Y}_\alpha$) are not needed and their corresponding 3D CAR models are neither estimated nor synthesised. On the contrary, the input Gaussian pyramid $\ddot{Y}^{(k)}_{r,\alpha}$ at the most coarse level contains original texture colour spectrum and is needed (and thus estimated) for the edited texture synthesis. If the Laplacian pyramids of both textures have similar numerical values, then the edited texture colour spectrum is unchanged, otherwise its colour spectrum is a compromise between both textures colour spectra. The edited fine-resolution synthetic colour texture is obtained from the pyramid collapse procedure (inversion process to (2),(3) modified to (8)).

## 3 Experimental Results

Figs.1,2 show six examples of different natural or man made colour textures edited using the presented algorithm. All original natural colour textures (upper

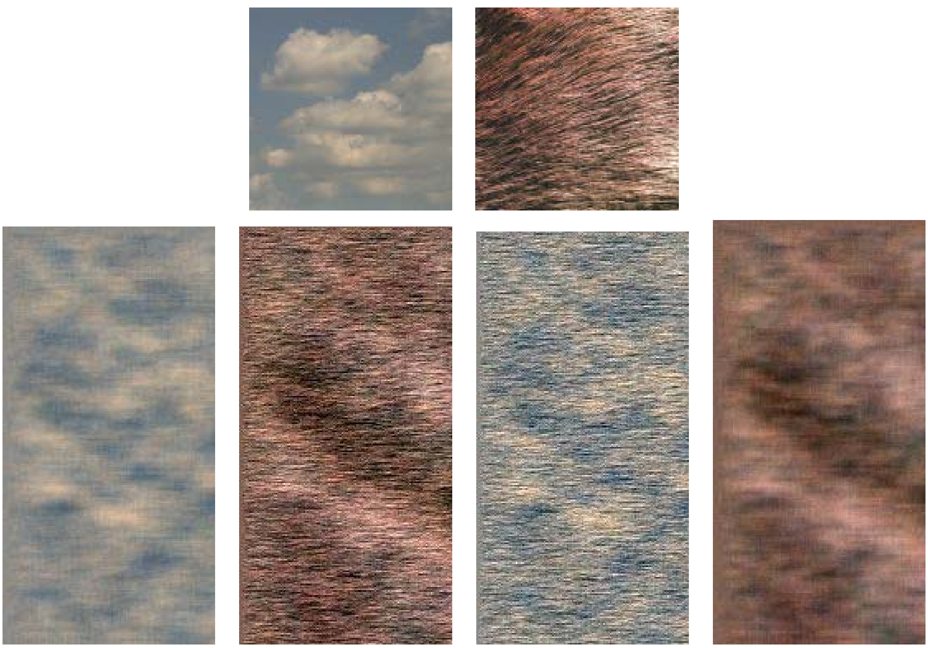

**Fig. 1.** Natural cloud and fur textures (upper row), their resynthesis using a set of 3D CAR models (bottom left) and their edited counterparts (bottom right)

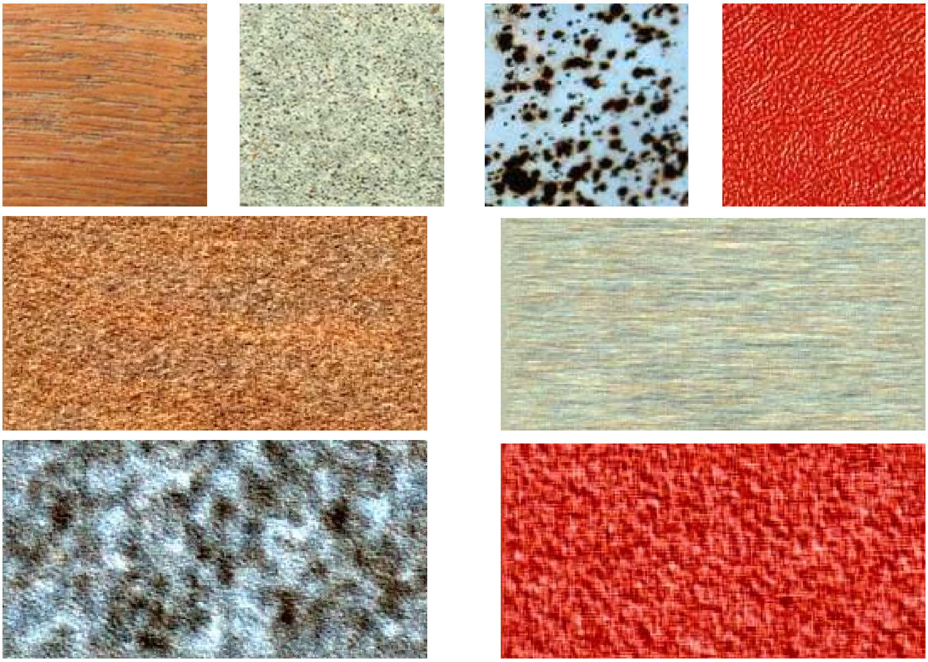

**Fig. 2.** Wood, tile, lichen, and leather natural textures and their resynthesised edited counterparts using the 3D CAR models (middle and bottom)

rows) are taken either from the VisTex [26] database or from our own extensive colour texture database. The images on Fig.1-bottom left show synthesised enlarged examples of the input textures while the Figs.1-bottom right,2-middle,bottom rows present results from the presented texture editing method with frequency modification using the alternate column texture as the template texture $Y_\delta$ with the reduction factor $n = 2$ and the number of pyramid layers $k \in \{2, 3\}$. The edited textures are generated fully automatically and they clearly demonstrate original texture frequency modified to resemble the template texture frequency. The method can be easily combined with some texture segmenter if we need to edit separately single textures appearing in the scene. The method allows very high compression ratio, because only tens parameters for every fractional 3D CAR model have to be stored regardless of the required texture enlargement. This extreme compression ration ($1 : 10^6$ for BTF modelling [21]) is the prerequisite for BTF editing applications where alternative texture editing methods cannot be used due to unsolvable memory requirements.

## 4 Conclusions

A simple fully automatic colour texture editing method is proposed. The method allows to synthesise and enlarge artificial textures which resemble both their parents textures. The edited texture inherits primarily spectral information from one parent and frequency information from the other one. This procedure can be repeated for more complex lineage trees which allows to inherit visual properties from more than two parent textures. The method allows very high compression ratio for transmission or storing texture information, while sometimes compromises visual quality of the resulting texture, similarly as any other adaptive texture model. The edited texture analysis as well as synthesis is extremely fast (due to complete analytical solution) and can be used in real-time applications. The method can be easily generalised also for other types of textures such as the Bidirectional Texture Function (BTF) or dynamic textures.

## Acknowledgements

This research was supported by the grant GAČR 102/08/0593 and partially by the MŠMT grants 1M0572 DAR, 2C06019.

## References

1. Brooks, S., Dodgson, N.A.: Integrating procedural textures with replicated image editing. In: Proceedings of the 3rd International Conference on Computer Graphics and Interactive Techniques in Australasia and Southeast Asia 2005, Dunedin, New Zealand, November 29 - December 2, pp. 277–280. ACM, New York (2005)
2. Ashikhmin, M.: Synthesizing natural textures. In: ACM Symposium on Interactive 3D Graphics, pp. 217–226 (2001)
3. Bar-Joseph, Z., El-Yaniv, R., Lischinski, D., Werman, M.: Texture mixing and texture movie synthesis using statistical learning. IEEE Transactions on Visualization and Computer Graphics 7, 120–135 (2001)
4. 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, 127–150 (2001)
5. Hertzmann, A., Jacobs, C.E., Oliver, N., Curless, B., Salesin, D.H.: Image analogies. ACM Trans. Graph., 327–340 (2001)
6. Wiens, A.L., Ross, J.: Gentropy: evolving 2d textures. Computers & Graphics 26, 75–88 (2002)
7. Wang, X., Wang, L., Liu, L., Hu, S., Guo, B.: Interactive modeling of tree bark. In: Proc. 11th Pacific Conf. on Comp. Graphics and Appl., pp. 83–90. IEEE, Los Alamitos (2003)
8. Brooks, S., Cardle, M., Dodgson, N.A.: Enhanced texture editing using self similarity. In: VVG, pp. 231–238 (2003)
9. Brooks, S., Dodgson, N.A.: Self-similarity based texture editing. ACM Trans. Graph 21, 653–656 (2002)
10. Khan, E.A., Reinhard, E., Fleming, R.W., Bülthoff, H.H.: Image-based material editing. ACM Trans. Graph 25, 654–663 (2006)
11. Besag, J.: Spatial interaction and the statistical analysis of lattice systems. Journal of the Royal Statistical Society, Series B B-36, 192–236 (1974)
12. Kashyap, R.: Analysis and synthesis of image patterns by spatial interaction models. In: Kanal, L., Rosenfeld, A. (eds.) Progress in Pattern Recognition, vol. 1. North-Holland, Elsevier (1981)

13. 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)
14. Bennett, J., Khotanzad, A.: Multispectral random field models for synthesis and analysis of color images. IEEE Trans. on Pattern Analysis and Machine Intelligence 20, 327–332 (1998)
15. Bennett, J., Khotanzad, A.: Maximum likelihood estimation methods for multispectral random field image models. IEEE Trans. on Pattern Analysis and Machine Intelligence 21, 537–543 (1999)
16. Haindl, M., Havlíček, V.: A multiresolution causal colour texture model. In: Amin, A., Pudil, P., Ferri, F., Iñesta, J.M. (eds.) SPR 2000 and SSPR 2000. LNCS, vol. 1876, pp. 114–122. Springer, Heidelberg (2000)
17. Haindl, M.: Texture synthesis. CWI Quarterly 4, 305–331 (1991)
18. Haindl, M.: Texture modelling. In: Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics, Orlando, USA, vol. VII, pp. 634–639. International Institute of Informatics and Systemics (2000)
19. Haindl, M., Havlíček, V.: A simple multispectral multiresolution markov texture model. In: Texture 2002, The 2nd international workshop on texture analysis and synthesis, Copenhagen, pp. 63–66. Heriot-Watt University (2003)
20. Haindl, M., Havlíček, V.: A multiscale colour texture model. In: Proceedings of the 16th International Conference on Pattern Recognition, pp. 255–258. IEEE Computer Society, Los Alamitos (2002)
21. Haindl, M., Filip, J.: Fast BTF texture modelling. In: Texture 2003. Proceedings, Edinburgh, pp. 47–52. IEEE Press, Los Alamitos (2003)
22. Haindl, M., Filip, J., Arnold, M.: BTF image space utmost compression and modelling method. In: Proceedings of the 17th IAPR International Conference on Pattern Recognition, vol. III, pp. 194–197. IEEE, Los Alamitos (2004)
23. Haindl, M., Filip, J.: A fast probabilistic bidirectional texture function model. LNCS, pp. 298–305. Springer, Heidelberg (2004)
24. Haindl, M., Filip, J.: Extreme compression and modeling of bidirectional texture function. IEEE Transactions on Pattern Analysis and Machine Intelligence 29, 1859–1865 (2007)
25. Filip, J., Haindl, M., Chetverikov, D.: Fast synthesis of dynamic colour textures. In: Proceedings of the 18th International Conference on Pattern Recognition, ICPR 2006, vol. IV, pp. 25–28. IEEE Computer Society, Los Alamitos (2006)
26. Vision texture (vistex) database. Technical report, Vision and Modeling Group, `http://www-white.media.mit.edu/vismod/`