STRUCTURE-BASED SYNTHESIS OF MATERIAL SAMPLES

ADIB AKL

School of Engineering, Holy Spirit University of Kaslik (USEK), P.O. Box 446, Jounieh, Lebanon
E-mail: adibakl@usek.edu.lb

Abstract - This paper presents an image processing algorithm allowing to synthesize material textures of arbitrary shapes. The method consists in constructing, using an arbitrary image, a structure layer of the material. The obtained structure layer then constrains the synthesis of the material texture. The structure layer is represented by the field of second-moment matrices. To ensure that the method faithfully reproduces the visual aspect and characteristics of the input material sample, several tests using various constraint images are implemented. The obtained results highlight the capability of the proposed algorithm to synthesize arbitrary-shaped material textures while respecting the local characteristics of the exemplar. This paves the way toward the synthesis of 3D material textures of arbitrary shapes from 2D material samples, with applications in virtual material design and materials characterization.

Keywords - Local orientation, Material sample, Second-moment matrix, Structure, Synthesis

I. INTRODUCTION

Materials science which started by materials discovered in nature, is quickly evolving to designed materials that are able to adapt to their environment. Starting from atoms and molecules, materials are now designed from scratch [1].

The evolution of materials science is driven by the synergy of physics, mechanics and engineering where image processing reveals important contribution in this challenge [2]. In particular, it is crucial to design and analyze materials by synthesizing new models from two or three-dimensional image models, with applications in the characterization of the physical assembly process of materials, virtual material design, studying imaging techniques interaction with materials, separating materials appearance from their intrinsic properties and obtaining a realistic description of the material structure. However, the appearance of the material normally varies with image scale, image settings and imaging techniques [3, 4].

The diversity of textured images [5] has led to different synthesis algorithms. For instance, the method proposed in [6] uses the nearest neighbor search to generate a new textured image by sampling values from the initial input sample [5-6]. The non-parametric method of Wei and Levoy [7], relying on a Markov random field, models the texture as a stationary and local random process. The authors in [8] propose a synthesis method consisting in collecting the pixels following an already specified order, then synthesizing the output pixels based on a probability density function. An image synthesis algorithm based on the Parzen window is also proposed in [9]. A method that copies patches from the input to the output sample then stitches them together along optimal seams is proposed in [10]. Other authors use the most similar neighborhood search [11]. The algorithm is then accelerated by reducing the number of neighborhoods to compare with. Model-based synthesis methods were also proposed [12, 13]. These methods consist in generating a probabilistic model used to describe and synthesize the image, where the parameters of the model capture the image visual characteristics.

Most of these image synthesis methods show success in synthesizing textured images of different types. However, failure cases still exist, especially when dealing with complex images with high local structural variations.

This paper proposes an image synthesis algorithm allowing to synthesize material textures of arbitrary shapes. The algorithm proceeds in three steps: (i) a *constraint* field of second-moment matrices is created from an arbitrary chosen textured image; (ii) the initial textured image is used to construct a dictionary of intensities and second-moment matrices; (iii) the obtained *constraint* field of second-moment matrices is used to constrain the synthesis of the textured image by searching the best resemblance within the dictionary.

The paper is organized as follows: section 2 presents the creation of the *constraint* field of second-moment matrices from an imposed arbitrary texture. The construction of the searching dictionary is detailed in section 3. The synthesis process, constrained by the field of second-moment matrices, is presented in section 4. Results are discussed in section 5. Finally, conclusions are drawn in section 6.

II. CONSTRUCTION OF THE *CONSTRAINT*

2. 1. The Second-Moment Matrix

The second-moment matrix is a mathematical tool, commonly used in image processing to describe local patterns [14]. The second-moment matrix field *SM* (Equation (1)) of an image *IM* assigns a matrix for each pixel (*x*,*y*) of the image. It is defined as the field of local covariance matrices of the first partial derivatives of *IM* [15]:

$$SM = G_\sigma * \nabla IM \nabla IM^t = G_\sigma * \begin{bmatrix} IM_x^2 & IM_x IM_y \\ IM_x IM_y & IM_y^2 \end{bmatrix}, \quad (1)$$

where '*' denotes convolution, $\nabla IM = [IM_x \; IM_y]$ are the gradient fields of image *IM*, G_σ is a Gaussian weighting kernel used to smooth the matrix and $[\,]^t$ is the matrix transpose operator.

The orientation factor $O(x,y)$ (Equation 2) of a second-moment matrix $SM(x,y)$, varying between $-\pi/2$ and $\pi/2$, is calculated from its first eigenvector $[e_x \; e_y]$ as follows:

$$O(x,y) = \tan^{-1}\left(\frac{e_y}{e_x}\right) \quad (2)$$

An orientation field *O* can then be extracted from a second-moment matrix field *SM*. It consists of an orientation image presenting at each position (x,y), the orientation factor $O(x,y)$ of the corresponding matrix $SM(x,y)$ at the same position. The palette used to represent orientation fields is shown in Figure 1. This palette is used for all the illustrations and results in the paper.



Figure 1. Palette used to represent orientation fields.

2. 2. *Constraint* Second-Moment Matrix Field Calculation

The first step of the proposed synthesis algorithm is the construction of the *constraint* field of second-moment matrices SM_{con} from an imposed arbitrary textured image IM_{con}. This consists in three steps:

- Computation of the second-moment matrix field SM_{con} of IM_{con} and its associated orientation field O_{con}.
- Computation of the second-moment matrix field SM_{in} of the input material texture sample IM_{in}.
- Modulation of the orientations of SM_{in} by O_{con}. This is applied as follows:

At each position (x,y) of SM_{in}, we apply a rotation for matrix $SM_{in}(x,y)$ by the orientation angle $O_{con}(x,y)$, which is the angle at the corresponding position in the orientation field O_{con}. The *constraint* second-moment matrix field is then obtained by updating SM_{con}, at each position (x,y), as follows:

$$SM_{con}(x,y) \longleftarrow R(x,y).SM_{in}(x,y).R(x,y)^t, \quad (3)$$

where '.' is the matrix product and $R(x,y)$ is the rotation matrix, at position (x,y), given by:

$$R(x,y) = \begin{bmatrix} cos\{O_{con}(x,y)\} & sin\{O_{con}(x,y)\} \\ -sin\{O_{con}(x,y)\} & cos\{O_{con}(x,y)\} \end{bmatrix} \quad (4)$$

This method ensures that the matrices of the constructed *constraint* second-moment matrix field SM_{con} are rotated versions of the input field of matrices SM_{in} representing then the structure of the imposed arbitrary texture IM_{con}.

III. CONSTRUCTION OF THE SEARCHING DICTIONARY

The second step of the algorithm consists in creating, from the input material texture sample IM_{in} and its field of second-moment matrices SM_{in}, a dictionary of intensities and matrices.

Let $D(IM_{in})$ the dictionary created from IM_{in}: $D(IM_{in}) = \{N_k, k = 1...K\}$, a set of K two-dimensional neighborhoods N_k, where K is the number of sites (i.e. size) of IM_{in}. Each neighborhood N_k is given by:

$$N_k = N_k^{IM_{in}} \cup N_k^{SM_{in}}, \quad (5)$$

where $N_k^{IM_{in}}$ is the kth intensity neighborhood in IM_{in} and $N_k^{SM_{in}}$ is the corresponding second-moment matrix neighborhood in SM_{in}.

Note that both not causal square neighborhoods and causal L-shaped neighborhoods can be used.

In order to ensure that the different structures of the *constraint* second-moment matrix field SM_{con} exist in the dictionary, this latter is replaced by a dictionary $D_M(IM_{in})$ which includes $D(IM_{in})$. $D_M(IM_{in})$ consists of neighborhoods extracted from images $\{Rot_m(IM_{in}), m = 0,...,M-1\}$, where $Rot_m(IM_{in})$ is the rotation of IM_{in} by an angle $\varphi_m = 2\pi m/M$, together with the corresponding second-moment matrices.

Figure 2 presents an example of rotated versions created from an initial material sample. The figure shows, from upper-left to bottom-right, an initial anisotropic composite material sample, rotated versions by angles of π/6 and π/3. The orientation images extracted from the corresponding second-moment matrix fields are shown at the right of each image. Those images and matrix fields are at the basis of the construction of the dictionary used in the synthesis process.

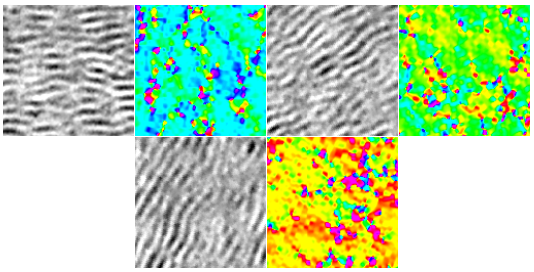

Figure 2. Example of rotated versions created from initial material sample. From upper-left to bottom-right: material sample, rotated versions by $\pi/6$ and $\pi/3$. Orientation images are shown at the right of each image.

IV. THE SYNTHESIS PROCESS

Having the *constraint* second-moment field SM_{con} and the dictionary $D_M\left(IM_{in}\right)$,the output texture IM_{out} is initialized by values chosen randomly from the input material sample IM_{in}, and then modified during the synthesis process as follows: for each position o=(x,y) in IM_{out}, the neighborhood $N_o = N_o^{IM_{out}} \cup N_o^{SM_{con}}$ is extracted ($N_o^{IM_{out}}$ is the neighborhood of intensities at position o in IM_{out} and $N_o^{SM_{con}}$ is the second-moment matrix neighborhood at the corresponding position in SM_{con}). The best resemblance is identified in $D_M\left(IM_{in}\right)$ and the corresponding pixel is copied to the target position o in IM_{out}.

The metric used to calculate the resemblance between two neighborhoods N_k and N_o is the Sum of Squared Errors (SSE) for intensity neighborhoods, combined with the Sum of Second-moment Matrices Dissimilarities (SSMD), defined in Equation (7), for the second-moment matrix neighborhoods:

$$D\left(N_k, N_o\right) = W \cdot SSE\left(N_k^{IM_{in}}, N_o^{IM_{out}}\right) + \left(1-W\right) \cdot SSMD\left(N_k^{SM_{in}}, N_o^{SM_{con}}\right), \quad (6)$$

where $N_k^{IM_{in}}$ and $N_k^{SM_{in}}$ are respectively the kth intensity and second-moment matrix neighborhoods in dictionary $D_M\left(IM_{in}\right)$, $N_o^{IM_{out}}$ and $N_o^{SM_{con}}$ are respectively the intensity and second-moment matrix neighborhoods at position o in IM_{out} and the corresponding position in SM_{con}, W is a weighting factor and *SSMD* is expressed by:

$$SSMD\left(N_1^{SM_1}, N_2^{SM_2}\right) = \sum_{i=1}^{N_M} Dist\left(N_1^{SM_1}(i), N_2^{SM_2}(i)\right), \quad (7)$$

where N_M is the number of matrices in each neighborhood, $N^{SM}(i)$ is the ith second-moment matrix in neighborhood N^{SM} and *Dist* is one of the four metrics of second-moment matrix resemblance proposed in [14]. In this paper, the Euclidean distance is used.

The pseudocode of the whole synthesis algorithm is presented in Figure 3.

When dealing with large-scale material samples, using large neighborhoods is expensive in terms of computational complexity. Gaussian pyramids of images and second-moment matrix fields [14] are then used to retranscribe the structures at different scales. Thus, the dictionary $D_M\left(IM_{in}\right)$ is constructed using multi-scale neighborhoods extracted from intensity and matrix pyramids. The pyramids are built from M rotated textures and from their corresponding second-moment matrix fields. Similarly, texture and matrix pyramids are built from IM_{out} and SM_{con}, respectively. The synthesis process starts at the highest level of the pyramid down to the lowest level. More precisely, in order to ensure that the added high frequency details are compatible with the already synthesized low frequency structures, the multi-resolution neighborhood at position o=(x,y) of level L contains its neighborhood at the same level as well as the neighborhood of the corresponding position o'=(x/2,y/2) at the previously synthesized level L+1 [7].

Input: IM_{in} & IM_{con}

SM_{con} ← CONSTRAINTCALCULATION (IM_{in} & IM_{con})
$D_M\left(IM_{in}\right)$ ← DICTIONARYCONSTRUCTION (IM_{in} & SM_{in})
IM_{out} ← NOISEINITIALIZATION (IM_{in})

For each position o=(x, y) in IM_{out} and SM_{con}
PIX_k ← *argmax*{BESTRESEMBLANCE (N_k VS N_o)}
$IM_{out}(x, y)$ ← PIX_k
End For

Output: IM_{out}

Figure 3. Pseudocode of the proposed synthesis method.

V. RESULTS

For space limitation reasons, a subset of the results obtained with the proposed algorithm is presented in this section. The left and right columns of Figure 4 show respectively, two different results obtained using imposed textures having waves and semicircles shapes. The input texture used in both results is the anisotropic composite material sample shown in Figure 2. The first to fifth rows of the figure show, the arbitrarily imposed texture IM_{con}, its field of second-moment matrices represented by its orientation image O_{con}, the output texture (IM_{dir}) directly synthesized by using the field of second-moment matrices of IM_{con} as a constraint for the synthesis process, the constraint second-moment matrix field (SM_{con}) constructed using the proposed approach and the output texture (IM_{out}) obtained by using SM_{con} to constrain the synthesis. A searching dictionary with an angular step of 5° is used in both results. Gaussian pyramids of two and three scales are used in the first and second result, respectively. All the presented results are obtained after applying three iterations with a causal neighborhood and a lexicographical scan.

It can be observed in Figure 4 that the proposed method succeeds in constructing a second-moment matrix field that represents the structures (orientations) of the arbitrarily imposed texture.

Consequently, the synthesis process constrained by the obtained field of second-moment matrices leads to restructured textures of arbitrary shapes covered by the patterns of the exemplar, while respecting its dynamics and visual aspects. In contrast, the synthesis process constrained by the second-moment matrix field of the arbitrary textures leads to images having regular patterns that do not exist in the input material sample. This justifies the importance of the constraint second-moment matrix field instauration algorithm.

Note that the proposed algorithm assumes that an input material sample having the same size of the arbitrary texture is available. Otherwise, a solution consists in synthesizing from the input sample, which usually has a small size, a textured image of same size as the arbitrarily imposed texture. The resulting texture is then used for the instauration of the constraint second moment matrix field.

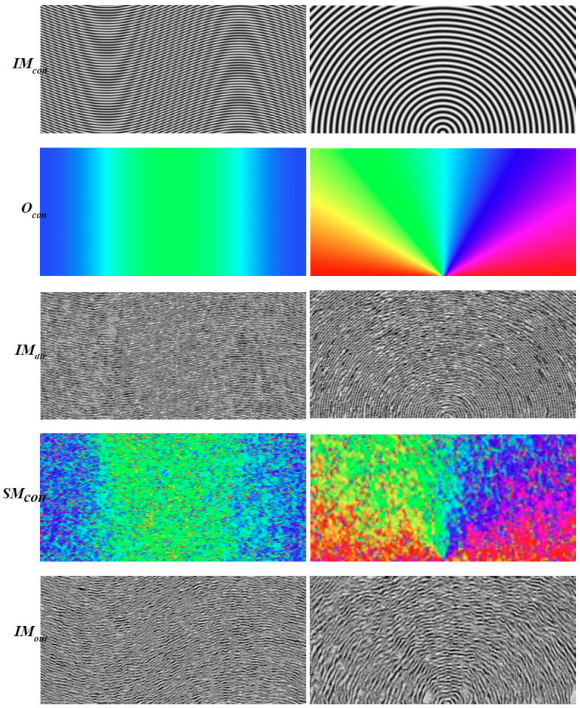

Figure 4. Two synthesis results obtained using the proposed algorithm. 1st to 5th rows: arbitrarily imposed textures, their orientation images, synthesized textures obtained using the second-moment matrix fields of the arbitrary textures to constrain the synthesis process, orientation images of the constructed constraint second-moment matrix fields and the output textures obtained using these constraint fields in the synthesis. The input texture used in both results is the material sample shown in Fig. 2.

VI. CONCLUSION

We have proposed a texture synthesis algorithm allowing to synthesize textures of arbitrary shapes. A *constraint* second-moment matrix field is modulated by the orientations of an imposed texture, and then used to constrain the synthesis of the output texture. Results show that the proposed method succeeds in synthesizing textured images of specified shapes while preserving the essential local characteristics of the material exemplar. As for future work, we aim at extending the proposed approach to synthesize arbitrary-shaped 3D textures from 2D material samples which would be beneficial in the analysis and design of virtual materials such as carbon materials, for example.

ACKNOWLEDGMENT

This project has been jointly funded with the support of the National Council for Scientific Research in Lebanon CNRS-L and the Holy Spirit University of Kaslik.

REFERENCES

[1] Bréchet, Y., " Materials science: from materials discovered by chance to made-to-measure materials", collège de France, (2013).

[2] Liu, Y. and Zhang, X., "Metamaterials: a new frontier of science and technology", *Chem. Soc. Rev.*, Vol. 40, (2011), 2494.

[3] Leyssale, J.-M., Da Costa, J.-P., Germain, C., Weisbecker, P. and Vignoles, G., "An image guided atomistic reconstruction of pyrolytic carbons", *App Phys Lett.*, Vol. 95, (2009), 231912.

[4] Leyssale, J.-M., Da Costa, J.-P., Germain, C., Weisbecker, P. and Vignoles, G., "Structural features of pyrocarbon atomistic models constructed from transmission electron microscopy images", *Carbon.*, Vol. 50, (2012), 4388-4400.

[5] Forsyth, D.A. and Ponce, J., "Texture, Computer Vision, A Modern Approach", Pearson Education, (2003).

[6] Efros, A. and Leung, T., "Texture synthesis by non-parametric sampling", in International Conference on Computer Vision , 2, (1999), 1033–1038.

[7] Wei, L.-Y. and Levoy, M.,"Fast texture synthesis using tree-structured vector quantization", in International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH), (2000), 479-488.

[8] Popat, K. and Picard, R. W., "Novel cluster-based probability model for texture synthesis, classification, and compression", in SPIE visual Communications and Image Processing, Boston, (1993).

[9] Paget, R. and Longstaff, I., "Texture synthesis via a noncausal nonparametric multiscale Markov random field", IEEE Transactions on Image Processing, Vol.7, (1998), 925-931.

[10] Kwatra, V., Schödl, A., Essa, I., Turk, V and Bobick, V., "Graphcut Textures: Image and Video Synthesis Using Graph Cuts", in ACM SIGGRAPH, (2003), 277-286.

[11] Ashikhmin, M., "Synthesizing natural textures", in 2001 Symposium on Interactive 3D Graphics, New York, NY, USA. ACM Press, (2001), 217–226.

[12] Cross, V. and Jain, A., "Markov Random Field Texture Models", IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-5,(1983), 25-39.

[13] Chellappa, V. and Kashyap, R.L, "Texture synthesis using 2-D noncausal autoregressive Models", IEEE Trans. Acoust. Speech Signal Process, Vol. 33, (1985), 194–203.

[14] Akl, A., Yaacoub, C., Donias, M., Da Costa, J.-P. and Germain, C., "Structure Tensor Based Synthesis of Directional Textures for Virtual Material Design", in IEEE International Conference on Image Processing (ICIP), (2014), 4867-4871.

[15] Akl, A., Yaacoub, C., Donias, M., Da Costa, J.-P. and Germain, C, "Texture Synthesis using the Structure Tensor", IEEE Transactions on Image Processing, Vol. 24, (2015), 4082-4095.

★★★