Bidirectional Texture Function Three Dimensional Pseudo Gaussian Markov Random Field Model*

Michal Havlíček †

3rd year of PGS, email: havlimi2@utia.cas.cz
Department of Mathematics
Faculty of Nuclear Sciences and Physical Engineering, CTU in Prague

advisor: Michal Haindl, Pattern Recognition Department, Institute of Information Theory and Automation, ASCR

Abstract. The Bidirectional Texture Function (BTF) is the recent most advanced representation of material surface visual properties. BTF specifies the changes of its visual appearance due to varying illumination and viewing angles. Such a function might be represented by thousands of images of given material surface. Original data cannot be used due to its size and some compression is necessary. This paper presents a novel probabilistic model for BTF textures. The method combines synthesized smooth texture and corresponding range map to produce the required BTF texture. Proposed scheme enables very high BTF texture compression ratio and may be used to reconstruct BTF space as well.

Keywords: BTF, texture analysis, texture synthesis, data compression, virtual reality

Abstrakt. Obousměrná funkce textury je nejpokročilejší v současné době používaná reprezentace vizuálních vlastností povrchu materiálu. Popisuje změny jeho vzhledu v důsledku měnících se úhlů osvětlení a pohledu. Tato funkce může být reprezentována tisíci obrazy daného povrchu materiálu. Původní data nelze díky jejich velikosti použít a je třeba je komprimovat. Tento článek představuje nový pravděpodobnostní model pro BTF textury. Tato metoda kombinuje syntetizovanou hladkou texturu a odpovídající hloubkovou mapu výsledkem čehož je požadovaná BTF textura. Navržený postup umožňuje velmi vysokou úroveň komprese BTF textur a může být také využit při rekonstrukci BTF prostoru.

Klíčová slova: BTF, analýza textur, syntéza textur, komprese dat, virtuální realita

1 Introduction

Bidirectional Texture Function (BTF) [3] is recent most advanced representation of real material surface [6]. It is a seven dimensional function describing surface texture appearance variations due to changing illumination and viewing conditions. The arguments of this function are planar coordinates, spectral plane, azimuthal and elevation angles of both illumination and view respectively.

Such a function for given material is typically represented by thousands of images of surface taken for several combinations of the illumination and viewing angles [16]. Direct

*This research was supported by the grant GAČR 102/08/0593.

†Pattern Recognition Department, Institute of Information Theory and Automation, ASCR.

utilization of acquired data is inconvenient because of extreme memory requirements [16]. Even simple scene with only several materials requires about terabyte of texture memory which is still far out of limits for any current and near future graphics hardware.

Several so called intelligent sampling methods, i.e., based on some sort of original small texture sampling, for example [4], were developed to solve this problem, but they still require to store thousands sample images of the original BTF. In addition, they often produce textures with disruptive visual effects except for the Roller algorithm [12]. Another disadvantage is that they are sometimes very computationally demanding [6].

Contrary to the sampling approaches utilization of mathematical model is more flexible and offers significant compression, because only several parameters have to be stored only. Such a model can be used to generate virtually infinite texture without visible discontinuities. On the other hand, mathematical model can only approximate real measurements, which may result in some kind of visual quality compromise.

One possibility is utilization of random field theory [8]. Generally, texture is assumed to be realization of random field. Additional assumptions further vary depending on particular model. BTF theoretically requires seven dimensional model owing to its definition, but it is possible to approximate general BTF model with a set of much simpler less dimensional ones, three [10],[13] and two dimensional [9],[11] in practice. Mathematical model based on random fields provides easy smooth texture generation with huge compression and visual quality ratio for a large set of textures [6].

Multiscale approach (Gaussian Laplacian pyramid (GLP), wavelet pyramid or subband pyramids) provides successful representation of both high and low frequencies present in texture so that the hierarchy of different resolutions of an input image provides a transition between pixel level features and region or global features [9]. Each resolution component is modelled independently.

We propose an algorithm for BTF texture modelling which combines material range map with synthetic smooth texture generated by multiscale three dimensional Pseudo Gaussian Markov Random Field (3D PGMRF) [1]. Overall texture visual appearance during changes of viewing and illumination conditions is simulated using displacement mapping technique [17].

2 BTF 3D PGMRF Model

The overall BTF 3D PGMRF model scheme can be found on Figure 1. First stage is material range map estimation followed by optional data segmentation (k-means clustering with color cumulative histograms of individual BTF images in perceptually uniform CIELAB colour space as the data features) [9]. An analysed BTF subspace texture is decomposed into multiple resolution factors using GLP [9]. Each resolution data are then independently modelled by their dedicated 3D PGMRF resulting with set of parameters. Multispectral fine resolution subspace component can be then obtained from the pyramid collapse procedure, i.e., the interpolation of sub band components which is the inversion process to the creation of the GLP [9]. Resulting smooth texture is then combined with range map via displacement mapping filter of graphics hardware or software.

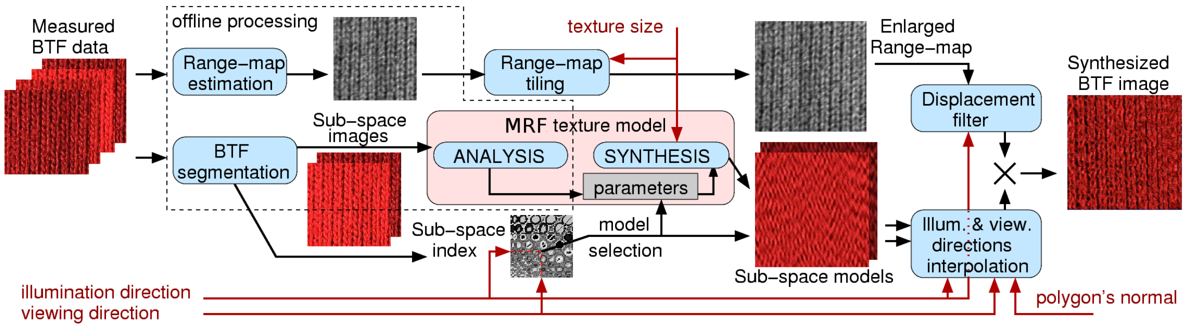

Figure 1: BTF 3D PGMRF model scheme.

2.1 Range Map

The overall roughness of surface significantly influences the BTF texture appearance. This attribute can be specified by range map which comprise information of relative height or depth of individual sites on the surface. Range map can be either measured on real surface or estimated from images of this surface by several existing approaches such as the shape from shading [7], shape from texture [5] or photometric stereo [18]. Since the number of mutually registered BTF measurements for fixed view is sufficient (e.g., 81 in case of the University of Bonn data [16]) it is possible to use over determined photometric stereo to obtain the most accurate outcome. Range map is then stored as a monospectral image where each pixel equals relative height or depth respectively of the corresponding pixel, i.e., point of the surface. If synthesized smooth texture is larger than stored range map then range map is enlarged by the Roller technique [12] chosen for its good properties.

3 3D PGMRF Model

Three dimensional texture random field models are defined as random values representing intensity levels on multiple two dimensional lattices (three in case of widely used colour spaces such as RGB, CIELAB, YUV, YIQ for instance, all of them are widely used in computer graphics, although number of lattices is not limited). The value at each lattice location is considered to be a linear combination of neighbouring ones and some additive noise component. All lattices are considered as double toroidal.

Let a location within an $M \times M$ two dimensional lattice be denoted by (i,j) with $i,j \in J$ where the set J is defined as $J = \{0, 1, \ldots, M-1\}$. The set of all lattice locations is then defined as $\Omega = \{ (i,j) : i,j \in J \}$. Let the value of an image observation at location (i,j) and lattice k be denoted by $y(i,j,k)$ and P equals number of lattices. All random variables forming vector $y(i,j) = (y(i,j,k))\, (i,j) \in \Omega,\ k \in \hat{P}$ are expected to have zero mean. Neighbour sets relating the dependence of points at lattice k on points at lattice l are defined as $N_{kl} = \{ (i,j) : i,j \in \pm J \}$ with the associated neighbour coefficients $\theta(k,l) = \{ \theta(i,j,k,l) : (i,j) \in N_{kl} \}$ where $\pm J = \{-(M-1), \ldots, -1, 0, 1, \ldots, M-1\}$ and $k,l \in \hat{P}$. We also use shortened notation: $\theta = \{ \theta(k,l);\, k,l \in \hat{P} \}$. Our model is defined on symmetric hierarchical contextual neighbour set (Figure 2), i.e., this holds: $r \in N_{kl} \iff -r \in N_{lk}$. Since all sets N_{kl} are equivalent in our implementation, although generally they do not have to be, we use shortened notation N for simplification purposes.

The 3D PGMRF model relates each zero mean pixel value by a linear combination of neighbouring ones and an additive uncorrelated Gaussian noise component [1]:

$$y(i,j,k) = \sum_{n=1}^{P} \sum_{(l,m)\in N} \theta(l,m,k,n)\ y(i+l,j+m,n) + e(i,j,k) \tag{1}$$

where

$$e(i,j,k) = \sum_{n=1}^{P} \sum_{(l,m)\in \Omega} c(l,m,k,n)\ w(i+l,j+m,n)$$

and $w(i,j,k)$ represents zero mean unit variance i.i.d. variable for $(i,j) \in \Omega,\ k \in \hat{P}$. Rewriting the autoregressive equation (1) to the matrix form, with random fields $y = \{\ y(i,j,k);\ (i,j) \in \Omega,\ k \in \hat{P}\ \}$ and $w = \{\ w(i,j,k);\ (i,j) \in \Omega,\ k \in \hat{P}\ \}$ model equations become $By = w$ where

$$B = \begin{pmatrix} B(\theta(1,1)) & B(\theta(1,2)) & \dots & B(\theta(1,P)) \\ B(\theta(2,1)) & B(\theta(2,2)) & \dots & B(\theta(2,P)) \\ \vdots & \vdots & \ddots & \vdots \\ B(\theta(P,1)) & B(\theta(P,2)) & \dots & B(\theta(P,P)) \end{pmatrix}.$$

Matrix B is in fact $PM^2 \times PM^2$ sized matrix composed of $M^2 \times M^2$ block circulant matrices

$$B(\theta(k,l)) = \begin{pmatrix} B(\theta(k,l))_1 & B(\theta(k,l))_2 & \dots & B(\theta(k,l))_M \\ B(\theta(k,l))_M & B(\theta(k,l))_1 & \dots & B(\theta(k,l))_{M-1} \\ \vdots & \vdots & \ddots & \vdots \\ B(\theta(k,l))_2 & B(\theta(k,l))_3 & \dots & B(\theta(k,l))_1 \end{pmatrix} \tag{2}$$

where each element of matrix (2): $B(\theta(k,l)_p)$ is $M \times M$ circulant matrix with elements $b(\theta(k,l))_p(m,n)$ defined as:

$$b(\theta(k,l))_p(m,n) = \begin{cases} 1 & k = l,\ p = l,\ m = n \\ -\theta(i,j,k,l) & i = p-1,\ j = ((n-m)\ mod\ M),\ (i,j) \in N \\ 0 & otherwise \end{cases}$$

Let us remark that the selection of an appropriate model support is important to obtain good results in modelling of a given random field. If used hierarchical contextual neighbourhood set is too small then corresponding model cannot capture all details of the random field. Contrariwise inclusion of the unnecessary neighbours increases both time and memory demands with possible model performance degradation as an additional source of noise.

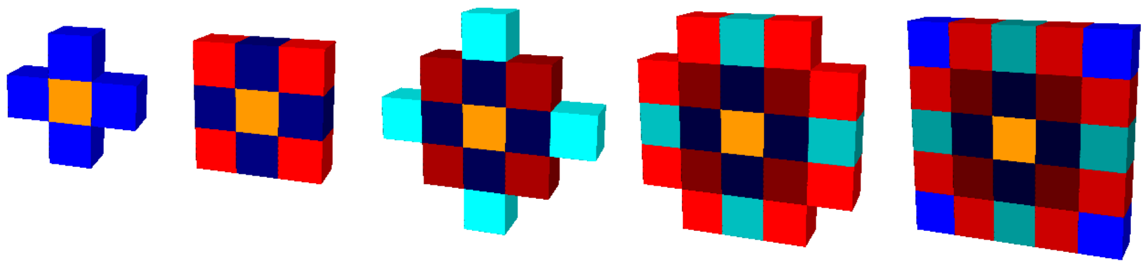

Figure 2: Examples of the used hierarchical contextual neighbourhood sets. The (0,0) position is represented by the central light square while relative neighbour locations are darker surrounding ones. First order neighbourhood to fifth order neighbourhood, from left to right.

3.1 Parameters Estimation

The model is completely specified by parameters $\theta = \{\ \theta(k,l) : k \geq l,\ k \in \hat{P},\ l \in \hat{P}\ \}$ (as $\theta(k,l) = \theta(l,k)$, $\forall k,l$ due to symmetry of neighbourhood) and vector ρ where each component $\rho(k),\ k \in \hat{P}$ of ρ specifies variance of noise component of lattice k. These parameters may be estimated by means of the Least Squares (LS) technique [1]. The LS estimates of the neighbour set coefficients $\theta(i,j,k,l),\ (i,j) \in N,\ k,l \in \hat{P}$ of vector θ are independent of the variance vector ρ. It is due to correlation structure of noise component [1]:

$$\varepsilon\{e(i,j,k)e(i+l,j+m,n)\} = \begin{cases} -\theta(l,m,k,n)\sqrt{\rho(k)\rho(n)} & (l,m) \in N\,, \\ \rho(n) & l=0,\ m=0,\ k=n\,, \\ 0 & otherwise\,. \end{cases}$$

If $\rho(k) = \rho(n)\ \forall k,n \in \hat{P}$ then the random field becomes strictly Gaussian Markov with $\hat{\theta}$ depending on $\hat{\rho}$ making impossible non iterative estimation [1].

Estimates may be derived from equating the observed values to their expected ones, i.e., $y(i,j) = Q^T(i,j)\theta,\ (i,j) \in \Omega$ where

$$Q(i,j) = \begin{pmatrix} q(i,j,1,1) & q(i,j,1,2) & \dots & 0 \\ 0 & q(i,j,2,1) & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & q(i,j,P,P) \end{pmatrix}^T$$

$$q(i,j,k,n) = \begin{cases} (y(i+l,j+m,k) + y(i-l,j-m,k),\ (l,m) \in N) & k=n \\ (y(i+l,j+m,n),\ (l,m) \in N) & k<n \\ (y(i-l,j-m,n),\ (l,m) \in N) & k>n \end{cases}$$

The LS solution $\hat{\theta}$ and $\hat{\rho}$ can be found then as [1]

$$\hat{\theta} = \left(\sum_{(i,j)\in\Omega} Q(i,j) Q^T(i,j) \right)^{-1} \left(\sum_{(i,j)\in\Omega} Q(i,j) y(i,j) \right) ,$$

$$\hat{\rho} = \frac{1}{M^2} \sum_{(i,j)\in\Omega} (y(i,j) - \hat{\theta}^T Q(i,j))^2 \quad .$$

This approximation of real values of parameters allows to avoid expensive numerical optimization method at the cost of accuracy [1].

Additional parameter is mean $\mu = (\mu(k))$, $k \in \hat{P}$. Mean of each spectral plane is estimated as the arithmetic mean and then is subtracted from the plane (prior to estimation of θ and ρ) so that image can be regarded as realization of zero mean random field.

3.2 Image Synthesis

Estimated model parameters $\hat{\theta}$, $\hat{\rho}$ and $\hat{\mu}$ represent original data. So that only their values (several real numbers) need to be stored instead of those data themselves thus this approach offers extreme compression.

A general multidimensional Gaussian Markov random field model has to be synthesized using some of the Markov Chain Monte Carlo (MCMC) method [8]. Due to the double toroidal lattice assumption it is possible to employ efficient non iterative synthesis based on the fast discrete Fourier transformation (DFT) [1].

The model equations (1) may be expressed in terms of the DFT of each lattice as

$$Y(i,j,k) = \sum_{n=1}^{P} \sum_{(l,m)\in N} \theta(l,m,k,n) Y(i,j,n) e^{\sqrt{-1}\omega} + \sqrt{\rho(k)} W(i,j,k) \tag{3}$$

where $Y(i,j,k)$ and $W(i,j,k)$ are the two dimensional DFT coefficients of the image observation $y(i,j,k)$ and noise sequence $w(i,j,k)$, respectively, and $\omega = \frac{2\pi(il+jm)}{M}$ with $(i,j) \in \Omega$ and $k \in \hat{P}$. Model equations (3) can be written in matrix form as $Y(i,j) = \Lambda(i,j)^{-1} \, \Sigma^{\frac{1}{2}} \, W(i,j)$ with the matrices Σ and $\Lambda(i,j)$ defined as [1]:

$$\Sigma = \begin{pmatrix} \rho(1) & 0 & \dots & 0 \\ 0 & \rho(2) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \rho(P) \end{pmatrix} ,$$

$$\Lambda(i,j) = \begin{pmatrix} \lambda(i,j,1,1) & \lambda(i,j,1,2) & \dots & \lambda(i,j,1,P) \\ \lambda(i,j,2,1) & \lambda(i,j,2,2) & \dots & \lambda(i,j,2,P) \\ \vdots & \vdots & \ddots & \vdots \\ \lambda(i,j,P,1) & \lambda(i,j,P,2) & \dots & \lambda(i,j,P,P) \end{pmatrix} ,$$

$$\lambda(i,j,k,n) = \begin{cases} 1 - \sum_{(l,m)\in N} \theta(l,m,k,n) \, e^{\sqrt{-1}\frac{2\pi(il+jm)}{M}} & k = n \\ -\sum_{(l,m)\in N} \theta(l,m,k,n) \, e^{\sqrt{-1}\frac{2\pi(il+jm)}{M}} & k \neq n \end{cases} \quad .$$

The synthesis process begins with generation of two dimensional arrays of white noise w with help of pseudo random number generator for each spectral plane independently. It is followed by two dimensional discrete fast Fourier transform so that arrays W are obtained. Transformation $\Lambda(i,j)^{-1}\ \Sigma^{\frac{1}{2}}\ W(i,j)$ is then computed for each discrete frequency index $(i,j) \in \Omega$. Following step which is inverse two dimensional fast discrete Fourier transform results with image y with zero mean spectral planes so desired mean $\mu(k)$ need to be added to corresponding plane k, $\forall k \in \hat{P}$.

4 Results

We have tested BTF 3D PGMRF model on BTF colour textures from the University of Bonn BTF measurements [16] which represents the most accurate ones available to date [6]. Every material in the database is represented by 6561 images, 800 × 800 RGB pixels each, corresponding to 81 × 81 different view and illumination angles respectively.

The open source project Blender[1] with plugin for BTF texture support [14] was used to render the results. Very simple scene consisting one source of light one three dimensional object represented by polygons and one camera (its coordinates defines view angles) was rendered several times with varying illumination angles while view angles stayed fixed. Synthetic smooth texture combined with range map in displacement mapping filter of Blender was mapped on the object. Several examples may be reviewed on Figures 3 and 4 where visual quality of synthesised BTF may be compared with measured BTF.

The model was also tested on colour textures picked from Amsterdam Library of Textures (ALOT)[2] [2] which consists more coloured, but less dense sampled materials.

5 Conclusion

The main benefit of the presented method is realistic representation of texture colourfulness, which is naturally apparent in case of very distinctively coloured textures. Any simpler two dimensional random field model is not almost able to achieve such results due to colour information loss caused by necessary spectral decorrelation of input data [9]. The multiscale approach is more robust and sometimes allows better results than the single scale one it is when model cannot represent low frequencies properly. This model offers efficient and seamless enlargement of BTF texture to arbitrary size and very high BTF texture compression ratio which cannot be achieved by any other sampling based BTF texture synthesis method while still comparable with other random field BTF models [6]. This can be useful for, e.g., transmission, storing and modelling realistic visual surface texture data with possible application in robust visual classification, human perception study, segmentation, virtual prototyping, image restoration, aging modelling, face recognition and many others [6]. On the other hand the model has still moderate computation complexity. Described approach does not need any kind of time consuming numerical optimisation, e.g., Markov chain Monte Carlo method which is usually employed for such tasks [8]. In addition analysis complexity is not important too much since it is performed

[1] http://www.blender.org

[2] http://staff.science.uva.nl/~ aloi/public_alot/

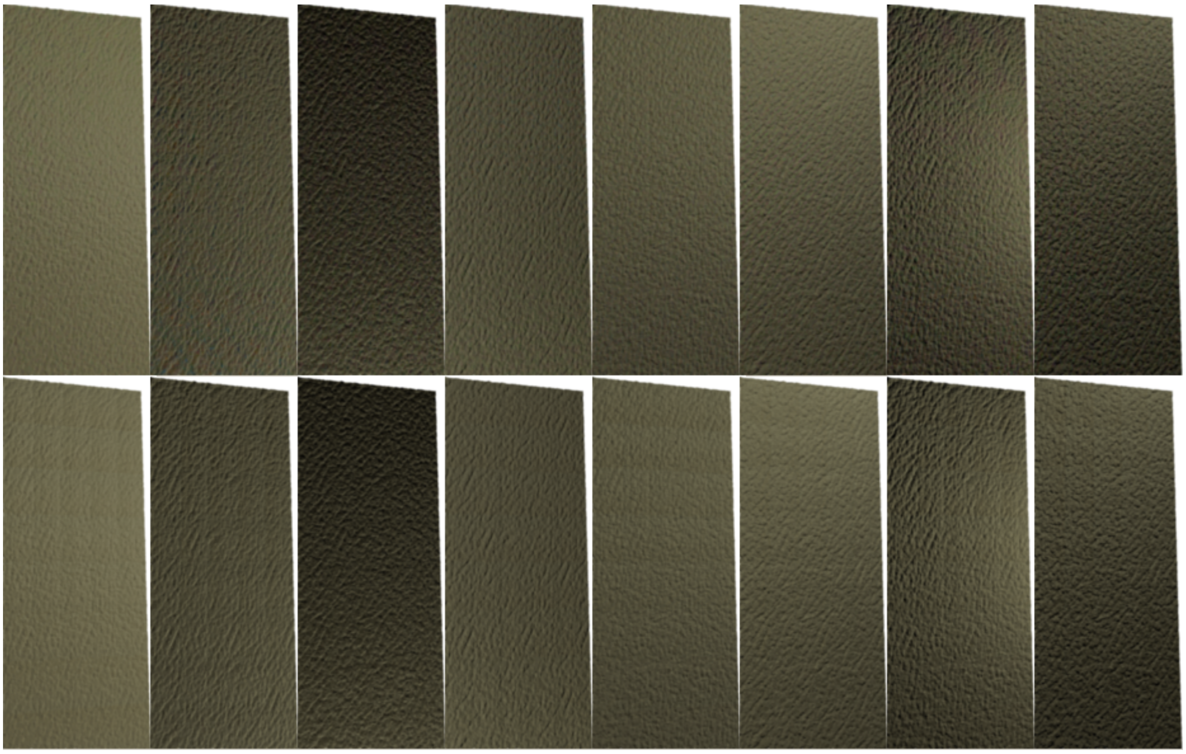

Figure 3: A curved plane with mapped BTF. Bottom row: the original measured BTF (artificial leather). Top row: the synthesised BTF. Each column represents one unique illumination condition. Camera stayed fixed for all shots.

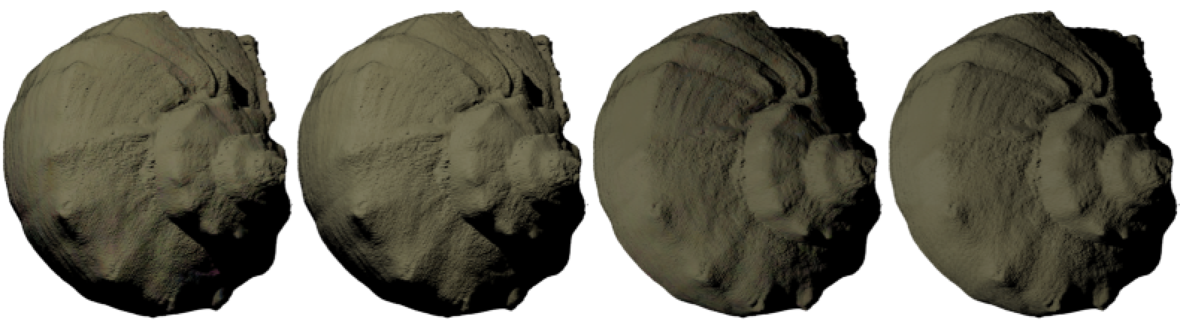

Figure 4: BTF mapped on complex geometry. The original measured BTF of artificial leather (2nd and 4th object from left) and corresponding (same illumination and view angles) synthesised BTF (1st and 3rd object from left).

once per material and offline. Both analysis and synthesis steps may be performed in parallel. Utilizing displacement mapping is both efficient (due to direct hardware support) and improve overall visual quality of the result. In addition, this model may be used to reconstruct BTF space, i.e., synthesize missing parts, previously unmeasured, of the BTF space. Oh the other hand the method is based on the mathematical model in contrast to intelligent sampling type methods and as such it can only approximate realism of the original measurement. The approximation strongly depends on several factors such as size and nature of training data and size of neighbourhood set.

6 Future Work

This BTF model might be further tested and compared with other random field based models. Overall texture visual quality comparison is complex and not yet completely solved problem. We would like to focus on texture overall colour quality comparison because direct pixel to pixel comparison (or based on texture geometry) seems to be inconvenient due to stochastic character of synthesised textures. One possibility might be Generalized Colour Moments [15].

Very interesting task would be extension of current implementation by means of parallel programming, for example with use of OpenMP[3] interface or other multithreading techniques (TBB[4], UPC[5]).

An extensive utilization of graphics processing unit seems to be applicable as well, but requires more sophisticated adaptation of current implementation where all computation is performed in the central processing unit. It would be possible to utilize framework OpenCL[6] or standard OpenGL[7]. Such improvements would notably increase overall performance which would be beneficial especially in case of virtual reality system requiring as fast as possible or even real time render and thus fast texture synthesis as well.

References

[1] J. Bennett, A. Khotanzad. *Multispectral Random Field Models for Synthesis and Analysis of Color Images.* IEEE Transactions on Pattern Analysis and Machine Intelligence **20**(3) (1998), 327–332.

[2] G. J. Burghouts, J. M. Geusebroek. *Material-specific Adaption of Color Invariant Features.* Pattern Recognition Letters **30** (2009), 306–313.

[3] K. Dana, S. Nayar, B van Ginneken, J. Koenderink. *Reflectance and Texture of Real-World Surfaces.* Proceedings of IEEE Conference Computer Vision and Pattern Recognition (1997), 151–157.

[3] http://openmp.org
[4] http://threadingbuildingblocks.org
[5] http://upc.gwu.edu
[6] www.khronos.org/opencl
[7] www.opengl.org

[4] J. De Bonet. *Multiresolution sampling procedure for analysis and synthesis of textured images.* Proceedings of SIGGRAPH 97, ACM (1997), 361–368.

[5] P. Favaro, S. Soatto. *3-D shape estimation and image restoration: exploiting defocus and motion blur.* Springer-Verlag New York Inc. (2007).

[6] J. Filip, J, M. Haindl. *Bidirectional texture function modeling: A state of the art survey.* IEEE Transactions on Pattern Analysis and Machine Intelligence **31**(11), (2009) 1921–1940.

[7] R. T. Frankot, R. Chellappa. *A method for enforcing integrability in shape from shading algorithms.* IEEE Trans. on Pattern Analysis and Machine Intelligence **10**(7), (1988) 439–451.

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

[9] M. Haindl, J. Filip. *A Fast Probabilistic Bidirectional Texture Function Model.* Proceedings of ICIAR (lecture notes in computer science 3212) **2**, Springer-Verlag, Berlin Heidenberg (2004), 298–305.

[10] M. Haindl, J. Filip, M. Arnold. *BTF Image Space Utmost Compression and Modelling Method.* Proceedings of 17th ICPR **3**, IEEE Computer Society Press (2004), 194–198.

[11] M. Haindl, J. Filip. *Fast BTF Texture Modeling.* Proceedings of the 3rd International Workshop on Texture Analysis and Synthesis (2003), 47–52.

[12] M. Haindl, M. Hatka. *BTF Roller.* Texture 2005: Proceedings of the 4th International Workshop on Texture Analysis and Synthesis (2005), 89–94.

[13] M. Haindl, M. Havlíček. *Bidirectional Texture Function Simultaneous Autoregressive Model.* Computational Intelligence for Multimedia Understanding, Lecture Notes in Computer Science **7252**, Springer Berlin / Heidelberg (2012), 149–159.

[14] M. Hatka. *Vizualizace BTF textur v Blenderu.* Doktorandské dny 2009, sborník workshopu doktorandů FJFI oboru Matematické inženýrství, České vysoké učení technické v Praze (2009), 37–46.

[15] F. Mindru, T. Tuytelaars, L. Van Gool, T. Moons. *Moment invariants for recognition under changing viewpoint and illumination.* Computer Visual Image Understanding **94**(1–3), Elsevier Science Inc., (2004), 3–27.

[16] G. Müller, J. Meseth, M. Sattler, R. Sarlette, R. Klein. *Acquisition, Compression, and Synthesis of Bidirectional Texture Functions.* State of the art report, Eurographics (2004), 69–94.

[17] X. Wang, X. Tong, S. Lin, S. Hu, B. Guo, H.-Y. Shum. *View-dependent displacement mapping.* ACM SIGGRAPH 2002 **22**(3), ACM Press (2003), 334–339.

[18] R. Woodham. *Photometric method for determining surface orientation from multiple images.* Optical engineering **19**(1), (1980) 139–144.