A Roller - Fast Sampling-Based Texture Synthesis Algorithm

Michal Haindl
Institute of Information Theory and Automation
Academy of Sciences of the Czech Republic
CZ18208, Prague, Czech Republic
haindl@utia.cas.cz

Martin Hatka
Faculty of Nuclear Sciences
and Physical Engineering
Czech Technical University
CZ11519 Prague, Czech Republic
hadis@email.cz

ABSTRACT

This paper describes a method for synthesizing natural textures that realistically matches given colour texture appearance. The novel texture synthesis method, which we call the roller, is based on the overlapping tiling and subsequent minimum error boundary cut. An optimal double toroidal patch is seamlessly repeated during the synthesis step. While the method allows only moderate texture compression it is extremely fast due to separation of the analytical step of the algorithm from the texture synthesis part, universal, and easily implementable in a graphical hardware for purpose of real-time colour texture rendering.

Keywords
Virtual Reality Models, Texture Synthesis.

1. INTRODUCTION

Virtual or augmented reality systems (VR) require object surfaces covered with realistic nature-like colour textures to enhance realism in virtual scenes. To make virtual worlds realistic detailed scene models must be built. Satisfactory models require not only complex 3D shapes accorded with the captured scene, but also lifelike colour and texture. This will increase significantly the realism of the synthetic scene generated. Textures provide useful cues to a subject navigating in such a VR environment, and they also aid in the accurate detailed reconstruction of the environment.

WSCG SHORT papers proceedings ISBN 80-903100-9-5
WSCG'2005, January 31-February 4, 2005
Plzen, Czech Republic.

The purpose of a synthetic texture is to reproduce a given digitized texture image so that both natural and synthetic texture will be indiscernible. However modelling of a natural texture is a very challenging and difficult task, due to unlimited variety of possible surfaces, illumination and viewing conditions simultaneously with the strong discriminative functionality of the human visual system. The related texture modelling approaches may be divided primarily into intelligent sampling and model-based-analysis and synthesis, but no ideal method for texture synthesis exists. Each of existing approaches has its advantages and also limitations.

Model-based colour texture synthesis [Bes74], [Kas81],[Ben98], [Hai91a], [Hai00], [Zhu00],[Gri03], [Hai04] requires non-standard multi-dimensional (3D for static colour textures or even 7D for bidirectional texture function) models. If a 3D texture space can be factorized then these data can be modelled using a set of less-dimensional 2D random field models, otherwise it is necessary to use some 3D random field model. Among such possible models the Gaussian Markov random fields are advantageous not only because they do not suffer with some problems of alternative options (see [Hai91b], [Hai91a],

[Hai00], [Hai00] for details) but they are also relatively easy to synthesize and still flexible enough to imitate a large set of natural and artificial textures. Unfortunately real data space can be decorrelated only approximately, hence the independent spectral component modelling approach suffers with some loss of image information. Alternative full 3D models allow unrestricted spatial-spectral correlation modelling, but its main drawback is large amount of parameters to be estimated and in the case of Markov models also the necessity to estimate all these parameters simultaneously. Model-based methods are mostly too difficult to be implemented in modern graphical card processors.
Intelligent sampling approaches [DeB97], [Efr99], [Efr01], [Hee95], [Xu00] rely on sophisticated sampling from real texture measurements. Given a randomly selected starting block of texture in the image, they propagate out from it selecting new texture blocks. For each new block in the image, all neighboring blocks that have already been generated are checked and the example image (or images) is searched for similar textures. The k best such matches are found and then randomly choosen the corresponding new texture patch from among them. The methods [Efr01],[Efr99], [Wei01] all vary in how the blocks are represented, how similarity is determined, and how the search is performed. Intelligent sampling approaches are based on some sort of original small texture sampling and the best of them produce very realistic synthetic textures, usually better than the model-based methods. However these methods require to store original texture sample, often produce visible seams, they are mostly computationally demanding, they cannot generate textures unseen by the algorithm, and they cannot even approach the large compression ratio of the model-based methods.

The rest of the paper is organised as follows. The following section describes a simple sampling approach based on the repetition of a double toroidal tile carved from the original texture measurement. Results and conclusions are reported in the last section.

2. DOUBLE TOROIDAL TILE

The double toroidal tile (see Fig.1) is limited by the selected minimal rectangle to be inscribed in from the original texture measurement. The texture tile is assumed to be indexed on the regular two-dimensional toroidal lattice. The optimal lattice searched by the algorithm allows for seamless repetition in both horizontal and vertical directions, respectively.
Let us define the overlap error for a pixel r as follows:

$$\psi_r^h = \left(Y_r - Y_{r+[N-h,0]}\right)^2 \quad \forall r \in I_h \ ,$$

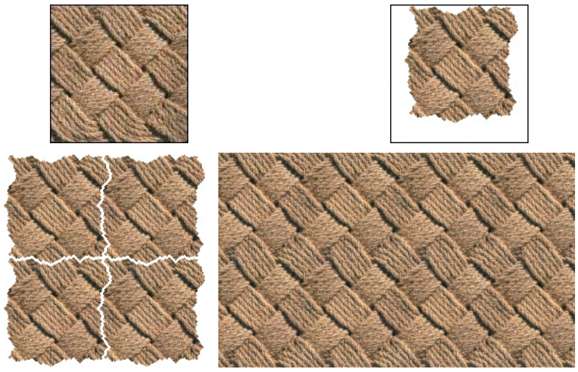


Figure 1. The roller principle - upper row input texture and toroidal tile, bottom row texture generation and the result, respectively.

$$\psi_r^v = \left(Y_r - Y_{r+[0,M-v]}\right)^2 \quad \forall r \in I_v \ ,$$

where Y_r denotes a multispectral pixel indexed on the $N \times M$ underlying lattice. The multiindex r has two components $r = [r_1, r_2]$, the first component is row and the second one column index, respectively. The index sets I_h, I_v are defined

$$\begin{aligned} I_h &= (1,\ldots,h) \times (1,\ldots,M) \ , \\ I_v &= (1,\ldots,N) \times (1,\ldots,v) \ , \end{aligned}$$

h and v lie in preselected intervals $h \in \langle h_{\min}; h_{\max}\rangle$ and $v \in \langle v_{\min}; v_{\max}\rangle$. The optimal horizontal and vertical overlaps are found from the following two relations for $\zeta \in \{h, v\}$:

$$\zeta^* = \min_{\zeta} \left\{ \frac{1}{\zeta} \sum_{\forall r \in I_\zeta} \psi_r^\zeta \right\} \ .$$

Optimal Cut

The optimal cuts for both the horizontal and vertical edge is searched using the dynamic programming method. Alternatively we can use some other suboptimal search such as the A^* algorithm if necessary to speed up also the analytical part of the method. However for most applications the fast synthesis is prerequisite while the computation time for separately and only once solved analytical part is of no importance. Both optimal cuts have to minimize the overall path error

$$\begin{aligned} \Psi_r^{h^*} &= \psi_r^{h^*} + \min\left\{ \Psi_{r-[1,1]}^{h^*}, \Psi_{r-[0,1]}^{h^*}, \Psi_{r+[1,-1]}^{h^*} \right\} \ , \\ \Psi_r^{v^*} &= \psi_r^{v^*} + \min\left\{ \Psi_{r-[1,1]}^{v^*}, \Psi_{r-[1,0]}^{v^*}, \Psi_{r+[-1,1]}^{v^*} \right\} \ . \end{aligned}$$

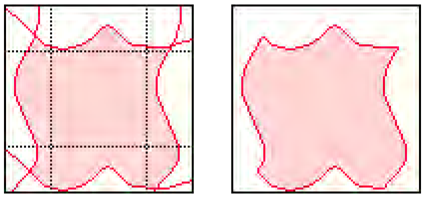


Figure 2. The optimal tile cuts in both directions.

The combination of both optimal vertical and horizontal cuts creates the toroidal tile as is demonstrated on the Fig.2.

Synthesis

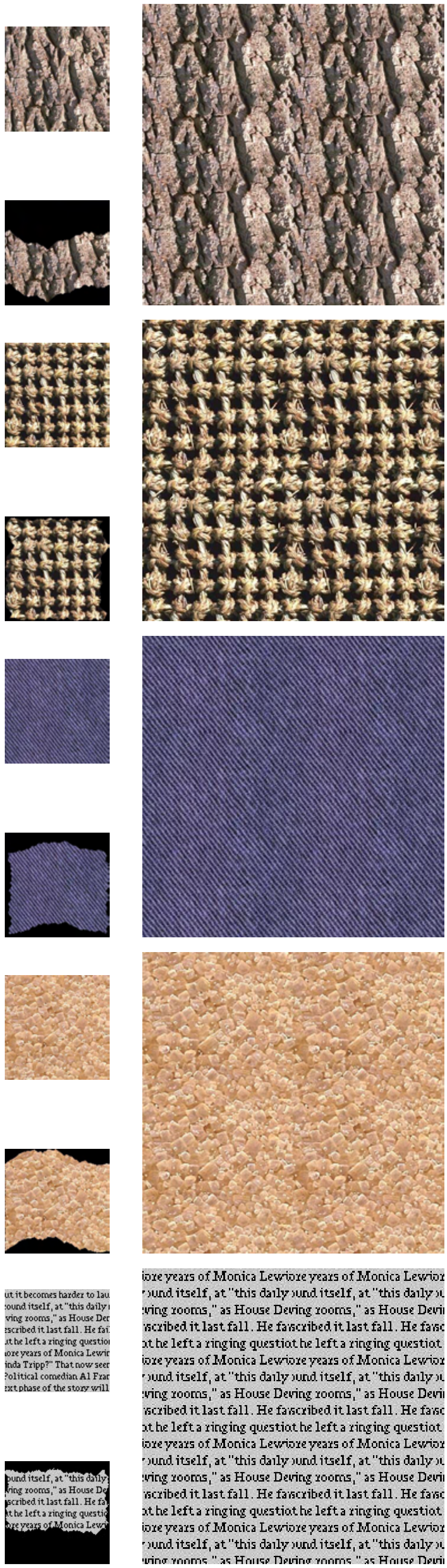
The synthesis of any required colour texture size is simple repetition of the created double toroidal tile in both directions until the required texture is generated. There is no computation involved in this step hence it can be easily implemented in real time or inside the graphical card processing unit.
The complete roller algorithm is as follows:

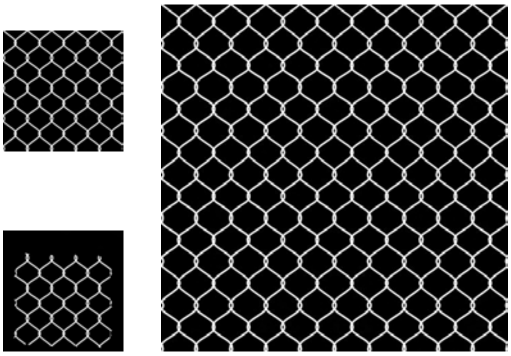
- Analysis
  1. Find the optimal horizontal and vertical overlaps h^*, v^*.
  2. Search for optimal horizontal and vertical cuts starting from $I_h \cap I_v$.
  3. Create the double toroidal texture tile.
- Synthesis

The range of horizontal and vertical overlapping intervals are the only parameters specified by the user. The analytical part is completely separated from the synthesis. The most time consuming part of the analysis is optimal cuts search whose time requirement is proportional to $T \propto v^2hN - 2v^2h^2 + Mvh^2$. The optimal overlap search time is negligible and the synthesis step contains no computations.

3. RESULTS AND CONCLUSIONS

We have tested the algorithm on several hundred colour and grayscale textures from the VisTex database [Pic95], Noctua database, Brodatz textures [Bro66] and mainly from our extensive Prague texture database, which currently contains over 500 colour textures. Tested textures were either natural such as bark, wood, plants, water, etc., or man-made knitwear, upholstery, brick wall, textiles, food products and many others. Some results (bark, rattan, jeans cloth, sugar, text and wire mesh) are demonstrated in the following images.





Resulting textures are mostly surprisingly good for such a very simple algorithm. For example our results on the text texture (the second from the bottom) are indistinguishable (see [Efr01]) from results on the same texture using much more complicated and slower image quilting algorithm [Efr01]. Obviously there is no optimal texture modelling method and also the presented method fails on some textures. However on most of our failure examples also some alternative intelligent sampling methods failed. The test results of our algorithm on our extensive natural texture collection are encouraging. The presented method is extremely fast, very simple and easily implementable even in the graphical processing unit. The method offers moderate compression ratio for transmission or storing texture information while it has negligible computation complexity. The roller method can be used for easy and fast seamless synthesis of any required texture size for many natural or man made textures. The method's extension for bidirectional texture functions is straightforward.

ACKNOWLEDGMENTS

This research was supported by the EC projects no. IST-2001-34744 RealReflect, FP6-507752 MUSCLE and grants No.A2075302, 1ET400750407 of the Grant Agency of the Academy of Sciences CR.

References

[Ben98] BENNETT, J., AND KHOTANZAD, A. 1998. Multispectral random field models for synthesis and analysis of color images. *IEEE Trans. on Pattern Analysis and Machine Intelligence 20*, 3 (March), 327–332.

[Bes74] BESAG, J. 1974. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society, B-36*, 2 (February), 192–236.

[Bro66] BRODATZ, P. 1966. *Textures: A Photographic Album for Artists and Designers*. Dover Publications.

[DeB97] DE BONET, J. 1997. Multiresolution sampling procedure for analysis and synthesis of textured images. In *Proc. SIGGRAPH 97*, ACM, 361–368.

[Efr01] EFROS, A. A., AND FREEMAN, W. T. 2001. Image quilting for texture synthesis and transfer. In *SIGGRAPH 2001, Computer Graphics Proceedings*, ACM Press / ACM SIGGRAPH, E. Fiume, Ed., 341–346.

[Efr99] EFROS, A. A., AND LEUNG, T. K. 1999. Texture synthesis by non-parametric sampling. In *Proc. Int. Conf. on Computer Vision (2)*, 1033–1038.

[Gri03] GRIM, J., AND HAINDL, M. 2003. Texture modelling by discrete distribution mixtures. *Computational Statistics Data Analysis 41*, 3-4 (Jan.), 603–615.

[Hai00] HAINDL, M., AND HAVLÍČEK, V. 2000. A multiresolution causal colour texture model. In *Advances in Pattern Recognition, Lecture Notes in Computer Science 1876*, F. J. Ferri, J. M. Inesta, and P. Pudil, Eds. Springer-Verlag, Berlin, August, ch. 1, 114 –122.

[Hai02b] HAINDL, M., AND HAVLÍČEK, V. 2002. A multiscale colour texture model. In *Proceedings of the 16th IAPR International Conference on Pattern Recognition*, IEEE Press, Quebec City, R. Kasturi, D. Laurendeau, and C. Suen, Eds., vol. I, 255 – 258.

[Hai04] HAINDL, M., GRIM, J., SOMOL, P., PUDIL, P., AND KUDO, M. 2004. A gaussian mixture-based colour texture model. In *Proc. 17th IAPR Int. Conf. on Pattern Recognition*, IEEE Press, Los Alamitos, J. Kittler, M. Petrou, and M. Nixon, Eds., vol. 3, 177–180.

[Hai91a] HAINDL, M. 1991. Texture synthesis. *CWI Quarterly 4*, 4 (December), 305–331.

[Hai91b] HAINDL, M. 1991. Texture synthesis. Tech. Rep. CS-R9139, Centrum voor Wiskunde en Informatica, Amsterdam, The Netherlands.

[Hai00] HAINDL, M. 2000. Texture modelling. In *Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics*, International Institute of Informatics and Systemics, Orlando, USA, B. Sanchez, J. M. Pineda, J. Wolfmann, Z. Bellahse, and F. Ferri, Eds., vol. VII, 634–639.

[Hai03] HAINDL, M. FILIP, J. 2003. Fast btf texture modelling. In *Texture 2003*, IEEE Computer Society.

[Hee95] HEEGER, D., AND BERGEN, J. 1995. Pyramid based texture analysis/synthesis. In *Proc. SIGGRAPH 95*, ACM, 229–238.

[Kas81] KASHYAP, R. 1981. Analysis and synthesis of image patterns by spatial interaction models. In *Progress in Pattern Recognition 1*, Elsevier, North-Holland, L. Kanal and A.Rosenfeld, Eds.

[Pic95] PICKARD, R., GRASZYK, C., MANN, S., WACHMAN, J., PICKARD, L., AND CAMPBELL, L. 1995. Vistex database. Tech.rep., MIT Media Laboratory.

[Wei01] WEI, L., AND LEVOY, M. 2001. Texture synthesis over arbitrary manifold surfaces. In *SIGGRAPH 20001, Computer Graphics Proceedings*, ACM Press / ACM SIGGRAPH / Addison Wesley Longman.

[Xu00] XU, Y., GUO, B., AND SHUM, H. 2000. Chaos mosaic: Fast and memory efficient texture synthesis. Tech. Rep. MSR-TR-2000-32, Redmont.

[Zhu00] ZHU, S., LIU, X., AND WU, Y. 2000. 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 (June), 554–569.