Bidirectional Texture Function Modeling

28

Michal Haindl

Contents

Introduction . . . 1025
Visual Texture . . . 1027
  Bidirectional Texture Function . . . 1027
  BTF Measurement . . . 1029
Compound Markov Model . . . 1030
Principal Markov Model . . . 1031
  Principal Single Model Markov Random Field . . . 1032
  Non-parametric Markov Random Field . . . 1032
  Non-parametric Markov Random Field with Iterative Synthesis . . . 1033
  Non-parametric Markov Random Field with Fast Iterative Synthesis . . . 1035
  Potts Markov Random Field . . . 1037
  Potts-Voronoi Markov Random Field . . . 1038
  Bernoulli Distribution Mixture Model . . . 1040
  Gaussian Mixture Model . . . 1041
Local Markov and Mixture Models . . . 1042
  3D Causal Simultaneous Autoregressive Model . . . 1042
  3D Moving Average Model . . . 1046
  Spatial 3D Gaussian Mixture Model . . . 1047
Applications . . . 1049
  Texture Synthesis and Enlargement . . . 1050
  Texture Compression . . . 1053
  Texture Editing . . . 1053
  Illumination Invariants . . . 1053
  (Un)supervised Image Recognition . . . 1054
  Multispectral/Multi-channel Image Restoration . . . 1056
Conclusion . . . 1057
References . . . 1058

M. Haindl (✉)
Institute of Information Theory and Automation, Czech Academy of Sciences, Prague, Czechia
e-mail: haindl@utia.cas.cz

K. Chen et al. (eds.), *Handbook of Mathematical Models and Algorithms in Computer Vision and Imaging*, https://doi.org/10.1007/978-3-030-98661-2_103

Abstract

An authentic material's surface reflectance function is a complex function of over 16 physical variables, which are unfeasible both to measure and to mathematically model. The best simplified measurable material texture representation and approximation of this general surface reflectance function is the seven-dimensional bidirectional texture function (BTF). BTF can be simultaneously measured and modeled using state-of-the-art measurement devices and computers and the most advanced mathematical models of visual data. However, such an enormous amount of visual BTF data, measured on the single material sample, inevitably requires state-of-the-art storage, compression, modeling, visualization, and quality verification. Storage technology is still the weak part of computer technology, which lags behind recent data sensing technologies; thus, even for virtual reality correct materials modeling, it is infeasible to use BTF measurements directly. Hence, for visual texture synthesis or analysis applications, efficient mathematical BTF models cannot be avoided. The probabilistic BTF models allow unlimited seamless material texture enlargement, texture restoration, tremendous unbeatable appearance data compression (up to 1:1000 000), and even editing or creating new material appearance data. Simultaneously, they require neither storing actual measurements nor any pixel-wise parametric representation. Unfortunately, there is no single universal BTF model applicable for physically correct modeling of visual properties of all possible BTF textures. Every presented model is better suited for some subspace of possible BTF textures, either natural or artificial. In this contribution, we intend to survey existing mathematical BTF models which allow physically correct modeling and enlargement measured texture under any illumination and viewing conditions while simultaneously offering huge compression ratio relative to natural surface materials optical measurements. Exceptional 3D Markovian or mixture models, which can be either solved analytically or iteratively and quickly synthesized, are presented. Illumination invariants can be derived from some of its recursive statistics and exploited in content-based image retrieval, supervised or unsupervised image recognition. Although our primary goal is physically correct texture synthesis of any unlimited size, the presented models are equally helpful for various texture analytical applications. Their modeling efficiency is demonstrated in several analytical and modeling image applications, in particular, on a (un)supervised image segmentation, bidirectional texture function (BTF) synthesis and compression, and adaptive multispectral and multi-channel image and video restoration.

Keywords

Bidirectional texture function · Texture modeling · Markov random fields · Discrete distribution mixtures · Expectation-Maximization algorithm

Introduction

Multidimensional data modeling or understanding (or set of spatially related objects) is more accurate and efficient if we respect all interdependencies between single objects. Objects to be processed, for example, multispectral pixels, in a digitized image are often mutually dependent (e.g., correlated) with a dependency degree related to a distance between two objects in their corresponding data space. These relations can be incorporated into a pattern recognition or visualization process through an appropriate multidimensional data model. If such a model is probabilistic, we can benefit from a consistent Bayesian framework for solving many related visual or pattern recognition tasks.

Features derived from multidimensional data models are information preserving in the sense that they can be used to synthesize data spaces closely resembling original measurement data space as can be illustrated on the recent best visual representation of real material surfaces in the form of seven-dimensional bidirectional texture function (Haindl and Filip 2007; Filip and Haindl 2009). Virtual or augmented reality systems require object surfaces covered with physically correct nature-like color textures to enhance realism in visual scenes applied in computer games, CAD systems, or other computer graphics applications. Surface material appearance modeling thus aims to generate and enlarge a synthetic texture visually indiscernible from the visual properties of measured material, whatever the observation conditions might be.

While simple color textures can be either digitized measured natural textures or textures synthesized from an appropriate mathematical model, realistic 7D BTF textures require mathematical modeling. Measured BTF textures are far less convenient alternative, because of extreme virtual system memory demands, limited size measurements, visible discontinuities (if we apply some usual computer graphics sampling approach for texture enlargement (De Bonet 1997; Efros and Freeman 2001; Praun et al. 2000; Xu et al. 2000; Wei and Levoy 2000, 2001; Liang et al. 2001; Soler et al. 2002; Dong and Chantler 2002; Zelinka and Garland 2002; Haindl and Hatka 2005a,b; Ngan and Durand 2006)), or several other drawbacks (Haindl 1991). Some of these methods are based on per-pixel sampling (Wei and Levoy 2001; Tong et al. 2002; Zelinka and Garland 2003; Zhang et al. 2003) while other are patch-based sampling methods (Praun et al. 2000; Xu et al. 2000; Efros and Freeman 2001; Liang et al. 2001; Soler et al. 2002; Kwatra et al. 2003; Dong et al. 2010). Texture synthesis algorithms (Heeger and Bergen 1995; Liu and Picard 1996; Efros and Leung 1999; Portilla and Simoncelli 2000) view surface texture as a stochastic process and aim to produce new realizations that resemble an input exemplar by either copying pixels (non-parametric methods) or matching image statistics (parametric techniques). Some of these simple gray scale/color texture modeling methods, which also allow texture enlargement, could be formally applied independently for each BTF material space. However, this is infeasible for all about a thousand measurements for a single BTF material due to their enormous computing time and memory constraints. Furthermore, for example, a car interior usually has about 20 different materials to synthesize.

Principle component analysis (PCA)-based BTF approximation (Müller et al. 2003; Sattler et al. 2003; Ruiters et al. 2013) allows BTF lossy compression but not enlargement. Furthermore, projecting the measured data onto a linear space constructed by statistical analysis such as PCA results in low-quality data compression. Another compression method (Tsai and Shih 2012) is based on K-clustered tensor approximation or the polynomial wavelet tree (Baril et al. 2008).

BTF data can be approximated using separate texel models, i.e., spatially varying bidirectional reflectance distribution function (SVBRDF) models that combine texture mapping and BRDF models but sacrifice some spatial dependency information. A linear combination of multivariate spherical radial basis functions is used to model BTF as a set of texelwise BRDFs (SVBRDF) in Tsai et al. (2011). Another SVBRDF method (Wu et al. 2011) uses a parametric mixture model with a basis analytical BRDF function for texel modeling. Several SVBRDF models use multilayer perceptron neural networks (Aittala et al. 2016; Deschaintre et al. 2018; Rainer et al. 2020). A deep convolutional neural network VGG-19 is used in Aittala et al. (2016), while the convolutional neural network recovers SVBRDF from estimated normal, diffuse albedo, specular albedo, and specular roughness from a single image lit by a handheld flash in Deschaintre et al. (2018). A learned SVBRDF decoder in a multilayer perceptron neural model approximates BRDF values in Rainer et al. (2020). The SVBRDF methods approximate BTF quality, are computationally expensive due to the nonlinear optimization, allow only moderate compression ratio, require several manually tuned parameters, and do not allow BTF space enlargement.

Mathematical multidimensional data models are useful for describing many of the multidimensional data types provided that we can assume some data homogeneity, so some data characteristics are a translation invariant. While the 1D models like time series (Anderson 1971; Broemeling 1985) are relatively well researched, and they have a rich application history in control theory, econometrics, medicine, meteorology, and many other data mining or machine learning applications, multidimensional models are much less known (e.g., more than three-dimensional MRF), and their applications are still limited. The reason is not only unsolved theory difficulties but mainly their vast computing power demands, which prevented their more extensive use until recently.

Visual data models need nonstandard multidimensional (three-dimensional for static color textures, four-dimensional for videos, or even seven-dimensional for static BTFs) models. However, if such a nD data space can be factorized, then these data can also be approximated using a set of lower-dimensional probabilistic models. Although full visual nD models allow unrestricted spatial-spectral-temporal-angular correlation modeling, their main drawback is many parameters to be estimated, which require a correspondingly large learning set. In some models (e.g., Markov models), the necessity is to estimate all these parameters simultaneously.

We introduced (Haindl and Havlíček 1998, 2000, 2010, 2016, 2017b, 2018a,b; Haindl et al. 2012, 2015b), several efficient fast multiresolution Markov random field (MRF)-based models which exploit BTF space factorization. Our methods avoid the time-consuming Markov chain Monte Carlo simulation (MCMC) so typical for Markov models applications with one exception of the Potts MRF. Our models avoid some problems of alternative options (see Haindl 1991 for details), but they are also easy to analyze as well as to synthesize, and last but not least, they are still flexible enough to correctly imitate a broad set of natural and artificial textures or other spatial data.

We can categorize the model's applications into synthesis and analysis. Analytical applications include static or dynamic data un-/semi-/supervised recognition, scene understanding, data space analysis, motion detection, and numerous others. Typical synthesis applications are missing data reconstruction, restoration, image compression, and static or dynamic texture synthesis.

Visual Texture

The visual texture notion is closely tied to the human semantic meaning of surface material appearance, and texture analysis is an essential and frequently published area of image processing. However, there is still no mathematically rigorous definition of the texture that would be accepted throughout the computer vision community.

We understand a textured image or the *visual texture* (Haindl and Filip 2013) to be a realization of a random field, and our effort is to find its parameterizations in such a way that the real texture representing the specific material appearance measurements will be visually indiscernible from the corresponding random field's realization, whatever the observation conditions might be. Some work distinguishes between texture and color. We regard such separation between spatial structure and spectral information to be artificial and principally wrong because there is no bijective mapping between gray scale and multispectral textures. Thus, our random field model is always multispectral.

Bidirectional Texture Function

A natural material's surface general reflectance function (GRF), representing physically correct visual properties of surface materials and their variations under any observation conditions, is a complex function of 16 physical variables. It is currently unfeasible to measure or to model such a function mathematically. Practical applications thus require significant simplification, namely, using additional assumptions. These approximative assumptions neglect the most less significant variables to achieve a solvable problem, with the solution still far more realistic

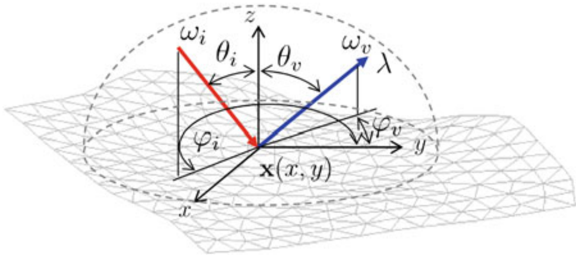

Fig. 1 BTF reflectance model

than the traditional three-dimensional static color texture representation. BTF can model complex lighting effects such as self-shadows, masking, foreshortening, interreflections, and multiple subsurface light scattering due to material surface microgeometry.

The seven-dimensional bidirectional texture function (BTF) reflectance model Fig. 1 is the best recent visual texture representation, which can still be simultaneously measured and modeled using state-of-the-art measurement devices and computers as well as the most advanced mathematical models of visual data. Thus, it is the most important representation for the high-end and physically correct surface materials appearance modeling. Nevertheless, BTF requires the most advanced modeling as well as high-end hardware support. The BTF reflectance model

$$Y_r^{BTF} = BTF(\lambda, x, y, \theta_i, \varphi_i, \theta_v, \varphi_v), \tag{1}$$

where Y_r^{BTF} is a random spectral reflectance vector at location r, r is a multiindex, and Y_r^{BTF} accepts six simplifying assumptions from GRF – light transport in material takes zero time ($t_i = t_v$ (incident time is equal to the reflection time) and $t_v = \emptyset$), reflectance behavior of the surface is time invariant ($t_v = t_i = const., t_v = t_i = \emptyset$); interaction with the material does not change wavelength ($\lambda_i = \lambda_v$), i.e., $\lambda_v = \emptyset$), constant radiance along light rays ($z_i = z_v = \emptyset$), no transmittance ($\theta_t = \varphi_t = \emptyset$), and incident light leaves at the same point.

Multispectral BTF is a seven-dimensional random function, which considers measurement dependency on color spectrum and planar material position, as well as its dependence on illumination incident light (lower index i) and viewing reflection light (lower index v) angles $BTF(r, \theta_i, \phi_i, \theta_v, \phi_v)$, where the multiindex $r = [r_1, r_2, r_3]$ specifies planar horizontal and vertical position in material sample image, r_3 is the spectral index, and θ, ϕ are elevation and azimuthal angles of the illumination and view direction vectors. The BTF measurements comprise a whole the hemisphere of light and camera positions in observed material sample coordinates according to selected quantization steps, and this is the main difference compared to the standard three-dimensional static color texture. This difference significantly improves the visual quality and realism of BTF representation and simultaneously complicates its measurement and modeling.

BTF Measurement

Accurate and reliable BTF acquisition is not a trivial task; only a few BTF measurement systems currently exist (for details see Haindl and Filip 2013; Schwartz et al. 2014; Dana et al. 1997; Koudelka et al. 2003; Sattler et al. 2003; Han and Perlin 2003; Müller et al. 2004; Wang and Dana 2006; Ngan and Durand 2006; Debevec et al. 2000; Marschner et al. 2005; Holroyd et al. 2010; Ren et al. 2011; Aittala et al. 2013, 2015). However, their number increases every year in response to the growing demand for photorealistic virtual representations of real-world materials. These systems are (similar to bidirectional reflectance distribution function (BRDF) measurement systems) based on the light source, video/still camera, and material sample. The main difference between individual BTF measurement systems is in the type of measurement setup allowing four degrees of freedom for camera/light, the type of measurement sensor (CCD, video, and some other), and light.

In some systems, the camera is moving, and the light is fixed (Dana et al. 1997; Sattler et al. 2003; Neubeck et al. 2005), while in others, e.g., Koudelka et al. (2003), it is just the opposite. There are also systems where both camera and light source remain fixed (Han and Perlin 2003; Müller et al. 2004).

The UTIA gonioreflectometer setup Fig. 2 consists of independently controlled arms with a camera and light. Its parameters, such as angular precision 0.03 degree, spatial resolution 1000 DPI, or selective spatial measurement, classify this

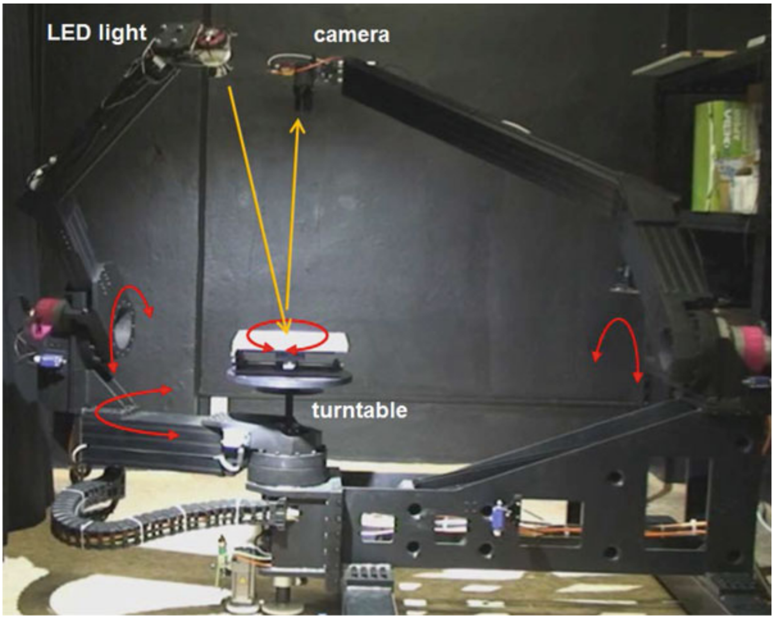

Fig. 2 UTIA gonioreflectometer

gonioreflectometer to the state-of-the-art devices. The typical resolution of the area of interest is around 2000×2000 pixels, sample size 7×7 [cm], and sensor distance ≈ 2 [m] with a field of view angle of $8.25°$, and each of them is represented using at least 16-bit floating-point value for a reasonable representation of high-dynamic-range visual information. Illumination source is 11 LED arrays, each having a flux of 280 lm at 0.7 A, spectral wavelength $450 - 700$ [nm], and its optics. The memory requirements for storage of a single material sample amount to 360 gigabytes per color channel but can be much more for a more precise spectral measurement.

We measure each material sample mostly in 81 viewing positions n_v and 81 illumination positions n_i, resulting in 6561 images per sample (4 terabytes of data).

Compound Markov Model

BTF data space is seven-dimensional, and thus it also requires seven-dimensional probabilistic models for physically correct BTF modeling, data compression, and enlargement with all related problems needed for robust estimation of all their numerous parameters. A practical alternative is to factorize a seven-dimensional problem into a set of lower-dimensional models with fewer parameters dedicated to model subparts of a BTF texture combined into a compound BTF model.

We exploit the compound Markov model for physically correct BTF modeling for either synthesis or analytical applications. Let us denote a multiindex $r = (r_1, r_2)$, $r \in I$, where I is a discrete two-dimensional rectangular lattice and r_1 is the row and r_2 the column index, respectively. The principal field pixel $X_r \in \mathscr{K}$ where $\mathscr{K}$ is the index set of K distinguished sub-models, i.e., $X_r \in \{1, 2, \ldots, K\}$ is a random variable with natural number value (a positive integer). Y_r is the multispectral pixel at location r and $Y_{r,j} \in \mathscr{R}$ is its j-th spectral plane component. Both random fields (X, Y) are indexed on the same $M \times N$ lattice I.

Let us assume that each multispectral observed texture $\tilde{Y}$ (composed of d spectral planes, e.g., $d = 3$ for color textures) and indexed on the $\tilde{M} \times \tilde{N}$ lattice $\tilde{I}$ (usually $\tilde{I} \subseteq I$ and $\tilde{M}, \tilde{N}$ are number of rows and columns of the measured BTF texture) can be modeled by a compound Markov random field model (CMRF), where the principal Markov random field (MRF) X controls switching to a regional local MRF model ${}^{i}Y$ where $Y = \bigcup_{i=1}^{K} {}^{i}Y$. Single K regional random field sub-models ${}^{i}Y$ are defined on their corresponding lattice subsets ${}^{i}I$, ${}^{i}I \cap {}^{j}I = \emptyset \;\; \forall i \neq j$, $I = \bigcup_{i=1}^{K} {}^{i}I$ $(X_r = X_s \; \forall r, s \in^{i} I)$ and they are of the same MRF type. These models differ only in their contextual support set ${}^{i}I_r$ and corresponding parameter sets ${}^{i}\theta$ (a set of all i-th local random field parameters). The same type of sub-models are assumed only for simplicity and can be omitted without any problems if needed. The BTF-CMRF model has a posterior probability

$$P(X, Y \mid \tilde{Y}) = P(Y \mid X, \tilde{Y}) P(X \mid \tilde{Y}) \tag{2}$$

and the corresponding optimal maximum a posteriori (MAP) solution is

$$(\hat{X}, \hat{Y}) = \arg \max_{X \in \Omega_X, Y \in \Omega_Y} P(Y \mid X, \tilde{Y})\, P(X \mid \tilde{Y}),$$

where Ω_X, Ω_Y are the corresponding configuration spaces for both random fields (X, Y). To avoid an iterative MCMC MAP solution for parameter estimation, we proposed the following two-step approximation $\breve{X}, \breve{Y}$ (Haindl and Havlíček 2010):

$$(\breve{X}) = \arg \max_{X \in \Omega_X} P(X \mid \tilde{Y}), \tag{3}$$

$$(\breve{Y}) = \arg \max_{Y \in \Omega_Y} P(Y \mid \breve{X}, \tilde{Y}). \tag{4}$$

This approximation significantly simplifies the BTF-CMRF estimation without compromising random sampling for its synthesis because it allows us to take advantage of the possible analytical estimation of all regional MRF models ${}^{i}Y$ in (4). We randomly sample the required enlarged texture in the same order, i.e., at first (3) and, consequently, based on this principal random field realization, the local random fields (4). Furthermore, there is no need to have a unique solution of the (3), (4) approximation because the aim is to obtain a visually indiscernible result or results from the target observation. The subsequent Markovian/mixture compound models use the notation BTF-CMRF$^{principal_model\ local_model}$ where the upper indices indicate the principal as well as the local model families.

Principal Markov Model

The principal part (X) of the BTF compound Markov models (BTF-CMRF) is assumed to be independent on illumination and observation angles, i.e., it is identical for all possible combinations $\phi_i, \phi_v, \theta_i, \theta_v$ azimuthal and elevation illumination/viewing angles, respectively. This assumption does not compromise the resulting BTF space quality because it influences only a material texture macrostructure independent of these angles for static BTF textures.

The principal random field $\breve{X}$ is estimated using simple K-means clustering of $\tilde{Y}$ in the RGB color space into a predefined number of K classes, where cluster indices are $\breve{X}_r \quad \forall r \in I$ estimates. We further use for simplicity the RGB color space, but any other color space can be used as well. The number of classes K can be estimated using the Kullback-Leibler divergence and considering a sufficient amount of data necessary to estimate all local Markovian models reliably. If the BTF texture contains subparts with distinct texture but similar colors, any more sophisticated texture segmenter (e.g., Haindl and Mikeš 2007; Haindl et al. 2009a,b, 2015a) can be used.

Principal Single Model Markov Random Field

The simplest principal model is a constant field that contains only one model BTF-CMRF$^{c\cdots}$ $P(X \mid \tilde{Y}) = const.$, i.e., $P(X_r \mid \tilde{Y}) = P(X_s \mid \tilde{Y}) \quad \forall r, s$. Then there is no need to use the MAP approximation (3), (4), and the compound Markov model simplifies into a single random field BTF-MRF model, and the BTF-MRF model can be any of the following local MRF models.

Non-parametric Markov Random Field

If we do not assume any specific principal control field parametric model, but rather we seamlessly and directly enlarge its realization from measured data (Fig. 3), we get several non-parametric principal control field approaches. The non-parametric principal field BTF-CMRF$^{NProl\ldots}$ (NProl...– a non-parametric roller-based principal field with any local random fields denoted as ...; see Figs. 3, 4, 16) can be modeled using the roller method (Haindl and Havlíček 2010) for optimal $\breve{X}$ compression and speedy enlargement to any required field size. The roller method (Haindl and Hatka 2005a,b) principle is the overlapping tiling and subsequent minimum error boundary cut. One or several optimal double toroidal data patches are seamlessly and randomly repeated during the synthesis step. This fully automatic method starts with minimal tile size detection, which is limited by the size of the principal field, the number of toroidal tiles we are looking for, and the sample spatial frequency content.

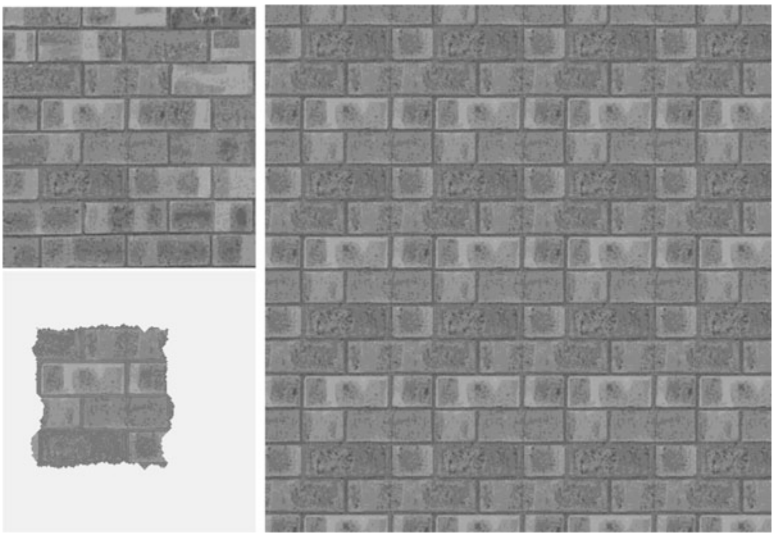

Fig. 3 Measured brick principal field (upper left), its optimal double toroidal patch (bottom left), and enlarged synthetic principal field (right, $K = 8$)

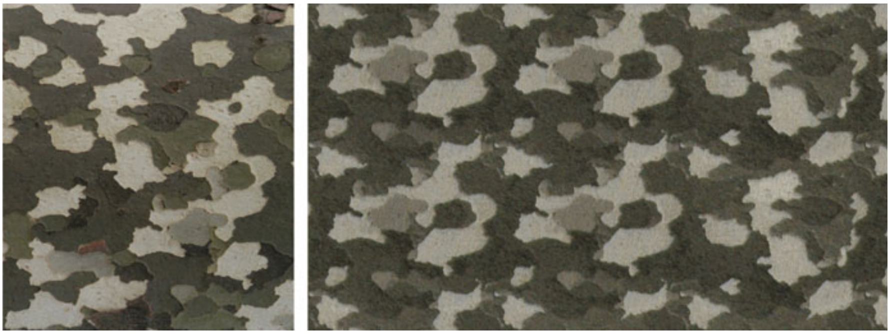

Fig. 4 Synthetic BTF-CMRF^{NProl3DCAR} enlarged color bark (right) estimated from their natural measurements (left)

Non-parametric Markov Random Field with Iterative Synthesis

The non-parametric principal random field $\breve{X}$ is estimated using simple K-means clustering of $\tilde{Y}$ in the RGB color space into a predefined number of K classes, where cluster indices ω_i are $\breve{X}_r \quad \forall r \in I$ estimates. The clustering resulting thematic map is used to compute region size histograms $\tilde{h}_i$ for all $i = 1, \ldots, K$ classes. Let us order classes according to the decreasing number of pixels $\tilde{n}_i$ belonging to each class, i.e., $\tilde{n}_1 \geq \tilde{n}_2 \geq \ldots \geq \tilde{n}_K$. Histograms $\tilde{h}_i$ are the only parameters required to store for the principal field.

Iterative Principal Field Synthesis

The iterative algorithm (Haindl and Havlíček 2018b) (Figs. 5 and 6) uses a data structure that describes membership in the region for each pixel. This data structure for each region additionally contains the class membership, size of the region and the requested number of regions of its size, all border pixels from both sides of the border, possibility to decrease or increase the region, and, for all classes, the histogram and regions, which can be increased or decreased. After any change in a pixel class assignment, this structure has to be updated.

0. The synthesized $M \times N$ required principal field is initialized to the largest class, and all histograms cells are rescaled using the scaling factor $\frac{MN}{\tilde{M}\tilde{N}}$, where $\tilde{M} \times \tilde{N}$ is the target (measured) texture size, i.e., $X_r^{(0)} = \omega_1 \quad \forall r \in I$ and $\tilde{h}_i \rightarrow h_i$ for $i = 1, \ldots, K$. A lattice multiindex r is randomly generated starting from the second-largest class ω_2 till the smallest size class ω_K. Class index X_r is changed to new value $X_r = \omega_i$ only if its previous value was $X_r = \omega_1$ and the total number of principal field pixels with class indicator ω_i is smaller than its final value n_i. After this initialization step, all classes have their correct required number of pixels but not yet their correct region size histograms.

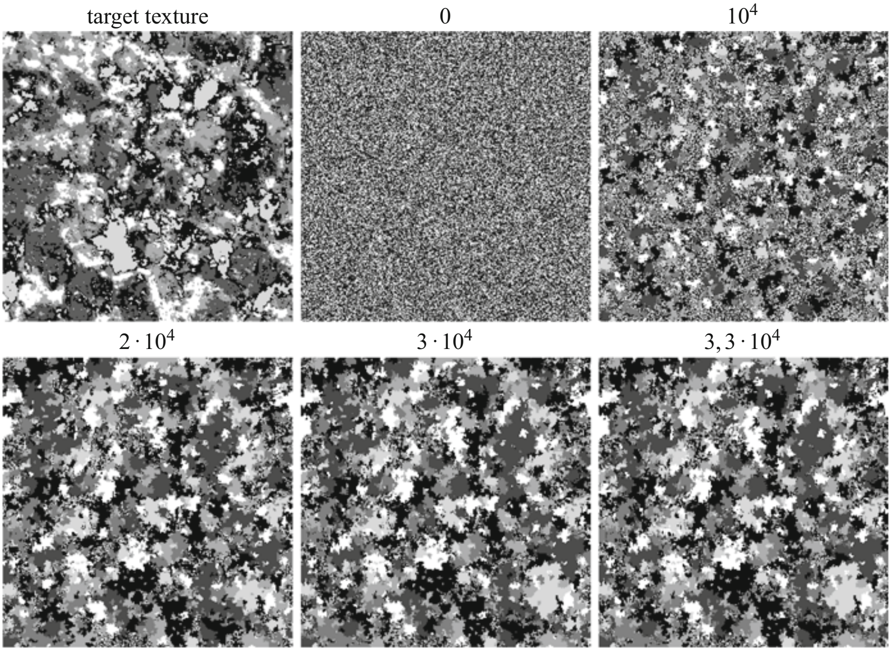

Fig. 5 The granite (Fig. 6) principal field synthesis. The target texture principal field, initialization, and selected iteration steps rightwards

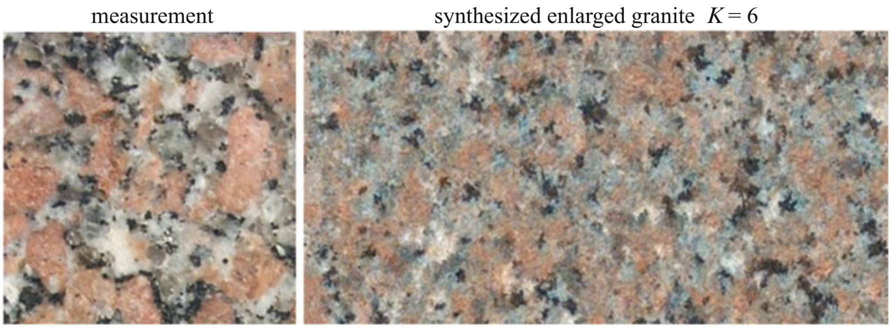

Fig. 6 The granite measurement and its synthetic enlargement (BTF-CMRF^{NPi3AR})

1. Pixels r and s are randomly selected with the following properties: The pixel r from the class ω_i is on the border between region $\downarrow \omega_i^A$ (a region A which can be decreased) and region $\uparrow \omega_j^B$ (a region B which can be increased). The pixel s from the class ω_j is on the border between region $\downarrow \omega_j^C$ (a region C which can be decreased) and region $\uparrow \omega_i^D$ (a region D which can be increased). These regions have to be distinct, i.e., $A \cap D = \emptyset$ and $B \cap C = \emptyset$. If such pixels r, s exist, go to step 5. If not repeat this step once more.

2. Gradually check all class couples starting from $\omega_1, \omega_2, \ldots, \omega_K$ to find pixels r, s which meet conditions in step 1. All regions corresponding to the chosen classes, ω_i and ω_j, are selected randomly. If such pixels r, s exist, go to step 5.
3. Randomly select a region from class ω_i, which has two neighboring regions of class ω_j such as one can be decreased and another increased. If there exist two border pixels r, s in the region ω_j, where r is a border pixel with a region to be increased and s with a region to be decreased, go to step 5.
4. Gradually check all classes with incorrect histogram, starting from $\omega_1, \omega_2, \ldots,$ ω_K; for every class ω_i gradually check all its regions $\uparrow \omega_i^A$ which can be increased; for each region $\uparrow \omega_i^A$, check every region neighboring border pixel r from class ω_j and region $\downarrow \omega_j^B$ (a region B which can be decreased), and find pixel s with the following properties: pixel s is from the class ω_i and region $\downarrow \omega_i^C$ (a region C which can be decreased), and pixel s is on the boarder of the region $\uparrow \omega_j^D$ from class ω_j (a region which can be increased). These regions have to be distinct, i.e., $A \cap C = \emptyset$ and $B \cap D = \emptyset$. If such pixels do not exist, go to step 7.
5. $X_r = \omega_j, X_s = \omega_i$ update the data structure.
6. If the number of iterations is less than a selected limit, go to 1.
7. Store the resulting principal field and stop.

Steps 1 and 2 allow simultaneous improvement of four regions, while step 3 improves two regions only. The algorithm converges to the correct class histograms $h_i \;\; i = 1, \ldots, K$.

Non-parametric Markov Random Field with Fast Iterative Synthesis

The non-parametric principal field (Haindl and Havlíček 2018a) BTF-CMRF$^{NPfi...}$ is estimated as in the previous section, and its synthesis is modified to be significantly faster at the cost of slightly compromised principal field variability. The fast algorithm compromise is its preference for convex regions instead of their general shapes but profits with faster convergency.

The median speed up between this method and the approach for the non-parametric principal field synthesis in section "Non-parametric Markov Random Field with Iterative Synthesis" is one-fifth of the required cycles to converge. Some textures (e.g., granite; Fig. 7) have sufficiently similar statistics of the synthesized regions with the principal target field already in the initialization step. Hence, the principal field synthesis even does not need any iterations. The lichen Fig. 8 principal target field (512 × 512) requires 29 137 iterations, while the previous iterative method needs nearly 5 times more (140 146) iterations to converge.

Iterative Principal Field Synthesis

The iterative algorithm is based on a similar data structure, which describes membership in the region for each pixel, as in the previous section. Both iterative algorithms differ only in their initialization steps.

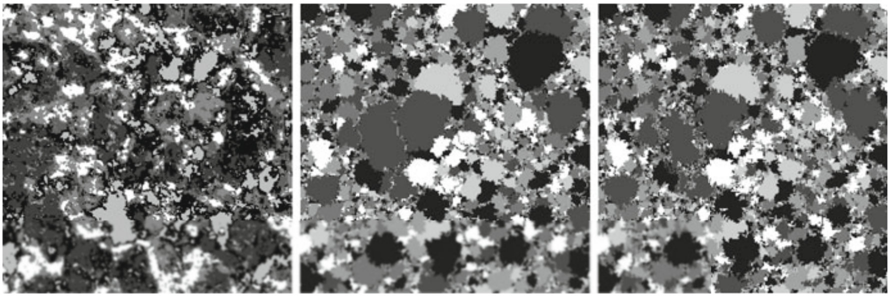

Fig. 7 The granite principal field synthesis. The target texture principal field, initialization, and a similar 10^4-th iteration step result

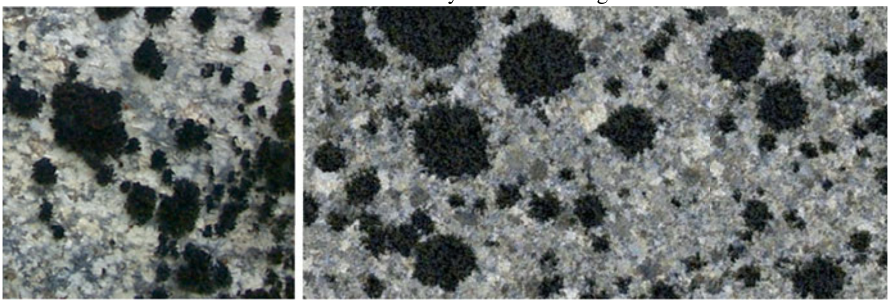

Fig. 8 The lichen measurement and its synthetic enlargement (BTF-CMRF^{NPfi3DCAR})

0. The synthesized $M \times N$ required principal field is initialized to the value ω_0 it means that pixel was not assigned to any class ω_i for $i = 1, \ldots, K$. All histogram cells are rescaled using the scaling factor $\frac{MN}{\tilde{M}\tilde{N}}$, i.e., $X_r^{(0)} = \omega_1 \;\; \forall r \in I$ and $\tilde{h}_i \to h_i$ for $i = 1, \ldots, K$. All regions from all classes $i = 1, \ldots, K$ are sorted by region size. Starting from the biggest region A_1 till the smallest region A_M, where M the is number of all regions, a lattice multiindex r is randomly generated. The first pixel X_r of the region A_j where $j = 1, \ldots, M$ and class ω_i is randomly selected and is changed to new value $X_r = \omega_i$ only if its previous value was $X_r = \omega_0$ All neighbors X_s of the pixel X_r which fulfil conditions $X_s = \omega_0$ and pixel X_s that has no neighbor from the class ω_i are added to the queue Q. Till the size of region A_j is higher than the number of actually added pixels, the next pixel X_r is randomly selected from the queue Q, the values are changed to $X_r = \omega_i$ and its neighbors are added to the queue Q if they meet the mentioned conditions. If the queue Q is empty and the size of the region A_j is higher than the number of actually assigned pixels, the rest of the pixels is randomly assigned to the class ω_i after the initialization of the last region A_M. After this initialization step,

all classes have their correct required number of pixels but not their correct region size histograms.

1.–7. Identical with the corresponding items in section "Iterative Principal Field Synthesis".

Steps 1 and 2 allow simultaneous improvement of four regions, while step 3 improves two regions only. The algorithm converges to the correct class histograms $h_i \;\; i = 1, \ldots, K$.

Potts Markov Random Field

The resulting thematic principal map $\breve{X}$ BTF-CMRF$^{2P\cdots}$ is represented by the hierarchical two-scale Potts model (Haindl et al. 2012)

$$\breve{X}^{(a)} = \frac{1}{Z^{(a)}} \exp\left\{ -\beta^{(a)} \sum_{s \in I_r} \delta_{X_r^{(a)} X_s^{(a)}} \right\} \tag{5}$$

where Z is the appropriate normalizing constant and $\delta()$ is the Kronecker delta function. The rough-scale-upper- level Potts model ($a = 1$) regions are further elaborated with the detailed fine-scale-level ($a = 2$) Potts model which models the corresponding subregions in each upper-level region. The parameter $\beta^{(a)}$ for both level models is estimated using an iterative estimator which starts from the upper β limit ($\beta_{\max}$) and adjusts (decreases or increases) its value until the Potts model regions have similar parameters (average inscribed squared region size and/or the region's perimeter) with the target texture switching field. This iterative estimator gives more resembling results with the target texture than the alternative maximum pseudo-likelihood method (Levada et al. 2008). The corresponding Potts models are synthesized (Fig. 9 – middle) using the fast Swendsen-Wang sampling method (Swendsen and Wang 1987).

Fig. 9 The rusty plate texture measurement, its principal synthetic field, and the final synthetic CMRF^{P3AR} model texture

Potts-Voronoi Markov Random Field

The principal field (X) of the CMRF BTF-CMRF$^{PV\cdots}$ model (Haindl et al. 2015b) is a mosaic represented as a Voronoi diagram (Aurenhammer 1991), and the distribution of the particular colors (texture classes) of the mosaic is modeled as a Potts random field which is built on top of the adjacency graph (G) of the mosaic. Figure 10 illustrates this model applied to the floor mosaic, while Fig. 11 shows this model applied to a glass mosaic synthesis in St. Vitus Cathedral in Prague Castle. The algorithm requires input in the form of a segmented mosaic with distinguishable regions of the same texture type.

After that follows the identification of the mosaic field centers and the estimation of the parameters of the 2D discrete point process, which samples the control points of the newly synthesized Voronoi mosaic. This sampling is done using a 2D histogram, which has shown to be sufficient for the good quality estimate. The only other parameter is the number of points to be sampled, which grows linearly with the required area of the synthetic image in the case of texture enlargement applications.

With the control points for the Voronoi mosaic cells having been sampled, we compute the Voronoi diagram, and optionally mark the delimiting edges between adjacent cells. The assignment of a regional texture model to each mosaic cell (the principal MRF $(P(X \mid \tilde{Y}))$) is then mapped by the flexible $K-$state Potts random field (Potts and Domb 1952; Wu 1982).

Let us denote $G = (V, E)$ the adjacency graph of the mosaic areas and

$$N_u = \{\forall v \in V : (u, v) \in E\}, \; u \in V \tag{6}$$

the 1st-order neighborhood, where V, E are the vertex and edge sets. Vertexes correspond to the particular areas in the mosaic, and there is an edge between two vertexes if their corresponding areas are directly next to each other.

The resulting thematic principal map $\check{X}$ is represented by the Potts model for a general graph

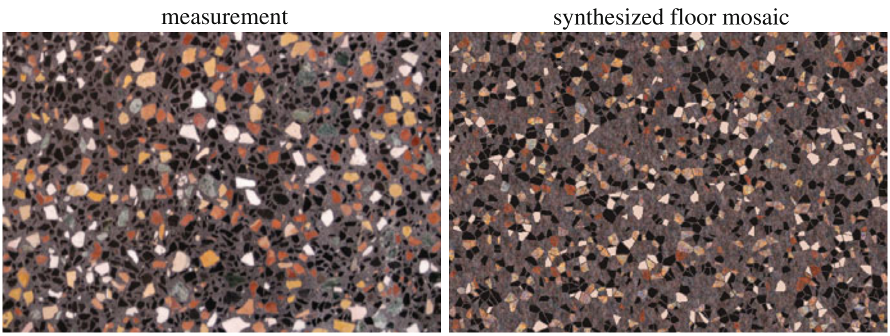

Fig. 10 The floor mosaic measurement and its synthesis (BTF-CMRF^{PV3DCAR})

Fig. 11 An example of St. Vitus Cathedral in Prague Castle stained glass window with two original panels (yellow arrows) replaced with synthetic images (BTF-CMRF^{PV3DCAR})

$$p(\breve{X}|\beta) = \frac{1}{Z} \exp\left\{ -\beta \sum_{u \in V, v \in N_u} \delta(X_u, X_v) \right\} \tag{7}$$

where Z is the appropriate normalizing constant and $\delta()$ is the Kronecker delta function. The parameter β is estimated from the K-means clustered input mosaic using the maximum pseudo-likelihood method described by Levada et al. (2008). The local density of the Potts field can be expressed as

$$p(X_u = q|X_{v \in N_u}, \beta) = \frac{\exp\left\{\beta \sum_{s \in N_u} \delta(q, X_v)\right\}}{\sum_{k=1}^{K} \exp\left\{\beta \sum_{v \in N_u} \delta(k, X_v)\right\}} \tag{8}$$

for which the pseudo-likelihood approximation is

$$PL(\beta) = \prod_{u \in V} p(X_u = q|X_{v \in N_u}, \beta). \tag{9}$$

Calculating the logarithm, differentiating, and setting the result equal to 0, we get the maximum pseudo-likelihood equation (10) for the β estimate:

$$\Psi(\beta) = -\sum_{u \in V} \frac{\sum_{k=1}^{K} \left(\sum_{v \in N_u} \delta(X_u, X_v) \right) \exp\left\{ \beta \sum_{v \in N_u} \delta(k, X_v) \right\}}{\sum_{k=1}^{K} \exp\left\{ \beta \sum_{v \in N_u} \delta(k, X_v) \right\}} + \sum_{u \in V} \sum_{v \in N_u} \delta(X_u, X_v) = 0. \tag{10}$$

The corresponding Potts models are synthesized using the fast Swendsen-Wang sampling method (Swendsen and Wang 1987), although for smaller fields, which the mosaics undoubtedly are, other sampling MCMC methods such as the Gibbs sampler (Geman and Geman 1984) can be used. Alternatively, the Metropolis algorithm (Metropolis et al. 1953) should also work sufficiently fast enough.

Bernoulli Distribution Mixture Model

The distribution $P(X_{\{r\}})$ is assumed to be multivariable Bernoulli mixture (BM) (Haindl and Havlíček 2017b). The mixture distribution $P(X_{\{r\}})$ has the form

$$P(X_{\{r\}}) = \sum_{m \in \mathscr{M}} P(X_{\{r\}} \mid m)\, p(m) = \sum_{m \in \mathscr{M}} \prod_{s \in I_r} p_s(Y_s \mid m)\, p(m), \tag{11}$$

where $\mathscr{M}$ is set of all mixture components, m a mixture component index, $\{r\}$ is a set of indices from I_r, and the principal field BTF-CMRF$^{BM\cdots}$ is further decomposed into separate binary bit planes of binary variables $\xi \in \mathscr{B}$, $\mathscr{B} = \{0, 1\}$ which are separately modeled and can be learned from much smaller training texture than a multi-level discrete mixture model (see examples in Fig. 14). We suppose that a bit factor of a principal field can be fully characterized by a marginal probability distribution of binary levels on pixels within the scope of a window centered around the location r and specified by the index set $I_r \subset I$, i. e., $X_{\{r\}} \in \mathscr{B}^{\eta}$ and $P(X_{\{r\}})$ is the corresponding marginal distribution of $P(X \mid \tilde{Y})$. The component distributions $P(\cdot \mid m)$ are factorizable, and multivariable Bernoulli

$$P(X_{\{r\}} \mid m) = \prod_{s \in I_r} \dot{\theta}_{m,s}^{X_s} (1 - \dot{\theta}_{m,s})^{1 - X_s} \qquad X_s \in X_{\{r\}}. \tag{12}$$

The mixture model parameters (11), (12) include component weights $p(m)$ and the univariate discrete distributions of binary levels. They are defined by one parameter $\dot{\theta}_{m,s}$ as a vector of probabilities:

$$p_s(\cdot \mid m) = (\dot{\theta}_{m,s}, 1 - \dot{\theta}_{m,s}). \tag{13}$$

The EM solution is (14), (15):

$$q^{(t)}(m \mid X_{\{r\}}) = \frac{p^{(t)}(m)\, P^{(t)}(X_{\{r\}} \mid m)}{\sum_{j \in \mathscr{M}} p^{(t)}(j)\, P^{(t)}(X_{\{r\}} \mid j)}, \tag{14}$$

$$p^{(t+1)}(m) = \frac{1}{|\mathscr{S}|} \sum_{X_{\{r\}} \in \mathscr{S}} q^{(t)}(m \mid\ X_{\{r\}}), \tag{15}$$

and

$$p_s^{(t+1)}(\xi \mid m) = \frac{1}{|\mathscr{S}|\, p^{(t+1)}(m)} \sum_{X_{\{r\}} \in \mathscr{S}} \delta(\xi, X_s)\, q^{(t)}(m \mid X_{\{r\}}), \quad \xi \in \mathscr{B}. \tag{16}$$

The total number of mixture (11), (13) parameters is thus $\dot{M}(1+\eta)$ $\dot{M} \in \mathscr{M}$ – confined to the appropriate norming conditions. The advantage of the multivariable Bernoulli model (13) is a simple switchover to any marginal distribution by deleting superfluous terms in the products $P(X_{\{r\}} \mid m)$.

Gaussian Mixture Model

The discrete principal field can be alternatively modeled (Haindl and Havlíček 2017b) by a continuous RF BTF-CMRF$^{GM\cdots}$ if we map single indices into continuous random variables with uniformly separated mean values and small variance. The synthesis results are subsequently inversely mapped back into a corresponding synthetic discrete principal field. We assume the joint probability distribution $P(X_{\{r\}})$, $X_{\{r\}} \in \mathscr{K}^{\eta}$ in the form of a normal mixture, and the mixture components are defined as products of univariate Gaussian densities

$$P(X_{\{r\}} \mid \mu_m, \sigma_m) = \prod_{s \in I_r} p_s(X_s \mid \mu_{ms}, \sigma_{ms}), \tag{17}$$

$$p_s(X_s \mid \mu_{ms}, \sigma_{ms}) = \frac{1}{\sqrt{2\pi}\sigma_{ms}} \exp\left\{ -\frac{(X_s - \mu_{ms})^2}{2\sigma_{ms}^2} \right\},$$

i. e., the components are multivariate Gaussian densities with diagonal covariance matrices. The maximum-likelihood estimates of the parameters $p(m)$, μ_{ms}, σ_{ms} can be computed by the expectation-maximization (EM) algorithm (Dempster et al. 1977; Grim and Haindl 2003). Anew we use a data set $\mathscr{S}$ obtained by pixel-wise shifting the observation window within the original texture image $\mathscr{S} = \{X_{\{r\}}^{(1)}, \ldots, X_{\{r\}}^{(K)}\}$, $X_{\{r\}}^{(k)} \subset X$. The corresponding log-likelihood function is maximized by the EM algorithm ($m \in \mathscr{M}, n \in \mathscr{N}, X_{\{r\}} \in \mathscr{S}$), and the iterations are (14), (15) and

$$\mu_{m,n}^{(t+1)} = \frac{1}{\sum_{X_{\{r\}} \in \mathscr{S}} q^{(t)}(m \mid X_{\{r\}})} \sum_{X_{\{r\}} \in \mathscr{S}} X_n \, q(m \mid X_{\{r\}}), \tag{18}$$

$$(\sigma_{m,n}^{(t+1)})^2 = -(\mu_{m,n}^{(t+1)})^2 + \frac{\sum_{X_{\{r\}} \in \mathscr{S}} X_n^2 \, q^{(t)}(m \mid X_{\{r\}})}{\sum_{X_{\{r\}} \in \mathscr{S}} q(m|X_{\{r\}})}. \tag{19}$$

Local Markov and Mixture Models

While the principal models control the overall large-scale low-frequency textural structure, the local models synthesize the detail, regional and fine-granularity spatial-spectral BTF information. Once we have synthesized the required size's principal random field, using some of the previously described models, we use it to synthesize the local random part (3) of the BTF compound random model Y. This local model is a mosaic of K random field sub-models. These sub-models are assumed to be of the same type, but they differ in parameters and contextual support sets. This assumption is for simplicity only and is not restrictive because every sub-model is estimated and synthesized independently; thus, the Y mosaic can be easily composed of different types of random field models.

Local i-th texture region (not necessarily continuous) models are view and illumination dependent; thus, they need to be ideally represented by models which can be analytically estimated as well as easily non-iteratively synthesized (BTF-CMRF^{NProl3DCAR} (Haindl and Havlíček 2010), BTF-CMRF^{2P3DCAR} (Haindl et al. 2012), BTF-CMRF^{PV3DCAR} (Haindl et al. 2015b), BTF-CMRF^{c3DGM} (Haindl and Havlíček 2016), BTF-CMRF^{BM3DCAR} (Haindl and Havlíček 2017b), BTF-CMRF^{GM3DCAR}, BTF-CMRF^{NProl3DMA} (Haindl and Havlíček 2017a), BTF-CMRF^{NPi3DCAR} (Haindl and Havlíček 2018b), BTF-CMRF^{NPfi3DCAR} (Haindl and Havlíček 2018a)).

3D Causal Simultaneous Autoregressive Model

The 3D causal simultaneous autoregressive model (3DCAR) is an exceptional model because all its statistics can be solved analytically, and it can be utilized to build much more complex nD data models. For example, the 7D BTF models illustrated in Fig. 4 are composed from up to one hundred 3DCARs.

A digitized image Y is assumed to be defined on a finite rectangular $N \times M \times d$ lattice I, and $r = (r_1, r_2, r_3) \in I$ denotes a pixel multiindex with the row, columns, and spectral indices, respectively. The notation $I_r^c \subset I$ is a causal or unilateral neighborhood of pixel r, i.e.,

$$I_r^c \subset I_r^C = \{s : 1 \leq s_1 \leq r_1, 1 \leq s_2 \leq r_2, s \neq r\}.$$

The 3D causal simultaneous autoregressive model (3DCAR) is the wide-sense Markov model that can be written in the following regression equation form:

$$\tilde{Y}_r = \sum_{s \in I_r^c} A_s \tilde{Y}_{r-s} + e_r \qquad \forall r \in I \tag{20}$$

where A_s are matrices (21) and the zero mean white Gaussian noise vector e_r has uncorrelated components with data indexed from I_r^c but noise vector components can be mutually correlated with a constant covariance matrix Σ.

$$A_{s_1,s_2} = \begin{pmatrix} a_{1,1}^{s_1,s_2}, \ldots, a_{1,d}^{s_1,s_2} \\ \vdots, \ddots, \vdots \\ a_{d,1}^{s_1,s_2}, \ldots, a_{d,d}^{s_1,s_2} \end{pmatrix} \tag{21}$$

where $d \times d$ are parameter matrices. The model can be expressed in the matrix form

$$Y_r = \gamma Z_r + e_r, \tag{22}$$

where

$$Z_r = [\tilde{Y}_{r-s}^T : \forall s \in I_r^c], \tag{23}$$

Z_r is a $d\eta \times 1$ vector, $\eta = card(I_r^c)$ and γ

$$\gamma = [A_1, \ldots, A_\eta] \tag{24}$$

is a $d \times d\eta$ parameter matrix. To simplify notation the multiindexes $r, s, \ldots$ have only two components further on in this section.

An optimal support can be selected as the most probable model given past data

$$Y^{(r-1)} = \{Y_{r-1}, Y_{r-2}, \ldots, Y_1, Z_r, Z_{r-1}, \ldots, Z_1\},$$

i.e., $\max_j \{p(\mathscr{M}_j \mid Y^{(r-1)})\}$. Simultaneous conditional density can be evaluated analytically from

$$p(Y^{(r-1)} \mid \mathscr{M}_j) = \int \int p(Y^{(r-1)} \mid \gamma, \Sigma^{-1}) p(\gamma, \Sigma^{-1} \mid \mathscr{M}_j) d\gamma d\Sigma^{-1} \tag{25}$$

, and for the implemented uniform priors start, we get a decision rule (Haindl and Šimberová 1992):

The most probable AR model given past data $Y^{(r-1)}$, the normal-Wishart parameter prior and the uniform model prior is the model $\mathscr{M}_i$ (Haindl 1983) for which

$$i = \arg\max_j \{D_j\}$$

$$D_j = -\frac{d}{2} \ln |V_{x(r-1)}| - \frac{\beta(r) - d\eta + d + 1}{2} \ln |\lambda_{(r-1)}| + \frac{d^2\eta}{2} \ln \pi \tag{26}$$
$$+ \sum_{i=1}^{d} \left[\ln \Gamma \left(\frac{\beta(r) - d\eta + d + 2 - i}{2} \right) - \ln \Gamma \left(\frac{\beta(0) - d\eta + d + 2 - i}{2} \right) \right]$$

where $V_{z(r-1)} = \tilde{V}_{z(r-1)} + V_{z(0)}$ with $\tilde{V}_{z(r-1)}$ defined in (31), $V_{z(0)}$ is an appropriate part of V_0 (31), $\beta(r)$ is defined in (27), (28) and $\lambda_{(r-1)}$ is (29).

The statistics (26) uses the following notation (27), (28), (29), (30) and (31):

$$\beta(r) = \beta(0) + r - 1 = \beta(r-1) + 1, \tag{27}$$

$$\beta(0) > \eta - 2, \tag{28}$$

and

$$\lambda_{(r)} = V_{y(r)} - V^T_{zy(r)} V^{-1}_{z(r)} V_{zy(r)}. \tag{29}$$

$$V_{r-1} = \tilde{V}_{r-1} + V_0, \tag{30}$$

$$\tilde{V}_{r-1} = \begin{pmatrix} \sum_{k=1}^{r-1} \tilde{Y}_k \tilde{Y}_k^T & \sum_{k=1}^{r-1} \tilde{Y}_k \tilde{Z}_u^T \\ \sum_{k=1}^{r-1} \tilde{Z}_k \tilde{Y}_k^T & \sum_{k=1}^{r-1} \tilde{Z}_k \tilde{Z}_k^T \end{pmatrix} = \begin{pmatrix} \tilde{V}_{y(r-1)} & \tilde{V}^T_{zy(r-1)} \\ \tilde{V}_{zy(r-1)} & \tilde{V}_{z(r-1)} \end{pmatrix}. \tag{31}$$

Marginal densities $p(\gamma \mid Y^{(r-1)})$ and $p(\Sigma^{-1} \mid Y^{(r-1)})$ can be evaluated from (32), (33), respectively.

$$p(\gamma \mid Y^{(r-1)}) = \int p(\gamma, \Sigma^{-1} \mid Y^{(r-1)}) d\Sigma^{-1} \tag{32}$$

$$p(\Sigma^{-1} \mid Y^{(r-1)}) = \int p(\gamma, \Sigma^{-1} \mid Y^{(r-1)}) d\gamma \tag{33}$$

The marginal density $p(\Sigma^{-1} \mid Y^{(r-1)})$ is the Wishart distribution density (Haindl 1983)

$$p(\Sigma^{-1} \mid Y^{(r-1)}) = \frac{\pi^{\frac{d(1-d)}{4}} \; |\Sigma^{-1}|^{\frac{\beta(r)-d\eta}{2}}}{2^{\frac{d(\beta(r)-d\eta+d+1)}{2}} \; \prod_{i=1}^{d} \Gamma(\frac{\beta(r)-d\eta+2+d-i}{2})} \; |\lambda_{(r-1)}|^{\frac{\beta(r)-d\eta+d+1}{2}}$$
$$\exp\left\{-\frac{1}{2}\, tr\{\Sigma^{-1}\lambda_{(r-1)}\}\right\} \tag{34}$$

with

$$E\left\{\Sigma^{-1} \mid Y^{(r-1)}\right\} = (\beta(r) - d\eta + d + 1)\,\lambda_{(r-1)}^{-1} \tag{35}$$

$$E\left\{(\Sigma^{-1} - E\{\Sigma^{-1} \mid Y^{(r-1)}\})^T(\Sigma^{-1} - E\{\Sigma^{-1} \mid Y^{(r-1)}\}) \mid Y^{(r-1)}\right\} = \frac{2(\beta(r) - d\eta + 1)}{\lambda_{(r-1)}\lambda_{(r-1)}^T}. \tag{36}$$

The marginal density $p(\gamma \mid Y^{(r-1)})$ is matrix t distribution density (Haindl 1983)

$$p(\gamma \mid Y^{(r-1)}) = \frac{\prod_{i=1}^{d} \Gamma(\frac{\beta(r)+d+2-i}{2})}{\prod_{i=1}^{d} \Gamma(\frac{\beta(r)-d\eta+d+2-i}{2})} \pi^{-\frac{d^2\eta}{2}} \, |\lambda_{(r-1)}|^{-\frac{d\eta}{2}} \, |V_{x(r-1)}|^{\frac{d}{2}} \left| I + \lambda_{(r-1)}^{-1}(\gamma - \hat{\gamma}_{r-1}) \, V_{z(r-1)}(\gamma - \hat{\gamma}_{r-1})^T \right|^{-\frac{\beta(r)+d+1}{2}} \tag{37}$$

with the mean value

$$E\left\{\gamma \mid Y^{(r-1)}\right\} = \hat{\gamma}_{r-1} \tag{38}$$

and covariance matrix

$$E\left\{(\gamma - \hat{\gamma}_{r-1})^T(\gamma - \hat{\gamma}_{r-1}) \mid Y^{(r-1)}\right\} = \frac{V_{z(r-1)}^{-1}\lambda_{(r-1)}}{\beta(r) - d\eta}. \tag{39}$$

Similar statistics can be easily derived (Haindl 1983) for the alternative Jeffreys non-informative parameter prior. Similar to other model statistics, also the predictive density can be analytically derived.

The one-step-ahead predictive posterior density for the normal-Wishart parameter prior has the form of d-dimensional Student's probability density (40) (Haindl 1983)

$$p(Y_r \mid Y^{(r-1)}) = \frac{\Gamma(\frac{\beta(r)-d\eta+d+2}{2})}{\Gamma(\frac{\beta(r)-d\eta+2}{2}) \, \pi^{\frac{d}{2}} \, (1 + Z_r^T V_{z(r-1)}^{-1} Z_r)^{\frac{d}{2}} \, |\lambda_{(r-1)}|^{\frac{1}{2}}} \left(1 + \frac{(Y_r - \hat{\gamma}_{r-1}Z_r)^T \lambda_{(r-1)}^{-1}(Y_r - \hat{\gamma}_{r-1}Z_r)}{1 + Z_r^T V_{z(r-1)}^{-1} Z_r}\right)^{-\frac{\beta(r)-d\eta+d+2}{2}}, \tag{40}$$

with $\beta(r) - d\eta + 2$ degrees of freedom; if $\beta(r) > d\eta$ then the conditional mean value is

$$E\left\{Y_r \mid Y^{(r-1)}\right\} = \hat{\gamma}_{r-1} Z_r , \tag{41}$$

and

$$E\left\{(Y_r - \hat{\gamma}_{r-1} Z_r)(Y_r - \hat{\gamma}_{r-1} Z_r)^T \mid Y^{(r-1)}\right\} = \frac{1 + Z_r V_{z(r-1)}^{-1} Z_r^T}{(\beta(r) - d\eta)} \lambda_{(r-1)} . \tag{42}$$

The 3DCAR model can be made adaptive if we modify its recursive statistics using an exponential forgetting factor, i.e., a constant $\varphi \approx 0.99$. This forgetting factor smaller than 1 is used to weigh the influence of older data. The numerical stability of 3DCAR can be guaranteed if all its recursive statistics use the square root factor updating applying either the Cholesky or LDL^T decomposition (Haindl 2000), respectively.

The 3DCAR (analogously also the 2DCAR model) model has advantages in analytical solutions (Bayes, ML, or LS estimates) for I_r, $\hat{\gamma}$, $\hat{\sigma}^2$, $\hat{Y}_r$ statistics. It allows straightforward, fast synthesis, adaptivity, and building efficient recursive application algorithms.

3D Moving Average Model

Single multispectral texture factors Y are modeled using the extended version ($3D\,MA$) of the moving average model (Li et al. 1992; Haindl and Havlíček 2017a). A stochastic multispectral texture can be considered to be a sample from a 3D random field defined on an infinite 2D lattice. The model assumes that each factor is the output of an underlying system, which completely characterizes it in response to a 3D uncorrelated random input. This system can be represented by the impulse response of a linear 3D filter. The intensity values of the most significant pixels, together with their neighbors, are collected and averaged. The resultant 3D kernel is used as an estimate of the impulse response of the underlying system. A synthetic mono-spectral factor can be generated by convolving an uncorrelated 3D random field with this estimate. Suppose a stochastic multispectral texture denoted by Y is the response of an underlying linear system that completely characterizes the texture in response to a 3D uncorrelated random input E_r; then, Y_r is determined by the difference equation

$$Y_r = \sum_{s \in I_r} B_s E_{r-s} \tag{43}$$

where B_s are constant matrix coefficients and $I_r \subset I$.

Hence, Y_r can be represented as $Y_r = h(r) * E_r$ where the convolution filter $h(r)$ contains all parameters B_s. In this equation, the underlying system behaves as a 3D filter, where we restrict the system impulse response to have significant values only within a finite region. The geometry of I_r determines the causality or non-causality of the model.

The parameter estimation can be based on the modified random decrement technique (RDT) (Cole Jr 1973; Asmussen 1997). RDT assumes that the input is an uncorrelated random field. If every pixel component is higher than its corresponding threshold vector component and simultaneously at least one of its four neighbors is less than this threshold, the pixel is saved in the data accumulator. The procedure begins by selecting thresholds usually chosen as some percentage of the standard deviation of each spectral plane's intensities separately. In addition to that, a 3D MA model also requires to estimate the noise spectral correlation, i.e.,

$$\begin{aligned} E\{E_r E_s\} &= 0 \qquad && \forall r_1 \neq s_1 \vee r_2 \neq s_2, \\ E\{E_{r_1,r_2,r_3} E_{r_1,r_2,\bar{r}_3}\} &\neq 0 \qquad && \forall r_3 \neq \bar{r}_3. \end{aligned}$$

The synthetic factor can be generated simply by convolving an uncorrelated 3D RF E with the estimate of B according to (43). All generated factors form a new Gaussian pyramid. Fine resolution synthetic smooth texture is obtained by the collapse of the pyramid, i.e., an inverse procedure of that one creating the pyramid. This model can be used for materials which consist of several types of relatively small regions with fine-granular inner structure such as sand, grit, cork, lichen, or plaster. Figure 12 illustrates the visual quality of this simple model if the regional textures violate this fine-granularity assumption.

Spatial 3D Gaussian Mixture Model

A static homogeneous three-dimensional textural factor Y is assumed to be defined on a finite rectangular $M \times N \times d$ lattice I, $r = (r_1, r_2) \in I$ denotes a pixel multiindex with the row, columns, and indices, respectively. Let us suppose that Y represents a realization of a random vector with a probability distribution $P(Y)$. The statistical properties of interior pixels of the moving window on Y are translation invariant due to assumed textural homogeneity. They can be represented by a joint probability distribution, and the properties of the texture can be fully characterized

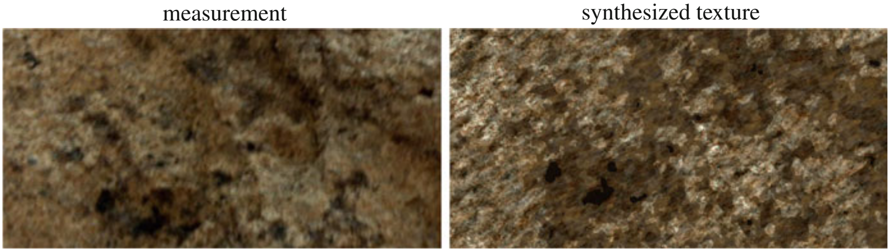

Fig. 12 The stone measurement and its synthesis (BTF-CMRF^{NP3DMA})

by statistical dependencies on a sub-field, i.e., by a marginal probability distribution of spectral levels on pixels within the scope of a window centered around the location r and specified by the index set:

$$I_r = \{r + s : |r_1 - s_1| \leq \alpha \wedge |r_2 - s_2| \leq \beta\} \subset I. \tag{44}$$

The index set I_r depends on modeled visual data and can have any other than this rectangular shape. $Y_{\{r\}}$ denotes the corresponding matrix containing all $d \times 1$ vectors Y_s in some fixed order arrangement such that $s \in I_r$, $Y_{\{r\}} = [Y_s \ \forall s \in I_r]$, $Y_{\{r\}} \subset Y$, $\eta =$ cardinality$\{I_r\}$, and $P(Y_{\{r\}})$ is the corresponding marginal distribution of $P(Y)$.

If we assume the joint probability distribution $P(Y_{\{r\}})$, in the form of a normal mixture (Haindl and Havlíček 2016)

$$\begin{aligned} P(Y_{\{r\}}) &= \sum_{m \in \mathcal{M}} p(m)\, P(Y_{\{r\}} \,|\, \mu_m, \Sigma_m) \qquad Y_{\{r\}} \subset Y, \\ &= \sum_{m \in \mathcal{M}} p(m) \prod_{s \in I_r} p_s(Y_s \,|\, \mu_{m,s}, \Sigma_{m,s}) \end{aligned} \tag{45}$$

where $Y_{\{r\}} \in \Re^{d \times \eta}$ is $d \times \eta$ matrix, μ_m is $d \times \eta$ mean matrix, Σ_m is $d \times d \times \eta$ a covariance tensor, and $p(m)$ are probability weights and the mixture components are defined as products of multivariate Gaussian densities

$$P(Y_{\{r\}} \,|\, \mu_m, \Sigma_m) = \prod_{s \in I_{\{r\}}} p_s(Y_s \,|\, \mu_{ms}, \Sigma_{ms}), \tag{46}$$

$$p_s(Y_s \,|\, \mu_{ms}, \Sigma_{ms}) = \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_{m,s}|^{\frac{1}{2}}} \exp\left\{ -\frac{1}{2} (Y_r - \mu_{m,s})^T \Sigma_{m,s}^{-1} (Y_r - \mu_{m,s}) \right\}, \tag{47}$$

i. e., the components are multivariate Gaussian densities with covariance matrices (53).

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

$$\mathscr{S} = \{Y_{\{r\}} \,\forall r \in I,\ I_r \subset I\} \qquad Y_{\{r\}} \in \Re^{d \times \eta}. \tag{48}$$

Parameter Estimation

The unknown parameters of the approximating mixture can be estimated using the iterative EM algorithm (Dempster et al. 1977). In order to estimate the unknown distributions $p_s(\cdot \,|\, m)$ and the component weights $p(m)$ we maximize the likelihood function (49) corresponding to the training set (48):

$$L = \frac{1}{|\mathscr{S}|} \sum_{Y_{\{r\}} \in \mathscr{S}} \log \left[\sum_{m \in \mathscr{M}} P(Y_{\{r\}} \mid \mu_m, \Sigma_m)\, p(m) \right]. \tag{49}$$

The likelihood is maximized using the iterative EM algorithm (with non-diagonal covariance matrices):

E:

$$q^{(t)}(m \mid Y_{\{r\}}) = \frac{\tilde{P}^{(t)}(Y_{\{r\}} \mid \mu_m, \Sigma_m)\, p^{(t)}(m)}{\sum_{j \in \mathscr{M}} P^{(t)}(Y_{\{r\}} \mid \mu_j, \Sigma_j)\, p^{(t)}(j)}, \tag{50}$$

M:

$$p^{(t+1)}(m) = \frac{1}{|\mathscr{S}|} \sum_{Y_{\{r\}} \in \mathscr{S}} q^{(t)}(m \mid Y_{\{r\}}), \tag{51}$$

$$\mu_{m,s}^{(t+1)} = \frac{1}{\sum_{Y_{\{r\}} \in \mathscr{S}} q^{(t)}(m \mid Y_{\{r\}})} \sum_{Y_{\{r\}} \in \mathscr{S}} Y_s q^{(t)}(m \mid Y_{\{r\}}). \tag{52}$$

The covariance matrices are

$$\begin{aligned} \Sigma_{m,s}^{(t+1)} &= \frac{\sum_{Y_{\{r\}} \in \mathscr{S}, Y_s \in Y_{\{r\}}} q^{(t)}(m \mid Y_{\{r\}})}{\sum_{Y_r \in \mathscr{S}} q^{(t)}(m \mid Y_{\{r\}})} (Y_s - \mu_{m,s}^{(t+1)})(Y_s - \mu_{m,s}^{(t+1)})^T \\ &= \frac{\sum_{Y_{\{r\}} \in \mathscr{S}, Y_s \in Y_{\{r\}}} q^{(t)}(m \mid Y_{\{r\}})\, Y_s Y_s^T}{\sum_{Y_r \in \mathscr{S}} q^{(t)}(m \mid Y_{\{r\}})} - \frac{p^{(t+1)}(m)\, |\mathscr{S}|\, \mu_{m,s}^{(t+1)} \left(\mu_{m,s}^{(t+1)}\right)^T}{\sum_{Y_r \in \mathscr{S}} q^{(t)}(m \mid Y_{\{r\}})}. \end{aligned} \tag{53}$$

The iteration process stops when the criterion increments are sufficiently small. The EM algorithm iteration scheme has the monotonic property $L^{(t+1)} \geq L^{(t)},\ t = 0, 1, 2, \ldots$ which implies the convergence of the sequence $\{L^{(t)}\}_0^\infty$ to a stationary point of the EM algorithm (local maximum or a saddle point of L). Figure 13 illustrates the usefulness of the BTF-CMRF^{3DGM} model for textile material modeling, while Fig. 18 shows this model applied to scratch restoration.

Applications

Numerous modeling applications can exploit the BTF models. The synthesis is beneficial not only for physically correct appearance modeling of surface materials under realistic and variable observation conditions (Figs. 15 and 17, upper row)

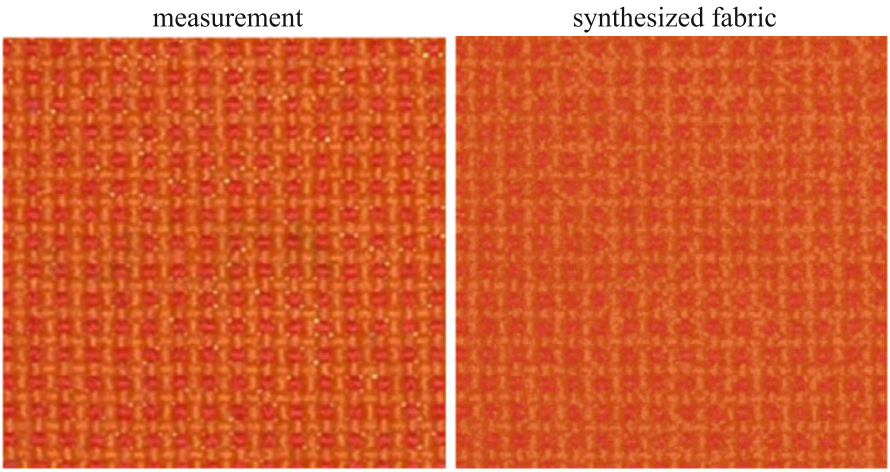

Fig. 13 The fabric measurement and its synthesis (BTF-CMRF^{3DGM})

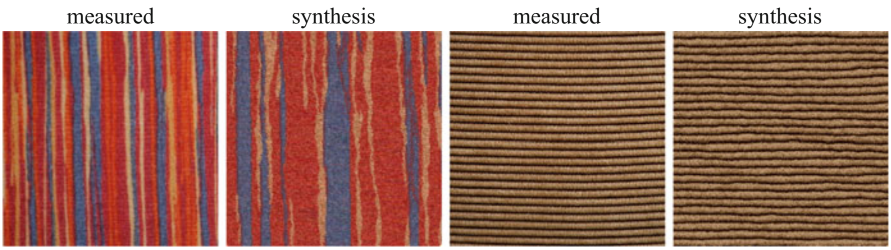

Fig. 14 Measured original cloth and corduroy materials and their synthesis using the $CRF^{BM-3CAR}$ model

but also for texture editing (Fig. 16), texture compression, or texture inpainting and restoration (Fig. 18). Various state-of-the-art unsupervised, semi-supervised, or supervised visual scene classification and understanding under variable observation conditions is the primary application for BTF analysis.

Texture Synthesis and Enlargement

Texture synthesis methods may be divided primarily into intelligent sampling and model-based methods (Fig. 14). They differ in need to store (sampling) or not (modeling) some actual texture measurements for new texture synthesis. Thus, even some methods which view texture as a stochastic process (Heeger and Bergen 1995; Efros and Leung 1999) still require to store an input exemplar. Sampling approaches De Bonet (1997), Efros and Leung (1999), Efros and Freeman (2001), Heeger and Bergen (1995), Xu et al. (2000), Dong and Chantler (2002), and Zelinka and Garland

(2002) rely on sophisticated sampling from real texture measurements, while the model-based techniques (Kashyap 1981; Haindl 1991; Haindl and Havlíček 1998, 2000; Bennett and Khotanzad 1998, 1999; Gimelfarb 1999; Paget and Longstaff 1998; Zhu et al. 2000) describe texture data using multidimensional mathematical models, and their synthesis is based on the estimated model parameters only. The mathematical model-based synthesis has an advantage in the possibility of seamless texture enlargement to any size (e.g., Fig. 6). The enlargement of a restricted texture measurement is always required in any application but cannot be achieved with sampling approaches without visible seams or repetitions.

The BTF modeling's ultimate aim is to create a visual impression of the same material without a pixel-wise correspondence to the finding condition model of the original measurements. Figure 15 shows the finding condition model of the beautiful gothic style relief (around 1370) of the Christ in Gethsemane (Prague) in the right and restored condition to a possible original appearance in the left.

The cornerstone of our BTF compression and modeling methods is the replacement of a vast number of original BTF measurements by their efficient parametric estimates derived from an underlying set of spatial probabilistic models and thus to allow a huge BTF compression ratio unattainable by any alternative sampling-based BTF synthesis method. Simultaneously these models can be used to reconstruct missing parts of the BTF measurement space or the controlled BTF space editing (Haindl and Havlíček 2009, 2012; Haindl et al. 2015b) by changing some of the model's parameters.

Textures without significant low frequencies such as Fig. 14-corduroy or Fig. 13-fabric can be modeled using simple local models only, either Markovian or mixtures such as 3DCAR, 3DMA, 3DBM, 3DGM, etc. Textures with substantial low frequencies (Figs. 4, 9, 14-cloth) will benefit from a compound version of the BTF model. Non-BTF textures can approximate low frequencies using a multiscale version of these models, e.g., pyramidal model (Haindl and Filip 2013).

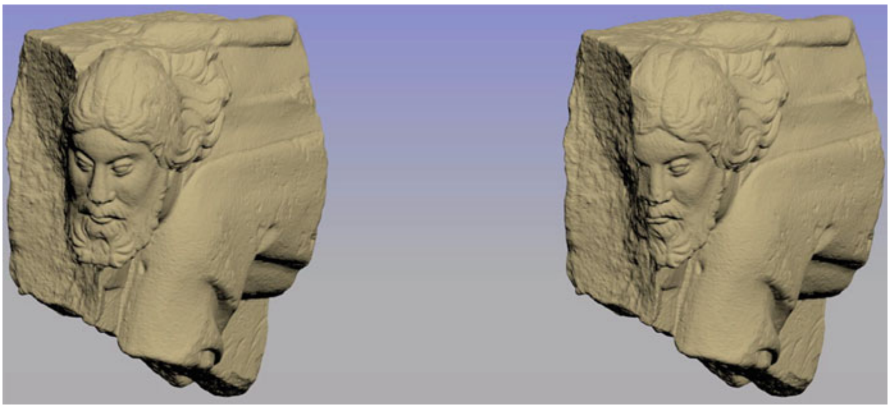

Fig. 15 3D model of the beautiful gothic style relief of the Christ in Gethsemane, Prague (finding condition model right, restored condition to a possible original appearance left) mapped with the BTF synthetic plaener using the $CMRF^{3CAR}$ model

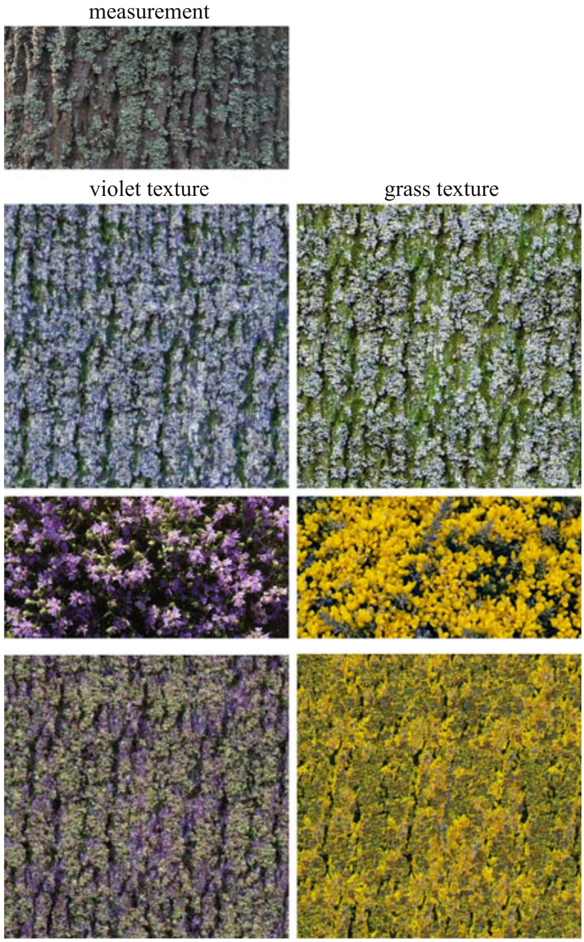

Fig. 16 Synthetic BTF-CMRF^{NProl3DCAR} edited and enlarged maple bark texture (second and fourth rows) with single sub-models estimated from their natural measurements (maple bark first and flowers third row)

The 3DCAR model is synthesized directly from its predictor (41) and Gaussian noise generator (22), (39). The advantage of a mixture model is its simple synthesis based on the marginals:

$$p_{n\,|\,\rho}(Y_n \,|\, Y_{\{\rho\}}) = \sum_{m=1}^{\dot{M}} W_m(Y_{\{\rho\}})\; p_n(Y_n \,|\, m), \tag{54}$$

where $W_m(Y_{\{\{\rho\}})$ are the a posterior component weights corresponding to the given sub-matrix $Y_{\{\rho\}} \subset Y_{\{r\}}$:

$$W_m(Y_{\{\rho\}}) = \frac{p(m)\,P_\rho(Y_{\{\rho\}} \,|\, m)}{\sum_{j=1}^{\dot{M}} p(j)\,P_\rho(Y_{\{\rho\}} \,|\, j)}, \tag{55}$$

$$P_\rho(Y_{\{\rho\}}\,|\,m) = \prod_{n\in\rho} p_n(Y_n\,|\,m). \tag{56}$$

There are several alternatives for the 3DGM model synthesis (Haindl et al. 2011) (Fig. 13). The unknown multivariate vector-levels Y_n can be synthesized by random sampling from the conditional density (54), or the mixture RF can be approximated using the GM mixture prediction.

Texture Compression

BTF – the best current measurable representation of a material appearance – requires tens of thousands of images using a sophisticated high-precision automatic measuring device. Such measurements result in a massive amount of data that can easily reach tens of terabytes for a single measured material. Nevertheless, these data have still insufficient spatial extent for any real virtual reality applications and have to be further enlarged using advanced modeling techniques. The resulting BTF size excludes its direct rendering in graphical applications, and compression of these huge BTF data spaces is inevitable. The usual car interior model requires more than 20 of such demanding BTF material measurements, and a similar problem holds for other applications of the physically correct appearance modeling such as computer games or film animations. A related problem is measurement data storage because storage technology is still the weak link, lagging behind recent developments in data sensing technologies. The apparent solution is mathematical modeling which allows replacing massive measured data with few thousand parameters and thus to reach tremendous unbeatable appearance data compression apart from unlimited seamless material texture enlargement. For example, the compression ratio relative to our BTF measurements is up to $1 : 1000000$.

Texture Editing

Material-appearance editing is a practical approach with vast potential for significant speedup and cost reduction in industrial virtual prototyping or various design applications. An editing process can simulate materials for which no direct measurements are available or not existing in Nature (Fig. 16). Another example of the edited texture is two panels with the artificial but fitting glass mosaic synthesis in St. Vitus Cathedral in Prague Castle stained glass window on Fig. 11. Such edited artifacts allow an artist to test several possible design alternatives or model defunct monuments.

Illumination Invariants

Textures are essential clues to specify objects present in a visual scene. However, the appearance of natural textures is highly illumination and view angle-dependent. As a consequence, the most recent realistic texture-based classification or segmentation methods require multiple training images (Varma and Zisserman 2005) captured under all possible illumination and viewing conditions for each class. Such learning is clumsy, probably expensive, and very often even impossible if required measurements are not available.

If we assume fixed positions of viewpoint and illumination sources, uniform illumination sources, and Lambertian surface reflectance, then two images $\tilde{Y}, Y$ acquired with different illumination spectra can be linearly transformed to each other:

$$\tilde{Y}_r = B\, Y_r \qquad \forall r. \tag{57}$$

It is possible to show (Vacha and Haindl 2007) that assuming (57) the following 3DCAR model-derived features are illumination invariant:

1. trace: $trace\, A_m,\ m = 1, \ldots, \eta\, K$
2. eigenvalues: $\nu_{m,j}$ of $A_m,\ m = 1, \ldots, \eta\, K,\ j = 1, \ldots, C$
3. $1 + X_r^T V_x^{-1} X_r$,
4. $\sqrt{\sum_r \left(Y_r - \hat{\gamma} X_r\right)^T \lambda^{-1} \left(Y_r - \hat{\gamma} X_r\right)}$,
5. $\sqrt{\sum_r \left(Y_r - \mu\right)^T \lambda^{-1} \left(Y_r - \mu\right)}$,

where μ is the mean value of the vector Y_r.

Above textural features derived from the 3DCAR model are robust to illumination direction changes, invariant to illumination brightness and spectrum changes, and simultaneously also robust to Gaussian noise degradation. We extensively verified this property on the BTF texture measurements, where illumination sources are spanned over 75% of possible illumination half-sphere. Figure 17 illustrates the application of 3DCAR model-derived features are illumination invariants to the unsupervised wood mosaic segmentation.

(Un)supervised Image Recognition

Unsupervised or supervised texture segmentation is the prerequisite for successful content-based image retrieval, scene analysis, automatic acquisition of virtual models, quality control, security, medical applications, and many others.

Similarly, robust surface material recognition requires the BTF data learning set. We classified 65 wood species measured in the BTF representation in the study Mikeš and Haindl (2019) using the state-of-the-art convolutional neural network (TensorFlow library (Google 2019; Krizhevsky 2009; Krizhevsky et al. 2012; Pattanayak 2017)). We documented (Mikeš and Haindl 2019) sharp classification accuracy decrease when using standard texture recognition approach, i.e., small

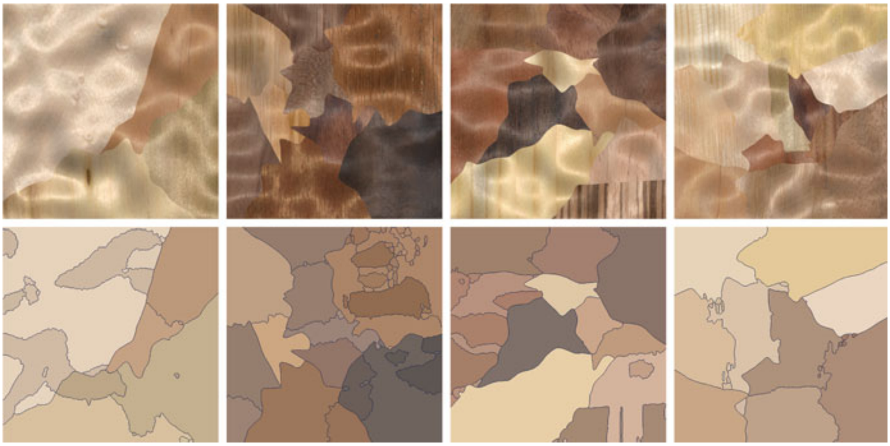

Fig. 17 BTF wood mosaic and the MW3-AR8^{i} model-based (Haindl et al. 2015a) unsupervised segmentation results

learning set size and the vertical viewing and illumination angle, which is a very inadequate representation of the enormous material appearance variability.

Although plentiful different methods were already published (Zhang 1997), the image recognition problem is still far from being solved. This situation is among others due to missing reliable performance comparison between different techniques. Only limited results were published (Martin et al. 2001; Sharma and Singh 2001; Ojala et al. 2002; Haindl and Mikeš 2008) on suitable quantitative measures that allow us to evaluate and compare the quality of segmentation algorithms.

Spatial interaction models and especially Markov random field-based models are increasingly popular for texture representation (Kashyap 1986; Reed and du Buf 1993; Haindl 1991), etc. Several researchers dealt with the difficult problem of unsupervised segmentation using these models, see for example Panjwani and Healey (1995), Manjunath and Chellapa (1991), Andrey and Tarroux (1998), Haindl (1999), and Matuszak and Schreiber (2009).

Our unsupervised segmenters (Haindl and Mikeš 2004, 2005, 2006; Haindl et al. 2015a) assume the multispectral or multi-channel textures to be locally represented by the parameters (Θ_r) of the multidimensional random field models possibly recursively evaluated for each pixel and several scales. The segmentation part of the algorithm is then based on the underlying Gaussian mixture model ($p(\Theta_r) = \sum_{i=1}^{K} p_i \, p(\Theta_r \mid \nu_i, \Sigma_i)$) representing the Markovian parametric space and starts with an over-segmented initial estimation, which is adaptively modified until the optimal number of homogeneous mammogram segments is reached. The corresponding mixture model equations ($p(\Theta_r)$, $p(\Theta_r \mid \nu_i, \Sigma_i)$) are solved using a modified EM algorithm (Haindl and Mikeš 2007).

The concept of decision fusion for high-performance pattern recognition is well known and widely accepted in the area of supervised classification, where (often very diverse) classification technologies, each providing complementary sources of information about class membership, can be integrated to provide more accurate, robust, and reliable classification decisions than single-classifier applications. Our method (Haindl and Mikeš 2007) circumvents the problem of multiple unsupervised segmenter combination by fusing multiple-processed measurements into a single segmenter feature vector.

Multispectral/Multi-channel Image Restoration

Physical imaging, processing or transmission systems, and a recording medium are imperfect, and thus a recorded image represents a degraded version of the original scene.

The image restoration task is to recover an unobservable image given the observed corrupted image $\ddot{Y}$ with respect to some statistical criterion. Image restoration is a busy research area for already several decades, and many restoration algorithms have been proposed (Andrews and Hunt 1977; Geman and Geman 1984; Acton and Bovik 1999; Loubes and Rochet 2009; Felsberg 2009; Burgeth et al. 2009; Polzehl and Tabelow 2009).

The image degradation is often supposed to be approximated by the linear degradation model:

$$\ddot{Y}_r = \sum_{s \in I_r} f_s \, Y_{r-s} + e_r \tag{58}$$

where f is a discrete representation of the unknown point-spread function. The point-spread function can be non-homogeneous, but we assume its slow changes relative to the size of an image. I_r is some contextual support set, and the degradation noise e is uncorrelated with the unobservable image. The point-spread function is unknown but such that we can assume the unobservable image Y to be reasonably well approximated by the expectation of the corrupted image

$$\hat{Y} = E\{\ddot{Y}\} \tag{59}$$

in regions with gradual pixel value changes.

Let us approximate after having observed $\ddot{Y}^{(j-1)} = \{\ddot{Y}_{j-1}, \dots, \ddot{Y}_1\}$ the mean value $\hat{Y}_j = E\{\ddot{Y}_j\}$ by the $E\{\ddot{Y}_j \mid \ddot{Y}^{(j-1)} = \ddot{y}^{(j-1)})$ where $\ddot{y}^{(j-1)}$ are known past realization for j. Thus, we suppose that all other possible realizations $\ddot{y}^{(j-1)}$ than the true past pixel values have negligible probabilities. This assumption implies conditional expectations approximately equal to unconditional ones, i.e.,

$$E\{\ddot{Y}_j\} \approx E\{\ddot{Y}_j \mid \ddot{Y}^{(j-1)}\}, \tag{60}$$

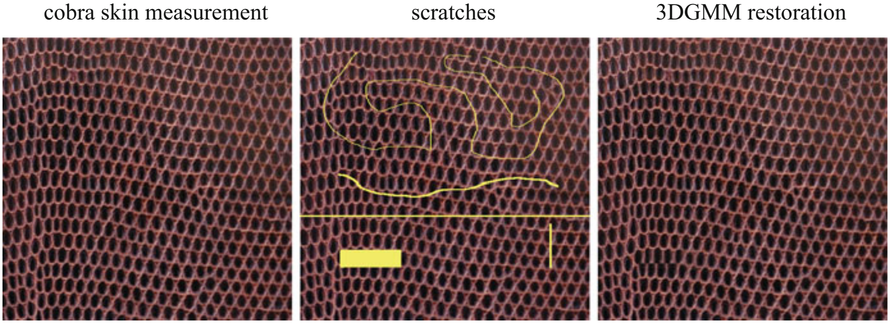

Fig. 18 Cobra skin scratch restoration using the spatial 3D Gaussian mixture model

and assuming the noisy image $\ddot{Y}$ can be represented by a 3DCAR model, then the restoration model as well as the local estimation of the point-spread function leads to a fast analytical solution (Haindl 2002). A similar restoration approach can also be derived for a multi-channel (Haindl and Šimberová 2002) or multitemporal (Haindl and Šimberová 2005) image restoration problems typically caused by random fluctuations originating mostly in the Earth's atmosphere during ground-based telescope observations.

A difficult restoration problem is to restore missing parts of an image or a spatially correlated data field. For example, every movie deteriorates with usage and time irrespective of any care it gets. Movies (on both optical and magnetic materials) suffer from blotches, dirt, sparkles, noise, scratches (Fig. 18), missing or heavily corrupted frames, mold, flickering, jittering, image vibrations, and other problems. For each kind of defect, usually a different kind of restoration algorithm is needed. The scratch notion means every coherent region with missing data (simultaneously in all spectral bands) in a color movie frame (Haindl and Filip 2002), static image, range map, radio-spectrograph (Haindl and Šimberová 1996), radar observation, color textures (Haindl and Havlíček 2015), etc. These missing data restoration methods (inpainting) exploit correlations in the spatial/spectral/temporal data space and benefit from the discussed Markovian or mixture (Fig. 18) random field models.

Conclusion

There is no single universal BTF model applicable for physically correct modeling of visual properties of all possible BTF textures. Every presented model is better suited for some subspace of possible BTF textures, either natural or artificial. Their selection depends primarily on their spectral and spatial frequency content as well as on available learning data. We present exceptional adaptive 3D Markovian or mixture models, either solved analytically or iteratively and quickly synthesized.

The presented compound Markovian models are rare exceptions in the Markovian model family that allow deriving extraordinarily efficient and fast data processing algorithms. All their statistics can be either evaluated recursively, and they either do not need any Monte Carlo sampling typical for other Markovian models or can use a fast form of such sampling (Potts random field). The 3DCAR models have an advantage over non-causal (3DAR) in their analytical treatment. It is possible to find the analytical solution of model parameters, optimal model support, model predictor, etc. Similarly, the 3DCAR model synthesis is straightforward, and this model can be directly generated from the model equation.

The mixture models are capable of reducing additive noise and restore missing textural parts simultaneously. They produce high-quality results, especially of regular or near-regular color textures. Their typical drawback the extensive learning date set requirement is lessened by the ample available BTF measurement space using a transfer learning approach.

The BTF-CMRF models offer a large data compression ratio (only tens of parameters per BTF), easy simulation, and fast, seamless synthesis of any required texture size. The methods have no restriction to the number of spectral channels; thus, they can be easily applied to hyperspectral BTFs. The methods can be easily generalized for color or BTF texture editing by estimating some local models from different target materials or for image restoration or inpainting.

The Markovian models can be used for image enhancement, e.g., utterly automatic mammogram enhancement, multispectral and multiresolution texture qualitative measures development, or image or video segmentation. Some of these models also allow robust textural features for texture classification when learning and classified textures differ in scale. The classifiers based on Markovian features can exploit illumination or geometric invariance properties and often outperform the state-of-the-art alternative methods on tested public databases (e.g., eye, bark, needles, textures).

Acknowledgments The Czech Science Foundation Project GAČR 19-12340S supported this research.

References

Acton, S., Bovik, A.: Piecewise and local image models for regularized image restoration using cross-validation. IEEE Trans. Image Process. **8**(5), 652–665 (1999)

Aittala, M., Weyrich, T., Lehtinen, J.: Practical SVBRDF capture in the frequency domain. ACM Trans. Graph. (Proc. SIGGRAPH) **32**(4), 110:1– 110:13 (2013)

Aittala, M., Weyrich, T., Lehtinen, J.: Two-shot SVBRDF capture for stationary materials. ACM Trans. Graph. **34**(4), 110:1–110:13 (2015). https://doi.org/10.1145/2766967

Aittala, M., Aila, T., Lehtinen, J.: Reflectance modeling by neural texture synthesis. ACM Trans. Graph. **35**(4), 65:1–65:13 (2016). https://doi.org/10.1145/2897824.2925917

Anderson, T.W.: The Statistical Analysis of Time Series. Wiley, New York (1971)

Andrews, H.C., Hunt, B.: Digital Image Restoration. Prentice-Hall, Englewood Cliffs (1977)

Andrey, P., Tarroux, P.: Unsupervised segmentation of markov random field modeled textured images using selectionist relaxation. IEEE Trans. Pattern Anal. Mach. Intell. **20**(3), 252–262 (1998)

Asmussen, J.C.: Modal analysis based on the random decrement technique: application to civil engineering structures. PhD thesis, University of Aalborg (1997)

Aurenhammer, F.: Voronoi diagrams-a survey of a fundamental geometric data structure. ACM Comput. Surv. (CSUR) **23**(3), 345–405 (1991)

Baril, J., Boubekeur, T., Gioia, P., Schlick, C.: Polynomial wavelet trees for bidirectional texture functions. In: SIGGRAPH'08: ACM SIGGRAPH 2008 talks, p. 1. ACM, New York (2008). https://doi.org/10.1145/1401032.1401072

Bennett, J., Khotanzad, A.: Multispectral random field models for synthesis and analysis of color images. IEEE Trans. Pattern Anal. Mach. Intell. **20**(3), 327–332 (1998)

Bennett, J., Khotanzad, A.: Maximum likelihood estimation methods for multispectral random field image models. IEEE Trans. Pattern Anal. Mach. Intell. **21**(6), 537–543 (1999)

Broemeling, L.D.: Bayesian Analysis of Linear Models. Marcel Dekker, New York (1985)

Burgeth, B., Pizarro, L., Didas, S., Weickert, J.: Coherence-enhancing diffusion filtering for matrix fields. In: Locally Adaptive Filtering in Signal and Image Processing. Springer, Berlin (2009)

Cole Jr, H.A.: On-line failure detection and damping measurement of aerospace structures by random decrement signatures. Technical Report TMX-62.041, NASA (1973)

Dana, K.J., Nayar, S.K., van Ginneken, B., Koenderink, J.J.: Reflectance and texture of real-world surfaces. In: CVPR, pp. 151–157. IEEE Computer Society (1997)

De Bonet, J.: Multiresolution sampling procedure for analysis and synthesis of textured images. In: ACM SIGGRAPH 97, pp. 361–368. ACM Press (1997)

Debevec, P., Hawkins, T., Tchou, C., Duiker, H.P., Sarokin, W., Sagar, M.: Acquiring the reflectance field of a human face. In: Proceedings of ACM SIGGRAPH 2000, Computer Graphics Proceedings, Annual Conference Series, pp. 145–156 (2000)

Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the em algorithm. J. R. Stat. Soc. B **39**(1), 1–38 (1977)

Deschaintre, V., Aittala, M., Durand, F., Drettakis, G., Bousseau, A.: Single-image svbrdf capture with a rendering-aware deep network. ACM Trans. Graph. **37**(4), 1–15 (2018). https://doi.org/10.1145/3197517.3201378

Dong, J., Chantler, M.: Capture and synthesis of 3D surface texture. In: Texture 2002, vol. 1, pp. 41–45. Heriot-Watt University (2002)

Dong, J., Wang, R., Dong, X.: Texture synthesis based on multiple seed-blocks and support vector machines. In: 2010 3rd International Congress on Image and Signal Processing (CISP), vol. 6, pp. 2861–2864 (2010). https://doi.org/10.1109/CISP.2010.5646815

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 (2001). citeseer.nj.nec.com/efros01image.html

Efros, A.A., Leung, T.K.: Texture synthesis by non-parametric sampling. In: Proceedings of International Conference on Computer Vision (2), Corfu, pp. 1033–1038 (1999). citeseer.nj.nec.com/efros99texture.html

Felsberg, M.: Adaptive filtering using channel representations. In: Locally Adaptive Filtering in Signal and Image Processing. Springer, Berlin (2009)

Filip, J., Haindl, M.: Bidirectional texture function modeling: a state of the art survey. IEEE Trans. Pattern Anal. Mach. Intell. **31**(11), 1921–1940 (2009). https://doi.org/10.1109/TPAMI.2008.246

Geman, S., Geman, D.: Stochastic relaxation, gibbs distributions and bayesian restoration of images. IEEE Trans. Pattern Anal. Mach. Intel. **6**(11), 721–741 (1984)

Gimelfarb, G.: Image Textures and Gibbs Random Fields. Kluwer Academic Publishers, Dordrecht (1999)

Google (2019) Tensorflow. Technical report, Google AI, http://www.tensorflow.org/

Grim, J., Haindl, M.: Texture modelling by discrete distribution mixtures. Comput. Stat. Data Anal. **41**(3–4), 603–615 (2003)

Haindl, M.: Identification of the stochastic differential equation of the type arma. PhD thesis, ÚTIA Czechoslovak Academy of Sciences, Prague (1983)

Haindl, M.: Texture synthesis. CWI Q. **4**(4), 305–331 (1991)

Haindl, M.: Texture segmentation using recursive Markov random field parameter estimation. In: Bjarne, K.E., Peter, J. (eds.) Proceedings of the 11th Scandinavian Conference on Image Analysis, Pattern Recognition Society of Denmark, Lyngby, pp. 771–776 (1999). http://citeseer.ist.psu.edu/305262.html; http://www.ee.surrey.ac.uk/Research/VSSP/3DVision/virtuous/Publications/Haindl-SCIA99.ps.gz

Haindl, M.: Recursive square-root filters. In: Sanfeliu, A., Villanueva, J., Vanrell, M., Alquezar, R., Jain, A., Kittler, J. (eds.) Proceedings of the 15th IAPR International Conference on Pattern Recognition, vol. II, pp. 1018–1021. IEEE Press, Los Alamitos (2000). https://doi.org/10.1109/ICPR.2000.906246

Haindl, M.: Recursive model-based colour image restoration. Lect. Notes Comput. Sci. (2396), 617–626 (2002)

Haindl, M., Filip, J.: Fast restoration of colour movie scratches. In: Kasturi, R., Laurendeau, D., Suen, C. (eds.) Proceedings of the 16th International Conference on Pattern Recognition, vol. III, pp. 269–272. IEEE Computer Society, Los Alamitos (2002). https://doi.org/10.1109/ICPR.2002.1047846

Haindl, M., Filip, J.: Extreme compression and modeling of bidirectional texture function. IEEE Trans. Pattern Anal. Mach. Intell. **29**(10), 1859–1865 (2007). https://doi.org/10.1109/TPAMI.2007.1139

Haindl, M., Filip, J.: Visual Texture. Advances in Computer Vision and Pattern Recognition. Springer, London (2013). https://doi.org/10.1007/978-1-4471-4902-6

Haindl, M., Hatka, M.: BTF Roller. In: Chantler, M., Drbohlav, O. (eds.) Texture 2005. Proceedings of the 4th International Workshop on Texture Analysis, pp. 89–94. IEEE, Los Alamitos (2005a)

Haindl, M., Hatka, M.: A roller – fast sampling-based texture synthesis algorithm. In: Skala, V. (ed.) Proceedings of the 13th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision, pp. 93–96. UNION Agency – Science Press, Plzen (2005b)

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, pp. 297–302. ASCO Art, Bratislava (1998)

Haindl, M., Havlíček, V.: A multiresolution causal colour texture model. Lect. Notes Comput. Sci. (1876), 114–122 (2000)

Haindl, M., Havlíček, V.: Texture editing using frequency swap strategy. In: Jiang, X., Petkov, N. (eds.) Computer Analysis of Images and Patterns. Lecture Notes in Computer Science, vol. 5702, pp. 1146–1153. Springer (2009). https://doi.org/10.1007/978-3-642-03767-2_139

Haindl, M., Havlíček, V.: A compound MRF texture model. In: Proceedings of the 20th International Conference on Pattern Recognition, ICPR 2010, pp. 1792–1795. IEEE Computer Society CPS, Los Alamitos (2010). https://doi.org/10.1109/ICPR.2010.442

Haindl, M., Havlíček, V.: A plausible texture enlargement and editing compound markovian model. In: Salerno, E., Cetin, A., Salvetti, O. (eds.) Computational Intelligence for Multimedia Understanding. Lecture Notes in Computer Science, vol. 7252, pp. 138–148. Springer, Berlin/Heidelberg (2012). https://doi.org/10.1007/978-3-642-32436-9_12, http://www.springerlink.com/content/047124j43073m202/

Haindl, M., Havlíček, V.: Color Texture Restoration, pp. 13–18. IEEE, Piscataway (2015). https://doi.org/10.1109/ICCIS.2015.7274540

Haindl, M., Havlíček, V.: Three-dimensional gaussian mixture texture model. In: The 23rd International Conference on Pattern Recognition (ICPR), pp. 2026–2031. IEEE (2016). https://doi.org/978-1-5090-4846-5/16/\protect\T1\textdollar31.0, http://www.icpr2016.org/site/

Haindl, M., Havlíček, M.: A compound moving average bidirectional texture function model. In: Zgrzynowa, A., Choros, K., Sieminski, A. (eds.) Multimedia and Network Information Systems, Advances in Intelligent Systems and Computing, vol. 506, pp. 89–98. Springer International Publishing (2017a). https://doi.org/10.1007/978-3-319-43982-2_8

Haindl, M., Havlíček, V.: Two compound random field texture models. In: Beltrán-Castañón, C., Nyström, I., Famili, F. (eds.) 2016 the 21st IberoAmerican Congress on Pattern Recognition (CIARP 2016). Lecture Notes in Computer Science, vol. 10125, pp. 44–51. Springer International Publishing AG, Cham (2017b). https://doi.org/10.1007/978-3-319-52277-7_6

Haindl, M., Havlíček, V.: BTF compound texture model with fast iterative non-parametric control field synthesis. In: di Baja, G.S., Gallo, L., Yetongnon, K., Dipanda, A., Castrillon-Santana, M., Chbeir, R. (eds.) Proceedings of the 14th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS 2018), pp. 98–105. IEEE Computer Society CPS, Los Alamitos (2018a). https://doi.org/10.1109/SITIS.2018.00025

Haindl, M., Havlíček, V.: BTF compound texture model with non-parametric control field. In: The 24th International Conference on Pattern Recognition (ICPR 2018), pp. 1151–1156. IEEE (2018b). http://www.icpr2018.org/

Haindl, M., Mikeš, S.: Model-based texture segmentation. Lect. Notes Comput. Sci. (3212), 306–313 (2004)

Haindl, M., Mikeš, S.: Colour texture segmentation using modelling approach. Lect. Notes Comput. Sci. (3687), 484–491 (2005)

Haindl, M., Mikeš, S.: Unsupervised texture segmentation using multispectral modelling approach. In: Tang, Y., Wang, S., Yeung, D., Yan, H., Lorette, G. (eds.) Proceedings of the 18th International Conference on Pattern Recognition, ICPR 2006, vol. II, pp. 203–206. IEEE Computer Society, Los Alamitos (2006). https://doi.org/10.1109/ICPR.2006.1148

Haindl, M., Mikeš, S.: Unsupervised texture segmentation using multiple segmenters strategy. In: Haindl, M., Kittler, J., Roli, F. (eds.) MCS 2007. Lecture Notes in Computer Science, vol. 4472, pp. 210–219. Springer (2007). https://doi.org/10.1007/978-3-540-72523-7_22

Haindl, M., Mikeš, S.: Texture segmentation benchmark. In: Lovell, B., Laurendeau, D., Duin, R. (eds.) Proceedings of the 19th International Conference on Pattern Recognition, ICPR 2008, pp. 1–4. IEEE Computer Society, Los Alamitos (2008). https://doi.org/10.1109/ICPR.2008.4761118

Haindl, M., Šimberová, S.: A multispectral image line reconstruction method. In: Theory & Applications of Image Analysis. Series in Machine Perception and Artificial Intelligence, pp. 306–315. World Scientific, Singapore (1992). https://doi.org/10.1142/9789812797896_0028

Haindl, M., Šimberová, S.: A high – resolution radiospectrograph image reconstruction method. Astron. Astrophys. **115**(1), 189–193 (1996)

Haindl, M., Šimberová, S.: Model-based restoration of short-exposure solar images. In: Abraham, A., Ruiz-del Solar, J., Koppen, M. (eds.) Soft Computing Systems Design, Management and Applications, pp. 697–706. IOS Press, Amsterdam (2002)

Haindl, M., Šimberová, S.: Restoration of multitemporal short-exposure astronomical images. Lect. Notes Comput. Sci. (3540), 1037–1046 (2005)

Haindl, M., Mikeš, S., Pudil, P.: Unsupervised hierarchical weighted multi-segmenter. In: Benediktsson, J., Kittler, J., Roli, F. (eds.) Lecture Notes in Computer Science. MCS 2009, vol. 5519, pp. 272–282. Springer (2009a). https://doi.org/10.1007/978-3-642-02326-2_28

Haindl, M., Mikeš, S., Vácha, P.: Illumination invariant unsupervised segmenter. In: Bayoumi, M. (ed.) IEEE 16th International Conference on Image Processing – ICIP 2009, pp. 4025–4028. IEEE (2009b). https://doi.org/10.1109/ICIP.2009.5413753

Haindl, M., Havlíček, V., Grim, J.: Probabilistic mixture-based image modelling. Kybernetika **46**(3), 482–500 (2011). http://www.kybernetika.cz/content/2011/3/482/paper.pdf

Haindl, M., Remeš, V., Havlíček, V.: Potts compound markovian texture model. In: Proceedings of the 21st International Conference on Pattern Recognition, ICPR 2012, pp. 29–32. IEEE Computer Society CPS, Los Alamitos (2012)

Haindl, M., Mikeš, S., Kudo, M.: Unsupervised surface reflectance field multi-segmenter. In: Azzopardi, G., Petkov, N. (eds.) Computer Analysis of Images and Patterns. Lecture Notes in Computer Science, vol. 9256, pp. 261–273. Springer International Publishing (2015a). https://doi.org/10.1007/978-3-319-23192-1_22

Haindl, M., Remeš, V., Havlíček, V.: BTF Potts Compound Texture Model, vol. 9398, pp. 939807–1–939807–11. SPIE, Bellingham (2015b). https://doi.org/10.1117/12.2077481

Han, J.Y., Perlin, K.: Measuring bidirectional texture reflectance with a kaleidoscope. ACM Trans. Graph. **22**(3), 741–748 (2003)

Heeger, D., Bergen, J.: Pyramid based texture analysis/synthesis. In: ACM SIGGRAPH 95, pp. 229–238. ACM Press (1995)

Holroyd, M., Lawrence, J., Zickler, T.: A coaxial optical scanner for synchronous acquisition of 3D geometry and surface reflectance. ACM Trans. Graph. (Proc. SIGGRAPH 2010) (2010). http://www.cs.virginia.edu/~mjh7v/Holroyd10.php

Kashyap, R.: Analysis and synthesis of image patterns by spatial interaction models. In: Kanal, L., Rosenfeld, A. (eds.) Progress in Pattern Recognition 1. Elsevier, North-Holland (1981)

Kashyap, R.: Image models. In: Young, T.Y., Fu, K.S. (eds.) Handbook of Pattern Recognition and Image Processing. Academic, New York (1986)

Koudelka, M.L., Magda, S., Belhumeur, P.N., Kriegman, D.J.: Acquisition, compression, and synthesis of bidirectional texture functions. In: Texture 2003: Third International Workshop on Texture Analysis and Synthesis, Nice, pp. 59–64 (2003)

Krizhevsky, A.: Learning multiple layers of features from tiny images. Master's thesis, University of Toronto (2009)

Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)

Kwatra, V., Schodl, A., Essa, I., Turk, G., Bobick, A.: Graphcut textures: image and video synthesis using graph cuts. ACM Trans. Graph. **22**(3), 277–286 (2003)

Levada, A., Mascarenhas, N., Tannus, A.: Pseudolikelihood equations for potts mrf model parameter estimation on higher order neighborhood systems. Geosci. Remote Sens. Lett. IEEE **5**(3), 522–526 (2008). https://doi.org/10.1109/LGRS.2008.920909

Li, X., Cadzow, J., Wilkes, D., Peters, R., Bodruzzaman II, M.: An efficient two dimensional moving average model for texture analysis and synthesis. In: Proceedings IEEE Southeastcon'92, vol. 1, pp. 392–395. IEEE (1992)

Liang, L., Liu, C., Xu, Y.Q., Guo, B., Shum, H.Y.: Real-time texture synthesis by patch-based sampling. ACM Trans. Graph. (TOG) **20**(3), 127–150 (2001)

Liu, F., Picard, R.: Periodicity, directionality, and randomness: wold features for image modeling and retrieval. IEEE Trans. Pattern Anal. Mach. Intell. **18**(7), 722–733 (1996). https://doi.org/10.1109/34.506794

Loubes, J., Rochet, P.: Regularization with approximated L^2 maximum entropy method. In: Locally Adaptive Filtering in Signal and Image Processing. Springer, Berlin (2009)

Manjunath, B., Chellapa, R.: Unsupervised texture segmentation using Markov random field models. IEEE Trans. Pattern Anal. Mach. Intell. **13**, 478–482 (1991)

Marschner, S.R., Westin, S.H., Arbree, A., Moon, J.T.: Measuring and modeling the appearance of finished wood. ACM Trans. Graph. **24**(3), 727–734 (2005)

Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings of 8th International Conference on Computer Vision, vol. 2, pp. 416–423 (2001). http://www.cs.berkeley.edu/projects/vision/grouping/segbench/

Matuszak, M., Schreiber, T.: Locally specified polygonal Markov fields for image segmentation. In: Locally Adaptive Filtering in Signal and Image Processing. Springer, Berlin (2009)

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: Equation of state calculations by fast computing machines. J. Chem. Phys. **21**, 1087–1092 (1953)

Mikeš, S., Haindl, M.: View dependent surface material recognition. In: Bebis, G., Boyle, R., Parvin, B., Koračin, D., Ushizima, D., Chai, S., Sueda, S., Lin, X., Lu, A., Thalmann, D., Wang, C., Xu, P. (eds.) 14th International Symposium on Visual Computing (ISVC 2019). Lecture Notes in Computer Science, vol. 11844, pp. 156–167. Springer Nature Switzerland AG (2019). https://doi.org/10.1007/978-3-030-33720-9_12, https://www.isvc.net/

Müller, G., Meseth, J., Klein, R.: Compression and real-time rendering of measured BTFs using local PCA. In: Vision, Modeling and Visualisation 2003, pp. 271–280 (2003)

Müller, G., Meseth, J., Sattler, M., Sarlette, R., Klein, R.: Acquisition, synthesis and rendering of bidirectional texture functions. In: Eurographics 2004, STAR – State of The Art Report, Eurographics Association, pp. 69–94 (2004)

Neubeck, A., Zalesny, A., Gool, L.: 3D texture reconstruction from extensive BTF data. In: Chantler, M., Drbohlav, O. (eds.) Texture 2005. Heriot-Watt University, Edinburgh (2005)

Ngan, A., Durand, F.: Statistical acquisition of texture appearance. In: Eurographics Symposium on Rendering, Eurographics (2006)

Ojala, T., Maenpaa, T., Pietikainen, M., Viertola, J., Kyllonen, J., Huovinen, S.: Outex: new framework for empirical evaluation of texture analysis algorithms. In: International Conference on Pattern Recognition, pp. I:701–706 (2002)

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

Panjwani, D., Healey, G.: Markov random field models for unsupervised segmentation of textured color images. IEEE Trans. Pattern Anal. Mach. Intell. **17**(10), 939–954 (1995)

Pattanayak, S.: Pro Deep Learning with TensorFlow. Apress (2017). https://doi.org/10.1007/978-1-4842-3096-1

Polzehl, J., Tabelow, K.: Structural adaptive smoothing: principles and applications in imaging. In: Locally Adaptive Filtering in Signal and Image Processing. Springer, Berlin (2009)

Portilla, J., Simoncelli, E.: A parametric texture model based on joint statistics of complex wavelet coefficients. Int. J. Comput. Vis. **40**(1), 49–71 (2000)

Potts, R., Domb, C.: Some generalized order-disorder transformations. In: Proceedings of the Cambridge Philosophical Society, vol. 48, pp. 106–109 (1952)

Praun, E., Finkelstein, A., Hoppe, H.: Lapped textures. In: ACM SIGGRAPH 2000, pp. 465–470 (2000)

Rainer, G., Ghosh, A., Jakob, W., Weyrich, T.: Unified neural encoding of BTFs. In: Computer Graphics Forum, vol. 39, pp. 167–178. Wiley Online Library (2020)

Reed, T.R., du Buf, J.M.H.: A review of recent texture segmentation and feature extraction techniques. CVGIP–Image Underst. **57**(3), 359–372 (1993)

Ren, P., Wang, J., Snyder, J., Tong, X., Guo, B.: Pocket reflectometry. ACM Trans. Graph. (Proc. SIGGRAPH) **30**(4) (2011). https://doi.org/10.1145/2010324.1964940

Ruiters, R., Schwartz, C., Klein, R.: Example-based interpolation and synthesis of bidirectional texture functions. In: Computer Graphics Forum, vol. 32, pp. 361–370. Wiley Online Library (2013)

Sattler, M., Sarlette, R., Klein, R.: Efficient and realistic visualization of cloth. In: Eurographics Symposium on Rendering (2003)

Schwartz, C., Sarlette, R., Weinmann, M., Rump, M., Klein, R.: Design and implementation of practical bidirectional texture function measurement devices focusing on the developments at the university of bonn. Sensors **14**(5), 7753–7819 (2014). https://doi.org/10.3390/s140507753. http://www.mdpi.com/1424-8220/14/5/7753

Sharma, M., Singh, S.: Minerva scene analysis benchmark. In: Seventh Australian and New Zealand Intelligent Information Systems Conference, pp. 231–235. IEEE (2001)

Soler, C., Cani, M., Angelidis, A.: Hierarchical pattern mapping. ACM Trans. Graph. **21**(3), 673–680 (2002)

Swendsen, R.H., Wang, J.S.: Nonuniversal critical dynamics in Monte Carlo simulations. Phys. Rev. Lett. **58**(2), 86–88 (1987). https://doi.org/10.1103/PhysRevLett.58.86

Tong, X., Zhang, J., Liu, L., Wang, X., Guo, B., Shum, H.Y.: Synthesis of bidirectional texture functions on arbitrary surfaces. ACM Trans. Graph. (TOG) **21**(3), 665–672 (2002)

Tsai, Y.T., Shih, Z.C.: K-clustered tensor approximation: a sparse multilinear model for real-time rendering. ACM Trans. Graph. **31**(3), 19:1–19:17 (2012). https://doi.org/10.1145/2167076.2167077

Tsai, Y.T., Fang, K.L., Lin, W.C., Shih, Z.C.: Modeling bidirectional texture functions with multivariate spherical radial basis functions. Pattern Anal. Mach. Intell. IEEE Trans. **33**(7), 1356 –1369 (2011). https://doi.org/10.1109/TPAMI.2010.211

Vacha, P., Haindl, M.: Image retrieval measures based on illumination invariant textural mrf features. In: CIVR'07: Proceedings of the 6th ACM International Conference on Image and Video Retrieval, pp. 448–454. ACM Press, New York (2007). https://doi.org/10.1145/1282280.1282346

Varma, M., Zisserman, A.: A statistical approach to texture classification from single images. Int. J. Comput. Vis. **62**(1–2), 61–81 (2005)

Wang, J., Dana, K.: Relief texture from specularities. IEEE Trans. Pattern Anal. Mach. Intell. **28**(3), 446–457 (2006)

Wei, L., Levoy, M.: Texture synthesis using tree-structure vector quantization. In: ACM SIGGRAPH 2000, pp. 479–488. ACM Press/Addison Wesley/Longman (2000). citeseer.nj.nec.com/wei01texture.html

Wei, L., Levoy, M.: Texture synthesis over arbitrary manifold surfaces. In: SIGGRAPH 2001, pp. 355–360. ACM (2001)

Wu, F.: (1982) The Potts model. Rev. Modern Phys. **54**(1), 235–268

Wu, H., Dorsey, J., Rushmeier, H.: A sparse parametric mixture model for BTF compression, editing and rendering. Comput. Graph. Forum **30**(2), 465–473 (2011)

Xu, Y., Guo, B., Shum, H.: Chaos mosaic: fast and memory efficient texture synthesis. Technical Report MSR-TR-2000-32, Redmont (2000)

Zelinka, S., Garland, M.: Towards real-time texture synthesis with the jump map. In: 13th European Workshop on Rendering, p. 99104 (2002)

Zelinka, S., Garland, M.: Interactive texture synthesis on surfaces using jump maps. In: Christensen, P., Cohen-Or, D. (eds.) 14th European Workshop on Rendering, Eurographics (2003)

Zhang, Y.J.: Evaluation and comparison of different segmentation algorithms. Pattern Recogn. Lett. **18**, 963–974 (1997)

Zhang, J.D., Zhou, K., Velho ea, L.: Synthesis of progressively-variant textures on arbitrary surfaces. ACM Trans. Graph. **22**(3), 295–302 (2003)

Zhu, S., Liu, X., Wu, Y.: Exploring texture ensembles by efficient Markov Chain Monte Carlo – toward a "trichromacy" theory of texture. IEEE Trans. Pattern Anal. Mach. Intell. **22**(6), 554–569 (2000)