Review

Bridges between multiple-point geostatistics and texture synthesis: Review and guidelines for future research

Gregoire Mariethoz [a,*], Sylvain Lefebvre [b]

[a] School of Civil and Environmental Engineering, University of New south Wales, Sydney, Australia
[b] ALICE team, INRIA, Nancy, France

ABSTRACT

Multiple-Point Simulations (MPS) is a family of geostatistical tools that has received a lot of attention in recent years for the characterization of spatial phenomena in geosciences. It relies on the definition of training images to represent a given type of spatial variability, or texture. We show that the algorithmic tools used are similar in many ways to techniques developed in computer graphics, where there is a need to generate large amounts of realistic textures for applications such as video games and animated movies. Similarly to MPS, these texture synthesis methods use training images, or exemplars, to generate realistic-looking graphical textures.

Both domains of multiple-point geostatistics and example-based texture synthesis present similarities in their historic development and share similar concepts. These disciplines have however remained separated, and as a result significant algorithmic innovations in each discipline have not been universally adopted. Texture synthesis algorithms present drastically increased computational efficiency, patterns reproduction and user control. At the same time, MPS developed ways to condition models to spatial data and to produce 3D stochastic realizations, which have not been thoroughly investigated in the field of texture synthesis.

In this paper we review the possible links between these disciplines and show the potential and limitations of using concepts and approaches from texture synthesis in MPS. We also provide guidelines on how recent developments could benefit both fields of research, and what challenges remain open.

© 2014 Elsevier Ltd. All rights reserved.

Contents

* Corresponding author. Tel.: +61 4 1566 2986; fax: +61 2 9385 6139.
  E-mail address: gregoire.mariethoz@minds.ch (G. Mariethoz).

1. Introduction

Numerical modeling is widely used to characterize natural phenomena. A major challenge in spatially distributed models is that most natural processes are heterogeneous: that is, strongly varying in space and in time. Moreover in many situations, models are informed by insufficient data, resulting in significant uncertainty in prediction outcomes. This problem has classically been addressed by geostatistical methods, which offer a framework to interpolate data and quantify the related uncertainty. The traditional geostatistical toolbox consists in theories such as variogram modeling, kriging and indicator simulation, which are well established and widely accepted (Chilès and Delfiner, 1999; Cressie and Wikle, 2011; Deutsch and Journel, 1992; Goovaerts, 1997; Kitanidis, 1997).

In contrast to these classical methods, a number of new methods have been developed in the last decade, known as Multiple-Point Simulations (MPS). They focus on modeling the spatial patterns of natural phenomena based on training images (TIs), which are explicit examples of the heterogeneity that is believed to be present. Although the principle of using training images is well established, several of issues remain for the concept to be widely used in geosciences. Most of these issues pertain to the development of efficient algorithms dealing with pattern matching, searching and synthesis. Existing algorithms can fail to properly represent complex connected features, and their applicability is compromised by heavy computational cost, making it difficult to use Monte-Carlo analysis for uncertainty estimation and resolution of inverse problems (Hendricks Franssen et al., 2009; Kitanidis, 1995; Zimmerman et al., 1998). Important research efforts have been devoted to solve such algorithmic issues in the last years.

In several aspects, MPS methods are very similar in their nature to techniques developed for animation movies, special effects and video games, where graphical realism and computationally efficient processing are key aspects. However, to this day the links between these two worlds have not been explored. The aim of this paper is to fill this gap by exploring the bridges to be made between multiple-point geostatistics and graphical texture synthesis, and to shed light on a host of possible developments that can emerge from adopting this interdisciplinary perspective. It is beyond our scope to provide a comprehensive review of either MPS or texture synthesis algorithms, in part because such reviews already exist (Hu and Chugunova, 2008; Wei et al., 2009), but also because our aim is to explore the links between both disciplines, and hence we chose to leave aside a number of approaches for which no such cross-discipline components have been identified. We focus in particular on the developments that we believe can lead to promising future research directions.

This paper starts by providing background information on MPS and the use of training images for modeling spatial processes. We then retrace some historical commonalities between the disciplines of geostatistics and example-based texture synthesis. The remaining sections review algorithmic aspects of example-based texture synthesis relevant for spatial models of natural phenomena. These include strategies for texture generation (pixel-based, patch-based, optimization-based), approaches to spatial data integration, the passage to 3D textures and issues related to computational efficiently. We conclude by recommending specific developments that can benefit both disciplines.

2. Background on multiple-point geostatistics

A large part of the success of geostatistics has been based on its ability to formulate the lack of spatial knowledge as a mathematical model of uncertainty. Variogram-based methods allow for defining spatial heterogeneity with a limited number of parameters such as mean, variance and variogram correlation range. In addition, these parameters can be conveniently inferred from the available data. Geostatistical simulation offers the possibility to generate scenarios of what an unknown reality could be. These present three essential requirements:

- *physical realism* entails that models present the same spatial/temporal continuity as the reality they are supposed to represent,
- *stochasticity* means that the models appropriately cover the space of uncertainty and
- *conditioning* ensures that they are constrained by data measured locally.

These tools have been heavily used in domains as diverse as hydrology (Goovaerts, 2000), hydrogeology (de Marsily et al., 2005; Goovaerts, 2000; Huysmans and Dassargues, 2012; Matheron, 1967), ecology (Relethford, 2008; ver Hoef, 2008), epidemiology (Goovaerts, 2010; Meliker et al., 2009), paleoclimate (Mariethoz et al., 2012), remote sensing (Atkinson et al., 2008; Jha et al., 2013a; Tang et al., 2013; Tatem et al., 2002), fishery (Ciannelli et al., 2008; Rufino et al., 2006), natural resources estimation (Caers, 2011; Chilès and Delfiner, 1999; Cressie and Wikle, 2011; Goovaerts, 1997; Lantuejoul, 2002; Matheron, 1965), with specific methods developed for petroleum engineering (Caers, 2005) and mining (Isaaks, 1984; Journel and Huijbregts, 1978). Some applications also took place outside the field of geosciences, such as in finance (Kanevski et al., 2008) or in medical imaging (Pham, 2012).

However, the convenience achieved by variogram-based approach comes at the price of strong assumptions regarding the modeling of specific patterns, in particular for the representation of connectivity (Journel and Zhang, 2006; Neuweiler and Vogel, 2007; Zinn and Harvey, 2003). The traditional tools are therefore not applicable for modeling real-world natural systems where such characteristics are important (Gómez-Hernández and Wen, 1998; Neuweiler et al., 2011; Sánchez-Vila et al., 1996; Western et al., 2001), raising concerns that the uncertainty quantification obtained with variogram-based methods might not entirely reflect the actual state of knowledge on a given problem (Kerrou et al., 2008).

Alternatives to variogram-based tools have been sought in the past decades to better constrain models to prior information. On the one hand, more elaborate parametric models have been developed, such as object-based methods that are especially used for geological modeling (Deutsch and Tran, 2002; Haldorsen and Chang, 1986; Webb and Anderson, 1996) but also find applications in rainfall simulation (Zhang and Switzer, 2007). Other high-order parametric approaches include spatial cumulants (Dimitrakopoulos et al., 2010; Mustapha et al., 2011) and copulas (Bárdossy and Li, 2008; Haslauer et al., 2012), which extend the concept of variograms to higher-order statistics. On the other hand, non-parametric approaches have also received a lot of attention. Here we define a non-parametric approach as using more than a few parameters. In particular, multiple-point simulations were developed to generate realizations of a spatial phenomenon conditioned by a description of the processes taking the form of a training image. Its premise is that the subjective information can be conveniently formulated as an "example". In this context, the use of training images is justified for several reasons:

(1) The data available are often too sparse to inform a non-parametric spatial model. However a training image is a spatially exhaustive source of patterns for such applications.
(2) When large amounts of spatial data are available, such as provided by remote sensing, it can be used as training image,

allowing for spatial modeling that considers the full richness contained in the data.

(3) For applications involving geological modeling, there is often an important subjective or interpretative component to spatial characterization (Frodeman, 1995). Training images have been one of the best ways of translating geological concepts into a computer modeling framework.

Conversely, there are cases where training image-based approaches are not directly applicable, for example when modeling extreme values.

A fundamental reason for the emergence of the concept of training image-based modeling is that stochastic models derived from parametric approaches, although statistically sound, are often deemed unrealistic from a subjective point of view. Training images have then been used to formalize the notion of "realism". The challenge in this regard is that it involves sources of knowledge that have a completely different formalism than most types of data classically measured in the field. The general conceptualization of a geological model by a training image is illustrated in Fig. 1. The training image (lower right) represents a geological conceptual model. One MPS realization using the method of Mariethoz and Kelly (2011) is shown, which is constrained by vertical wells data (binary data indicating the presence/absence of channels). Note that conditioning is not only local, but involves reproducing statistics of the data. In this example, the wells indicate a proportion of channels of 0.5, which is reproduced in the simulated model. Higher-order statistics could also be considered. Such type of conditioning is typical of geoscience problems and usually not considered in texture synthesis.

The use of training images is a radical change of perspective compared to the standard approach where a statistical model is adjusted based only on data. Instead, it builds on the principle of using analogues to construct stochastic representations of natural systems that lie at the intersection of example-driven spatial patterns coming from a training image and data-based constraints that may include physical laws, data and statistical inference. Fig. 2 illustrates this dual set of constraints: the red circle represents the models obtained solely by statistical inference (e.g. variogram-based), and the blue circle denotes the models constrained by the subjective notion of realism. The intersection of both circles results in an ensemble of models that are coherent with all available sources of information. In applications where data are scarce and contain low information content, the red circle would expand, encompassing more possible models. For example in the initial stage of a hydrogeological study, most of the knowledge is conceptual, and therefore a training image may be the only means of constraining the model space (Ronayne et al., 2008). As more data is acquired, the red ensemble becomes narrower, thus reducing uncertainty. Conversely, in the case of soil moisture estimation using satellite observations,
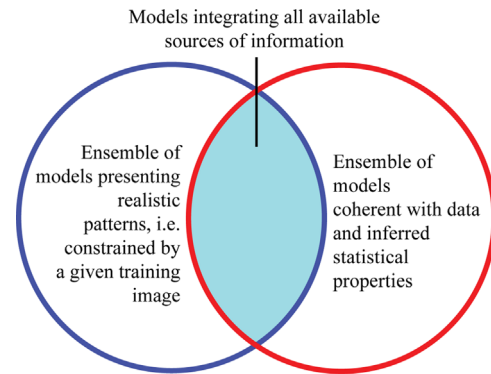

Fig. 2. Illustration of the duality in the constraints that apply for spatial modeling of natural phenomena. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

the dense data informs the spatial structure and at the same time provides local information (Jha et al., 2013a). In this situation, both red and blue circles largely overlap.

3. A brief historical perspective

Both MPS and example-based texture synthesis find their origins in Markov Random Fields (MRF), which is a general probabilistic formulation for spatial variables (Besag et al., 1991; Cressie and Wikle, 2011; Toftaker and Tjelmeland, 2013). The MRF approach is based on defining parameters that fully characterize a spatial law. These parameters are then inferred, generally using Markov chain Monte Carlo methods. Due to difficulties involved in determining these parameters, less rigorous but more efficient alternatives have been developed, where statistical parameters are imposed by construction rather than sampled (Deutsch and Journel, 1992). MPS was first investigated in this context (Guardiano and Srivastava, 1993; Journel and Alabert, 1989). Although the concept of using training images for geological representation was appealing, the first MPS algorithm developed was inefficient because it involved scanning the entire training image for each pixel in the simulation. The method could therefore not be used routinely. Concurrently, the earliest developments in example-based texture synthesis took place independently at about the same period (Popat and Picard, 1993). It should be noted here that earlier texture synthesis approaches have been proposed, and the field of *procedural* texture synthesis is still an active one (Lagae et al., 2010). However in this review we only concentrate on example-based methods.

A decade later in computer graphics, significant progress was achieved by the introduction of a tree structure to accelerate texture generation (Wei and Levoy, 2000). A similar algorithm for geostatistics was proposed in 2002 with Single Normal Equation Simulation (SNESIM), the first computationally-efficient MPS method (Strebelle, 2002). It was an adaptation of trees for categorical variable, allowing fast retrieval of conditional probabilities. Another major innovation brought by SNESIM was the use of multiple-grids to represent structures at different scales, although the related notion of pyramids for continuous variable had been earlier applied in texture synthesis (e.g. Paget and Longstaff, 1998; Popat and Picard, 1993; Wei and Levoy, 2000). Since then, new algorithms have been flourishing in both disciplines. In MPS, the initial methods were pixel-based, meaning that they determined the values of all pixels in an image sequentially. While pixel-based methods are flexible in terms of data conditioning, they are computationally demanding. A second family of methods was later developed, termed pattern-based, which is characterized by generating groups of pixels simultaneously (patches). Patterns borrowed from the training image are assembled
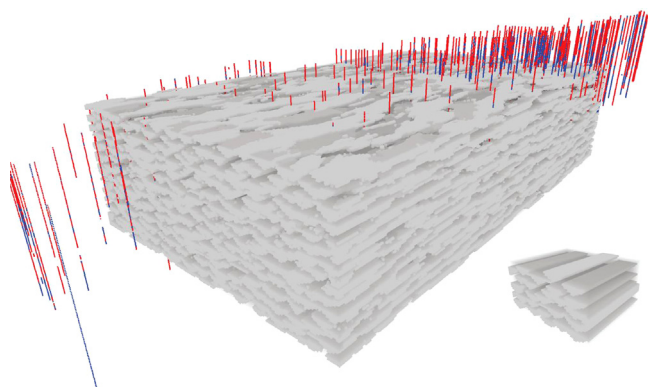

Fig. 1. Typical application of MPS in geosciences. Model size: $340 \times 200 \times 80$ nodes.

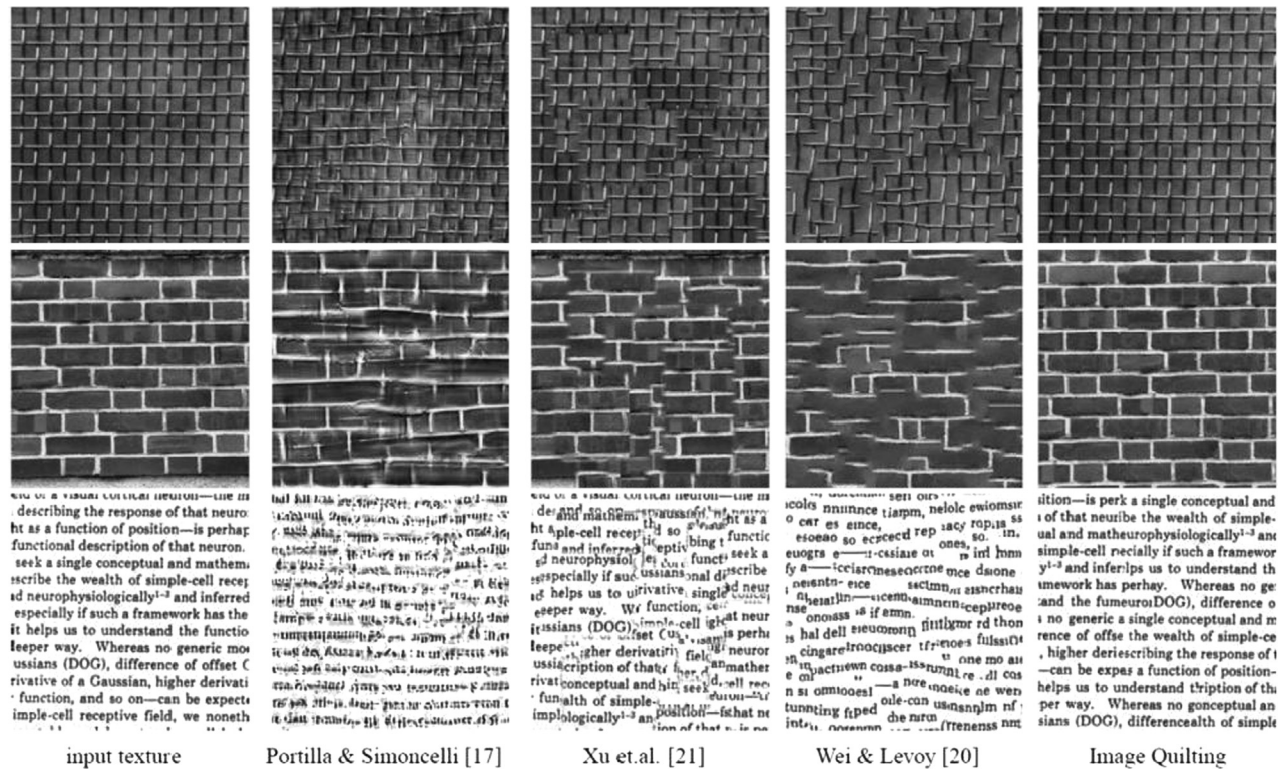| input texture | Portilla & Simoncelli [17] | Xu et.al. [21] | Wei & Levoy [20] | Image Quilting |

Fig. 3. Comparison of early methods for texture synthesis: Portilla and Simoncelli (2000), Xu et al. (2000), Wei and Levoy (2000) and Efros and Freeman (2001) for Image Quilting. Reproduced from Efros and Freeman (2001).

such that they overlap as seamlessly as possible (Arpat and Caers, 2007; El Ouassini et al., 2008; Honarkhah and Caers, 2010; Tahmasebi et al., 2012a; Zhang et al., 2006). At the same time, pixel-based methods continued to be developed (Boucher, 2009; Huysmans and Dassargues, 2011; Mustapha et al., 2011; Straubhaar et al., 2011), such that today these two families of methods coexist. Recently, iterative methods have been introduced for facies-based models (Hansen et al., 2012; Lange et al., 2012; Stien and Kolbjørnsen, 2011; Toftaker and Tjelmeland, 2013), which propose a Bayesian formulation of the constraints illustrated in Fig. 2.

During the same period in computer graphics, a transition was observed from parametric MRF-based texture synthesis (Cross and Jain, 1983; Paget and Longstaff, 1998) to non-parametric methods (de Bonet, 1997) based on exemplars (the computer graphics word for training images), which can be put in perspective with the passage from variogram-based to training image-based geostatistics. Subsequently, as in MPS, pixel-based methods (Efros and Leung, 1999) have been followed by patch-based methods (Efros and Freeman, 2001; Wei and Levoy, 2000). After this point however, both disciplines diverge, with approaches specific to texture synthesis, such as optimization-based (Kwatra et al., 2005) and appearance-space methods (Lefebvre and Hoppe, 2006). Fig. 3 illustrates and compares the results produced by some of the early texture synthesis approaches.

Another family of texture synthesis methods relates to wavelets-based techniques that aim at reproducing the spectral content of textures (Chang and Kuo, 1993; Galerne et al., 2012; Portilla and Simoncelli, 2000). Similar methods also have been later developed in geostatistics (Chatterjee and Dimitrakopoulos, 2012; Gloaguen and Dimitrakopoulos, 2009), which offer another interesting parallel between disciplines. However here we limit ourselves to developments in non-parametric, pixel-based and patch-based methods, and therefore we do not review these methods in detail.

Although both MPS and texture synthesis algorithms are non-parametric approaches incorporating subjectivity – either defined as visual realism or natural variability – they can be formulated and studied based on the MRF theory with a coherent mathematical underpinning, since they can be written as a Gibbs distribution (Daly, 2004; Toftaker and Tjelmeland, 2013; Wu et al., 2000; Zhu et al., 1998). In other words, it is shown that realizations of such processes correspond to an equilibrium state between the different constraints illustrated in Fig. 2, although in practice approximations are made for the sake of computational efficiency (Emery and Lantuéjoul, 2014).

To this day, a number of practical and fundamental issues related to MPS remain open. Computational requirements can still pose problems, especially when using large training images for 3D or 2D-temporal applications. The quality of the resulting models can in several cases be unsatisfactory in terms of reproducing the training image patterns, especially when specific conditioning is used such as soft probabilities (Liu, 2006) or when certain types of structures are considered. The simulation of elongated connected structures in particular is a recurrent problem. Although it received a lot of attention recently (Chugunova and Hu, 2008; de Vries et al., 2009; Honarkhah and Caers, 2012), treatment of some types of non-stationarity remains to be investigated. Another issue is that although some quantitative validation methods have been proposed (Boisvert et al., 2010; de Iaco and Maggio, 2011; Jung and Aigner, 2012), one of the most reliable validation criteria is still visually qualitative: i.e. a human looks at the models and judges how well they represent the training image patterns. Better methods are therefore needed to quantitatively validate the models produced by MPS algorithms.

4. Overview of example-based texture generation methods

Similarly to MPS, the aim of texture synthesis is to obtain a texture that:

- is visually similar to an exemplar (*visual realism*),
- does not contain artificial repetition (*stochasticity*) and
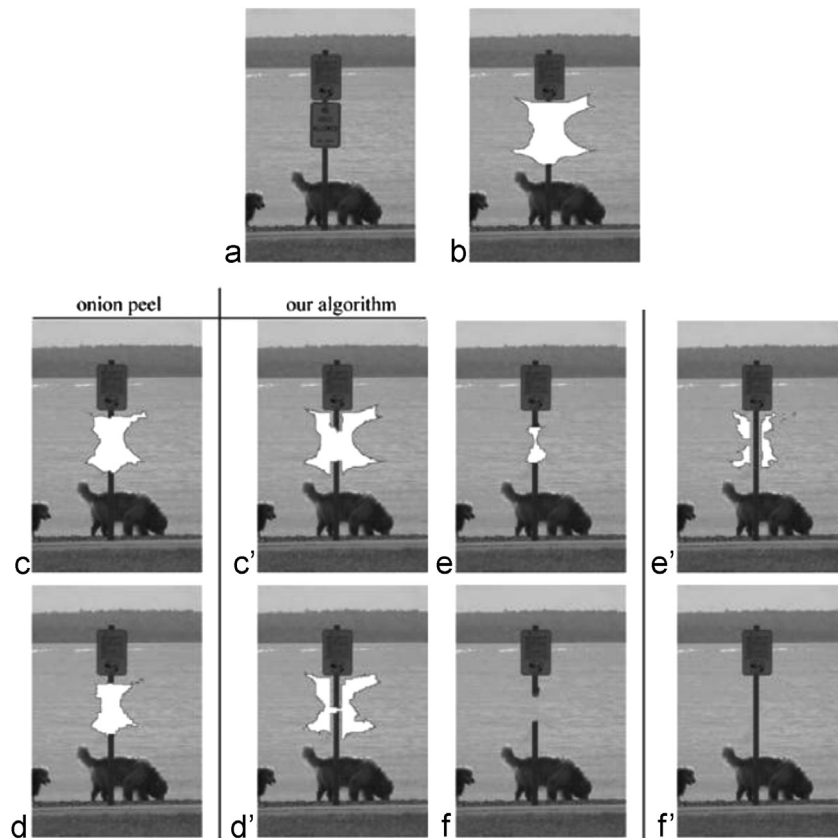- allows for artistic control by imposing local constraints.

Fig. 4. (a) Original image, (b) image with gap and (c)–(f) reconstruction with "onion peel" path. c'–f': Reconstruction with an adaptive path. Figure reproduced from Criminisi et al. (2004).

These points present obvious parallels with the aims of geostatistical simulation exposed in Section 2, although they are formulated differently and addressed in other ways as in geostatistics. In the following sections we point out algorithmic contributions that are specific to texture synthesis.

4.1. Simulation path

In geostatistics, numerous studies have focused on defining a path for the simulation, i.e. the order in which pixels are simulated (Daly, 2004; Liu and Journel, 2004). Unilateral paths offer the possibility of a rigorous dependence with successively simulated values, but with many algorithms it can lead to difficulties in conditioning. Conditioning under unilateral path can be achieved an alteration of the unilateral scheme in order to look ahead for conditioning data (Daly, 2004; Parra and Ortiz, 2011) or by using iterative methods (Kjønsberg and Kolbjørnsen, 2008). Despite this, the random simulation path is still widely used in geostatistics, which is justified by its flexibility to assimilate conditioning data (Fig. 2). The downside of the random path is that coherence between simulated values can be lost due to the use of a limited neighborhood, and it leads to approximations such as the need to drop some neighboring nodes. In texture synthesis, the unilateral path is most often used, and the random path is seen as an oddity, as reflected by Wei and Levoy (2001) who report that: "*Surprisingly, we have found that our algorithm works even if we visit pixels in random order.*"

The problem of path-specific structures has also been considered in computer graphics. In several approaches synthesis is performed in an order-independent manner (Lefebvre and Hoppe, 2005; Paget and Longstaff, 1998; Wei and Levoy, 2003). The idea is to abandon the sequential paradigm, where the value at each pixel (or patch)

depends on all previously simulated values, and replace it with an iterative process. A first simulation, made by randomly choosing pixels from the exemplar, is iteratively improved using neighborhood matching. The important point is that neighborhoods are always read from the previous step image, while new values are written in the next step image. Hence, the computations performed for each pixel are independent: pixels can be processed in any order without changing the result. This has important consequences for parallelization since the approach is to GPU implementations (see Section 7.3).

Texture synthesis also proposed some specific paths, such as the approach of Criminisi et al. (2004) that first simulates the regions that are on the continuation of the major image features. Hence a path is designed that starts from the known areas of an image, from where a front develops and progresses along the locations of major structures. The path is updated during the simulation to maximize the amount of reliable information surrounding the pixel currently simulated. The method is illustrated in Fig. 4. Similarly, Sun et al. (2005) let the user coarsely specify a set of paths along which structured, elongated regions of the image are synthesized first. Such approaches could be an avenue to address the issue of preserving the connectivity of elongated structures, which is a typical problem with MPS.

4.2. Pixel-based methods

Some key algorithmic texture synthesis components are very similar to notions developed in MPS. A first important concept adopted since the early days of computer graphics is the use of pyramids (Popat and Picard, 1993). Pyramids are the texture equivalent of multigrids used in MPS (Strebelle, 2002), but are particularly appropriate for the processing of continuous variables. In order to accelerate texture synthesis, the neighborhoods of

different scales are considered simultaneously. The principle is to first construct a pyramid from the original texture where level 0 of the pyramid is the original texture and level $n+1$ is a filtered, down-sampled version of level n. Reconstruction of fine pyramids from coarser ones can be accomplished by simple correspondence in the training image, since the image is available at the highest resolution pyramid level (Ashikhmin, 2001; Brooks and Dodgson, 2002; Wei and Levoy, 2000). Whereas geostatistical multi-grids are done by subsampling, various types of pyramids have been used in texture synthesis. Examples of pyramids are Gaussian functions, Laplacian pyramids, feature-based pyramids or steerable pyramids (de Bonet, 1997; Heeger and Bergen, 1995); Gaussian pyramid are however most often used.

The earliest algorithms for example-based texture synthesis were contemporary to the first MPS methods. The method of Popat and Picard (1993) is very similar to the one of Guardiano and Srivastava (1993), where a conditional pdf is constructed for each pixel, and a sample is drawn from it to simulate each pixel value. Paget and Longstaff (1998) implement a similar method that incorporates local annealing and also includes multiple-grids or pyramids. In the work by Efros and Leung (1999), the synthesized region is grown from an initial seed by assigning the output pixels one by one in an inside-out, onion layer fashion. All of these methods were however computationally demanding because of an exhaustive search in the exemplar. The major difference is that the method of Guardiano and Srivastava (1993) is designed for categorical variables and does not incorporate multiple-grids. Here the general absence of categorical variables in texture generation methods should be noted, with the notable exception of their use as covariates to improve the synthesis quality. However, most texture synthesis methods for continuous variables can be adapted to the categorical case with very little changes.

The approach of Wei and Levoy (2000) brought some important innovations. It sequentially simulates multiple pyramid levels along a unilateral path. For the first (coarsest) level, it initializes the output as a noise (i.e. randomly copying pixels from the TI to the simulation). A L-shaped template is used. The value of the first pixel is determined by finding the best match in the TI for its neighborhood. Since this is the first simulated pixel, its neighborhood will contain only noise pixels and thus it is essentially randomly copied from the input. As the synthesis progresses, the output neighborhood eventually only contains valid pixels. Noise values are therefore only influencing the synthesis of the initial values. When simulating the next pyramid levels, a complete, square template is added for lower resolutions so that the algorithm takes them into account for simulating the finer grid.

The approach of Wei and Levoy (2000) represents a major advance in terms of performance by introducing tree-based storage, which is possible because only fixed neighborhoods are used. Synthesis times are greatly reduced compared to an exhaustive search of the input texture. Their work pioneered the use of trees in texture synthesis, which have been widely used in further studies as well as in MPS. It should be noted that the Tree-Structured Vector Quantization (TSVQ) tree of Wei and Levoy (2000) is quite different than the one of Strebelle (2002) because it is used for continuous variables. In TSVQ, each node of the tree is a vector whose size corresponds to the number of nodes in the template. This vector is essentially a prototype summarizing all the vectors in the sub-tree under this node. Given a set of training vectors (the patterns in the training image), the procedure for building the tree consists in splitting into two, at each level, the sub-population of input vectors that are below this tree node. This partition is made according to a clustering algorithm (Lloyd, 1982). Searching the tree is very efficient, because in the context of texture synthesis one is only interested in finding a single best matching event (as opposed to geostatistics where a conditional

distribution is typically developed). At each level of the tree, the input vector is compared with the prototypes corresponding to both branches of the tree. The best matching prototype is systematically chosen until the leaf level of the tree is reached, returning the best matching data event.

4.3. Patch-based methods

The idea of using patches instead of pixels has been used early in texture synthesis, and gave rise to some of the fastest algorithms used nowadays (Lasram and Lefebvre, 2012). One of the early methods is the Chaos Mosaic (Xu et al., 2000) that randomly places patches in the target image, and then uses a smoothing filter to take care of the transitions between patches. This method is very fast but leads to visible artifacts. Liang et al. (2001) extend the pixel-based method of Wei and Levoy (2000) to patch-based simulation. The patches of the input texture are first stored in a tree structure, the simulation then proceeds by assembling patches using a unilateral path. Each patch shares an overlap region with the previously simulated areas. For the simulation of a patch, the tree is searched for all patches compatible with the overlap area, resulting in a pool of candidate patches, one of which is randomly selected for simulation. A final blending step makes sure that no effects appear at the patches junction. This method is relatively close to the one of Arpat and Caers (2007) in the context of MPS or the one of Tahmasebi and Sahimi (2013) that does not include the blending step. The implementation of Liang et al. (2001) is very efficient, with examples simulations of size 256×256 pixels generated in about 0.02 s on computers that would today be considered outdated. This performance is explained by several improvements. One of them is the use of a quadtree pyramid, which is a data structure in the tree that accounts for the multiresolution character of the problem. In the search tree, the patterns are stored in a multiresolution form. Therefore, the results of a search performed for a coarse resolution can be re-used at a finer resolution, which accelerates the simulation of the finer grids. Another improvement is the use of principal component analysis to reduce the dimensionality of the patches. A more detailed review of the acceleration strategies used in texture synthesis is carried on in Section 7 of this paper.

Image quilting (Efros and Freeman, 2001) is another elegant patch-based method. It tiles the domain with overlapping patches whose overlap regions agree as much as possible. Then it cuts them optimally such that they assemble with minimal overlap error. This minimum error boundary cut is computed using dynamic programming methods such as Dijkstra's algorithm (Dijkstra, 1959). The methodology is described in Fig. 5. The approach was then generalized by the graph cuts method of Kwatra et al. (2003) to accommodate arbitrary patch shapes that can overlap each other with much more flexibility than in Image Quilting. Kwatra et al. (2003) in fact solve a per-pixel labelling problem using alpha-expansion (iterated graph-cuts), making an interesting connection between pixel-based and patch-based approaches. One advantage of graph cuts is the possibility to generalize it to 3D (or nD) cases. For example it has been successfully applied for video textures, which are 2D-temporal (Wei et al., 2009). The flexibility of the graph cuts approach and the principle of reshaping the patches bears a large potential for conditioning simulations. The fact that patches can be updated locally means that it is possible to adapt the method to all types of conditioning constraints found in earth science applications. Although approximate because only locally accommodating conditioning data, the idea to change the shape of patches for conditioning is promising and goes in the direction of some recent developments in patch-based MPS methods (Parra and Ortiz, 2011; Tahmasebi et al., 2012a). Several patch-based texture synthesis approaches further deform the content of the patches to improve the alignment of
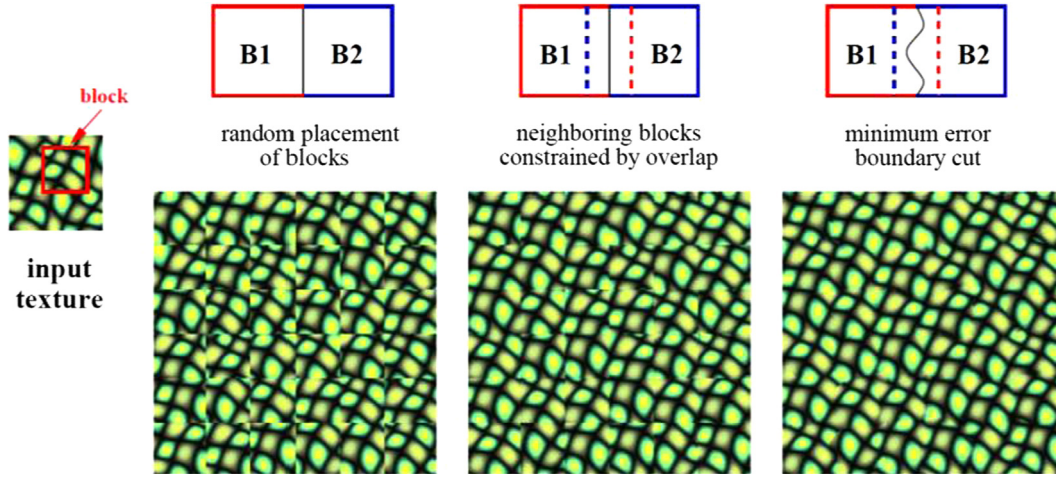
Fig. 5. Illustration of the image quilting algorithm. Three levels of increasing algorithmic complexity are represented, image quilting being the last one involving cutting the overlap regions such that the overlap error is minimized. Reproduced from Efros and Freeman (2001).

structures crossing their boundaries, e.g. the mesh quilting of Zhao et al. (2012) or the approach of Lasram and Lefebvre (2012). This seems especially well suited for MPS methods where elongated structures are considered. The approach of Darabi et al. (2012) extends the set of transformations to both geometric and photo-metric adaptation of the patches, allowing structure preserving interpolation between very different patterns.

4.4. Optimization-based methods

Optimization-based methods are in a sense similar to geosta-tistical simulations using simulated annealing (Deutsch and Journel, 1991; Peredo and Ortiz, 2011) or iterative methods (Lange et al., 2012; Stien and Kolbjørnsen, 2011) because they consider the domain as a whole and not pixel by pixel, and formulate the problem with an objective or likelihood function. As indicated by their name, these optimization-based methods do not perform sampling of a posterior distribution, conversely to the approaches used in geostatistics. Despite this, optimization-based methods are still interesting because they seem to offer large advantages in terms of patterns reproduction as well as computa-tional performance. In addition other goals can be conveniently integrated in the objective function. This has been demonstrated for generating temporally coherent animation of stationary patterns.

The premise of optimization-based methods is that the goal of texture synthesis is to honor a set of constraints that can be to a certain extent contradictory (Kwatra et al., 2005). On the one hand, one wants to produce a field that has patterns similar to the ones present in the training image, and this is done by condition-ing each pixel to its neighborhood. On the other hand, the different neighborhoods must be coherent with each other by overlapping in a seamless manner. The inconsistency in the different con-straints is expressed in terms of an energy measure. The energy of a single synthesized neighborhood is defined as its distance to the most similar neighborhood in the training image. The total energy of the synthesized texture is a global metric consisting of the sum of energies of individual neighborhoods. In the formulation of Kwatra et al. (2005), the texture energy at a given iteration t is defined as

$$E_t(\mathbf{x}; \{\mathbf{z}_p\}) = \sum_{p \in \mathbf{x}^\circ} \| \mathbf{x}_p - \mathbf{z}_p \|^2, \qquad (1)$$

where $\mathbf{x}$ is the simulated texture, $\{\mathbf{z}_p\}$ is the set of all patterns in the training image, $\mathbf{x}_p$ corresponds to the values of a pattern in $\mathbf{x}$ that is centered on pixel p, and $\mathbf{z}_p$ is the neighborhood in the

training image that is the most similar to $\mathbf{x}_p$ under the Euclidean norm. The energy is computed based only on a subset of all neighborhoods, $\mathbf{x}^\circ \subset \mathbf{x}$, chosen such that they are close enough to overlap. Note that prior to computing the energy, all best matching neighborhoods $\mathbf{z}_p$ corresponding to each output pixel in $\mathbf{x}^\circ$, need to be identified.

The energy function is then minimized using an algorithm similar to Expectation-Maximization (EM), consisting of two steps that are iterated. The algorithm starts with an initial guess of the simulation which can be noise or a tiling of the training image. Then, the M step (or search phase) consists in finding, for a set of control locations on the domain, the corresponding best matching neighborhoods in the training image. This step can be accelerated by using a tree search or other acceleration methods (see Section 7). The best matching locations are identified, but no pixel is updated in the simulation at this stage. Next, the E step (or optimization phase) aims at minimiz-ing the entire energy of the synthesized image. The values of the simulated variable are updated on the entire domain such that the total energy is minimized. With unconditional texture synthesis, this optimization is accomplished by simply averaging the values of the best matching neighborhoods overlapping each pixel. In case of additional constraints, the energy function has additional terms and the updated simulation values need to be determined by solving a least-square problem minimizing E_t with respect to $\mathbf{x}$. This is done by setting the derivative of the energy function with respect to $\mathbf{x}$ to zero, which yields a linear system of equations that can be solved for $\mathbf{x}$.

With these updated values, the following search phase will find new best matching neighborhoods in the simulation, which will necessitate new adaptations in another E step, etc. The over-smooth or blur produced at the E step will cause the next iteration of the M step to replace the neighborhoods corresponding to this region with ones that are more consistent with each other. This method is relatively computationally demanding, with reported times in the order of minutes for a 256×256 image. However, in more recent works, Huang et al. (2007) report very fast computa-tional times with an improved implementation of optimization-based simulation.

We note here that optimization-based texture synthesis has only been used with continuous variables. To be used in geosciences, where the use of categorical variables is frequent, the method would need to be adapted by applying two modifications:

(1) The formulation of the energy function would need to be changed by considering the proportion of mismatching pixels between neighborhoods $\mathbf{x}_p$ and $\mathbf{z}_p$.

(2) If more than two categories are considered, the weighted average of patterns used in the E step would not be applicable directly because of the absence of order relationships. In those cases it could be replaced by another method able to combine categorical patterns, for example using a patch cutting approach (Efros and Freeman, 2001).

5. Integration of localized data

5.1. Conditioning in geostatistics

One difference between MPS and texture synthesis is that geostatistical models need to integrate measured data. While some global information (type of patterns, histogram) is given by the training image, this information needs to be coherent with the measured data. Moreover, the data provide local information, i.e. the models strictly need to locally reproduce the measured values
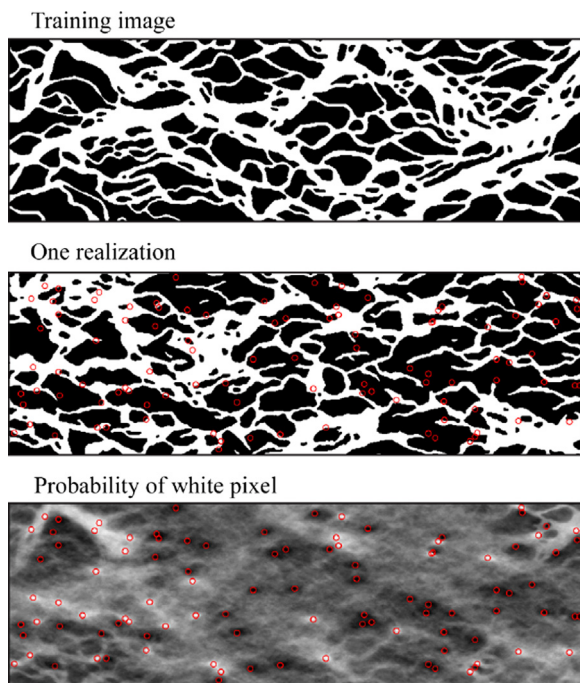

Fig. 6. The non-local effect of conditioning data. The training image comes from a satellite image of the Ganges delta, Bangladesh. Data locations are depicted by red circles. Probability of white pixels is based on 100 MPS realizations. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

at the measured locations. Such conditioning is critical because the data are often acquired at great cost and therefore the information needs to be optimally exploited. The most general way of proceeding is to adopt a mathematical model of conditioning with a Bayesian formulation where the training image defines a prior ensemble of models, and a likelihood function defines the relationship to data (Hansen et al., 2012; Toftaker and Tjelmeland, 2013). Hence to achieve conditioning in geostatistics, it is not enough that the point data is reproduced at the data location.

Depending on the structures present, the effect of a local datum can extend beyond the local level and constrain the occurrence of spatial patterns far from the datum location. Fig. 6 shows an example of this, where 100 data points are used with a training image depicting elongated and connected categorical values. Every single realization locally matches the data values, but the influence of the conditioning extends beyond the data location, which is visible on the probability of having a white pixel.

While local information can also be included in texture algorithms, the corresponding constraints are often not defined in such a formal way. For example an artist "paints" an image with a given texture, and then subjectively evaluates the result. In this context a fast interactive process is more desirable than a formal theory of conditioning. This is a significant difference between the fields of texture synthesis and geostatistics.

However, although mathematically rigorous, the Bayesian approach to conditioning is often computationally impractical. Many MPS algorithms are not analytically tractable and therefore they require expensive Monte-Carlo methods. As a consequence, MPS models are often conditioned using faster approximate methods, such that they reasonably correspond to the models that would have been obtained from a Bayesian approach. A practical approach often used for hard data is to condition models by construction. It can be seen as building models around the conditioning data, using them as a starting point and then building the rest of the model such that it embeds them. The result of these practical approaches is then relatively similar to what is done in texture synthesis (see next section), and in this sense we expect that bridges between disciplines can be built on these bases.

5.2. Hard constraints with optimization-based texture synthesis methods

Adding conditioning constraints can be easily formulated in the context of optimization-based texture synthesis. This is accomplished by adding terms to the energy function, such as for example the soft constraints (in the form of a flow field) shown in Kwatra et al. (2005). Kopf et al. (2007) use this formulation to

Fig. 7. Imposing local conditioning inside a 3D texture. The small images on the left represent the local constraints (top) and the associated weights (bottom). The resulting 3D texture is only affected locally, as seen on the horizontal cross-section (block in the middle). The rest of the domain remains stochastic, with no artifacts related to conditioning on the vertical cross-section (block on the right). The training image (not shown here) is 2D. Figure reproduced from Kopf et al. (2007).

impose local constraints, which are, in geostatistical terms, hard and soft conditioning data. An example of conditioning, in the context of 3D textures, is shown in Fig. 7. At the same time as local conditioning, an additional term is added in the energy function that considers the mismatch between the histogram of the simulation and the histogram of the training image. The main advantage is that even if the training image is mildly non-stationary, all areas of the training image have to be represented in the simulation. The authors note that corollary benefits of histogram constraints include an improvement in the overall patterns reproduction and an acceleration of the algorithm. An interesting application in geoscience would be to use this formulation to impose local probabilities or equivalent parameters for a certain volume (e.g. equivalent permeability).

5.3. Image completion/Gap-filling

Image completion is a special case of conditional simulation where the number of conditioning data is very large, typically larger than the area to be simulated. Training image-based gap-filling relies on using the patterns composing the known areas of the domain for completing the unknown parts. The use of texture synthesis to fill gaps in images has been illustrated since some of the earliest studies (Efros and Leung, 1999; Liang et al., 2001; Wei and Levoy, 2000), and here we only give a brief overview of the existing methods. In the context of MPS, the same principle has been applied to subsurface problem and with multivariate remote sensing data (e.g. Wu et al., 2008). It is shown in these studies that the path chosen for reconstruction has a strong influence on the final results. Another question left open is to decide whether the reconstructed pixels can be re-used as input or if the pattern match search should be limited to the original conditioning data only.

Drori et al. (2003) address both these issues. They achieve excellent reconstructions by using an iterative scheme where the unknown parts are initially filled by smooth interpolation from their boundaries. The known parts of the image are then used as a source of patterns for completing the unknown parts of the image based on neighborhood matching. The parts of the image that are originally known are attributed a high level of confidence, whereas the reconstructed areas have a level of confidence that is a function of the goodness of the neighborhood match. In subsequent iterations, the previously simulated areas are also used in the pattern search (providing a sort of recycling), but the pattern search takes account for the confidence terms that are inherited from the previous iteration. The logic is that at each iteration, the algorithm knows which parts of the original image should provide patterns (high confidence areas) to populate which other parts of the image (low confidence areas). As a consequence, the areas of low confidence become progressively smaller until all gaps are filled and the entire domain has a high confidence level. The

results after several iterations highly resemble the complete image from which the original sparse data were extracted. A drawback of the method is the computation time compared to other texture synthesis methods, the authors reporting the order of a minute for a 384×256 image.

Another significant improvement brought by Drori et al. (2003) is the use of adaptive neighborhoods, meaning that the size of the pattern searched for is inversely proportional to the frequency content of the image region being completed. The rationale is that large patches are likely to match well only in regions of smoothly varying content, while completing the frontier between two highly contrasted regions requires smaller patches. To determine if an image area is of low frequency in the exemplar, a quick analysis is carried based on low-order statistics (in this case the luminance histogram). Since a common problem with gap-filling is the reduced amount of patterns available, transformations are applied to the target patterns to increase the search space. The authors use what they term a search in different scales, under rotations and under reflections. The results show that such transformations allow generating new patterns that are not initially present in the original dataset, but still spatially coherent with the original training image.

5.4. Soft constraints: feature maps and non-stationarity modeling

Since most natural phenomena are spatially non-stationary, the ability to deal with non-stationarity is a key requirement for most applications of geostatistics. Similar needs exist in texture synthesis, where non-stationary is termed globally varying texture. Methods have been developed that instead of allowing each input pixel to go to any place in the output texture, a mechanism is used to control where each pattern can go (Fig. 8). The globally varying distribution of patterns is often conditioned by certain natural factors, such as patterns varying gradually from one side of the image to the other. When such a factor is explicitly present, either by measurement or by deduction, it is expressed as a control map for the corresponding globally varying texture. Recent developments in multiple-point geostatistics have addressed non-stationarity in a similar way (Chugunova and Hu, 2008). A similar concept has been implemented with different algorithms (Honarkhah and Caers, 2012; Jha et al., 2013b; Meerschman et al., 2014; Straubhaar et al., 2011).

A key concept in texture synthesis is that the feature map is seen as a filtered version of the primary variable. The feature map can be imposed, forcing certain types of structures to occur at specific locations (Efros and Freeman, 2001). Hertzmann et al. (2001) develop this idea as a framework named "image analogies" and define a notation that clearly states the problem: we wish to generate B′ to satisfy the relation
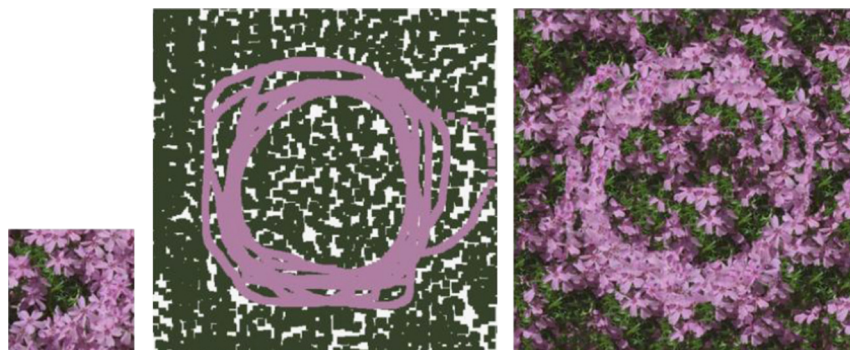
$$A : A' :: B : B' \qquad\qquad (2)$$

Fig. 8. User-guided texture synthesis. Left: exemplar, middle: user-defined control map; right: output. Figure reproduced from Ashikhmin (2001).

meaning that the relationship between two objects A and A′ is the same as the relation between two other objects B and B′. The user provides the algorithm with three inputs: the training image A, the filtered training image (or auxiliary variable of the training image) A′, and the new image to be filtered B. The result is then a simulation B′ which relates to B in the same way A′ relates to A. The principle of image analogies has since been used to impose special effects to pictures. A pair of training images is given, which correspond to (a) a photograph and (b) a graphical effect applied on this photograph (Fig. 9). Another photograph is given and the algorithm outputs an image of the graphical effect applied to the new photograph. The graphical filter is a transfer function that is borrowed from the training set and imposed on the target image. Graphical examples are available in Hertzmann et al. (2001) and Welsh et al. (2002). Lu et al. (2007) extend filters to include patterns induced by physical processes such as rust, cracking or dirt accumulation on 3D objects. In a first step, a real object is registered in 3D. An ageing process is induced, and temporal sequences of texture ageing are digitally captured. In general, ageing displays systematic variations that depend on physical parameters: for example rust is more likely to form at places that are more exposed. These relationships between the object surface and the occurrence of ageing are then used in an image analogies framework. The main advantage of these temporal filters is that they relate the object shape and the physical process originating the texture. Such an approach is similar to statistical modeling, but also to process-based modeling, and therefore could be used to build geological models with improved physical realism.

For applications in geostatistics, the concept of by-example filters could be turned around: the user then provides as inputs the training images A and A′ and the filtered output B′. Image analogies can then be used to find B, which corresponds to the underlying truth. For example, B′ can be an auxiliary variable such as a geophysical survey. Based on the relationship B:B′, it is possible to determine B, the underlying geological setting. This approach has potential applications in domains that involve the identification of a variable from indirect measurements, such as geophysics or remote sensing.

An alternative approach is to simulate the feature map jointly with the primary variable (rather than imposing it) to improve the results by enforcing coherence in the simulated structures (Kopf et al., 2007; Wu and Yu, 2004). Zhang et al. (2003) use what they call a texton mask that is a binary segmented version of the training image. The simulation method is then based on a distance having two terms, one for the continuous and one for the categorical segmented variable that is used as a guide. Such texton masks improve the continuity of the simulated features through the connectivity properties of the categorical variable. This simple method could be implemented in a straightforward manner with existing MPS codes to ensure better results. For the simulation of continuous fields, it would involve implementing texton maps in

an automated way, based on a thresholding or segmentation of the training image.

The approach of appearance-space texture synthesis (Lefebvre and Hoppe, 2006) extend the idea of texton maps by including a range of different feature maps from the exemplar, describing specific characteristics such as objects boundaries, distance to these boundaries, radiance, etc., which are called appearance vectors. These variables can often characterize a pixel better than its neighborhood in the variable of interest. Therefore, the neighborhoods can be substantially reduced (with as little as 4 neighbors) and the quality of the results is still preserved, offering significant computational gains. The authors use PCA to further reduce the dimensionality of the appearance vectors, and with a GPU implementation obtain computation gains of 3–4 orders of magnitude (see Section 7.2 for more details). Fig. 10 shows examples of this approach, where simulations done with only the training image (column "no feature distance") present artifacts, whereas these artifacts disappear when using feature maps consisting of (a) a binary variable describing the occurrence of certain objects in the training image and (b) a continuous variable consisting of the distance to these objects in the training image. These results suggest that the principle of such feature distances should be used whenever possible in MPS algorithms.

Another use of texton maps is to create a smooth transition from one texture to another. The principle is that although it is difficult to combine complex textures made of color patterns, it is much easier to combine binary images by using some type of averaging and smoothing. Associating a texton map to each training image, and blending the textons only, allows producing smooth and artifacts-free transitions from one texture to another (Zhang et al., 2003).

6. 3D textures

While computer graphics mostly deal with 2D textures mapped along surfaces, there is a number of situations where 3D solid textures are desirable, for instance to produce the illusion that an object has been carved out of a solid block of material. Since most often only 2D exemplars are available, the problem of synthesizing 3D images from 2D training images is just as acute in texture synthesis as in geological modeling.

The problem of simulating 3D volume from 2D training images has first been addressed by Wei (2002). At each pixel of the output volume, three interleaved 2D neighborhoods are extracted. The best matches are found independently in each of the three exemplars, corresponding to the three directions of space. The value of the voxel is then updated as the average of the three neighborhood center colors. This process is repeated at each voxel, and several iterations are performed. In geostatistics, Comunian et al. (2012) propose a similar scheme, formulated in a probabilistic framework and within the context of sequential simulation.

Fig. 9. The principle of filtering using image analogies (Hertzmann et al., 2001). A and A′ provide an example of a filtering relationship. Given a new image B, one can generate a filtered version of it using A:A′ as an example relationship. The inverse operation is also possible: given a filtered information B′, one can generate the original unfiltered image B.
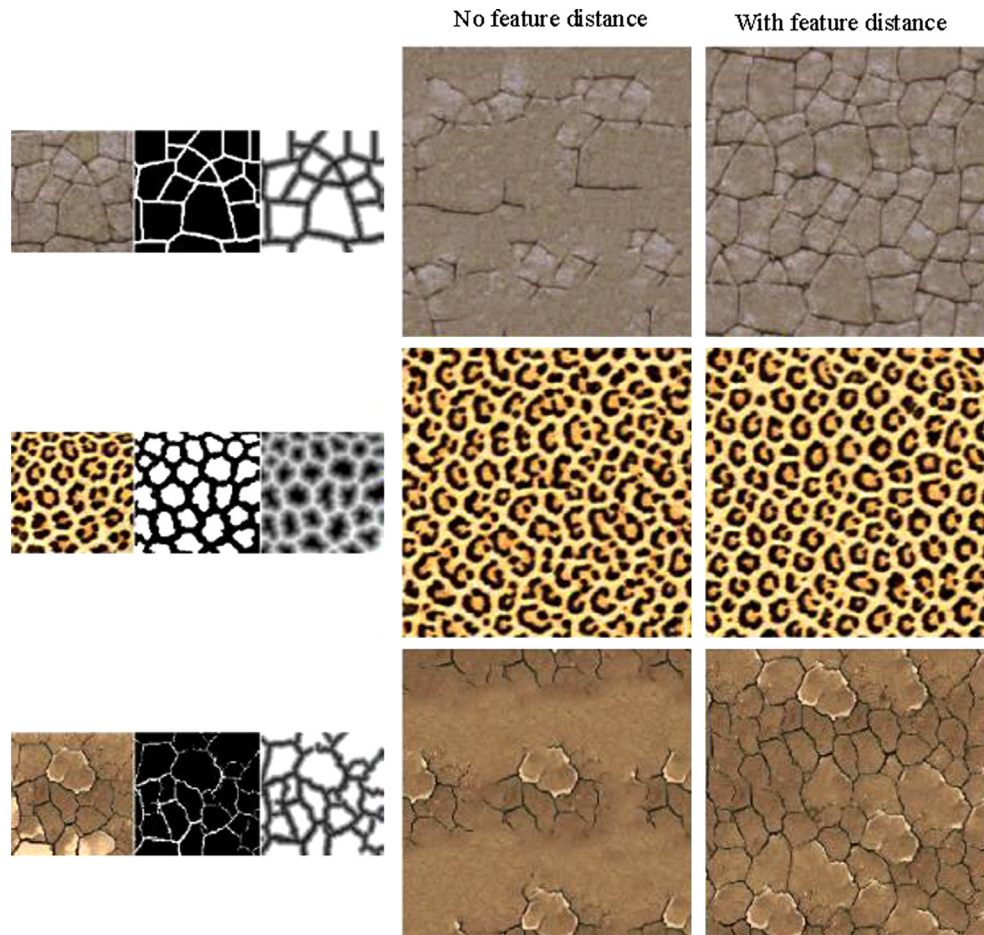
Fig. 10. Appearance space using feature maps. Left: exemplars and corresponding appearance vectors signaling the presence of specific features and the distance to the edge of such features; right: output textures without and with use of appearance vectors. Figure reproduced from Lefebvre and Hoppe (2006).

The approach consists in computing a conditional distribution of the pixel to simulate for each direction, and then to combine these distributions using a probability aggregation approach (Allard et al., 2012; Mariethoz et al., 2009), which is more rigorously defined than the averaging method of Wei (2002). Comunian et al. (2012) and Hajizadeh et al. (2011) also propose to simulate a 3D volume by successively generating 2D sections that are coherent with each other.

In texture synthesis, Kopf et al. (2007) generate 3D volumes from 2D training images by using a modified optimization-based synthesis algorithm. The optimization is formulated such that every slice through each node in the volume has to be similar to the 2D training image, and all neighborhoods need to agree with each other. A distance between several orthogonal 2D neighborhoods is simultaneously minimized. Different training images can be used for each direction, as shown in Fig. 11. The implementation is computationally efficient and allows imposing conditioning data.

Other applications of 3D textures include Manke and Wunsche (2009) who extend to 3D the method of Lefebvre and Hoppe (2005) and report similar results as Kopf et al. (2007) with smaller computation times. Dong et al. (2008) present an implementation of 3D solid texturing that can provide real-time 3D solid texture by synthesizing at only a few selected locations, from which it is possible to quickly query the model for finer details. It can be compared to a case where one would only simulate the coarsest multigrid, and populate the fine grids on demand. Wang et al. (2010) apply the principle of image analogies in 3D to automatically segment 3D images based on a 2D segmentation, as shown in Fig. 12.

7. Acceleration

Computer graphics applications often require generating textures at very high output rates of more than a million pixels a second. It can be the case for example in applications where texturized 3D objects need to be generated as the user explores the scene or need to be fractured in real-time, revealing inner structures. To this end, several strategies have been developed aimed at improving computational efficiency. These acceleration methods can be combined, bringing drastic improvement in simulation times, with synthesis speeds of up to 15.5 million pixels per second (Han et al., 2008). In contrast, recent multiple-point algorithms report computation times orders of magnitude slower, with commercial implementations reporting about 5000 nodes per second (Straubhaar et al., 2013; Zhang et al., 2012).

7.1. k-Coherence

The search tree approach (Strebelle, 2002; Wei and Levoy, 2000) resulted in important acceleration of the simulation algorithms, but more effective methods have been developed since, in particular those based on the notion of coherence (Ashikhmin, 2001), implying that values which are next to each other in the input will tend to remain together in the output. The k-coherence algorithm (Tong et al., 2002) is similar to the tree storage in the sense that it involves a preprocessing step that classifies the information of the training image in such a way that patterns can be retrieved fast. However, whereas the tree stores complete pattern information, k-coherence only stores similarities between
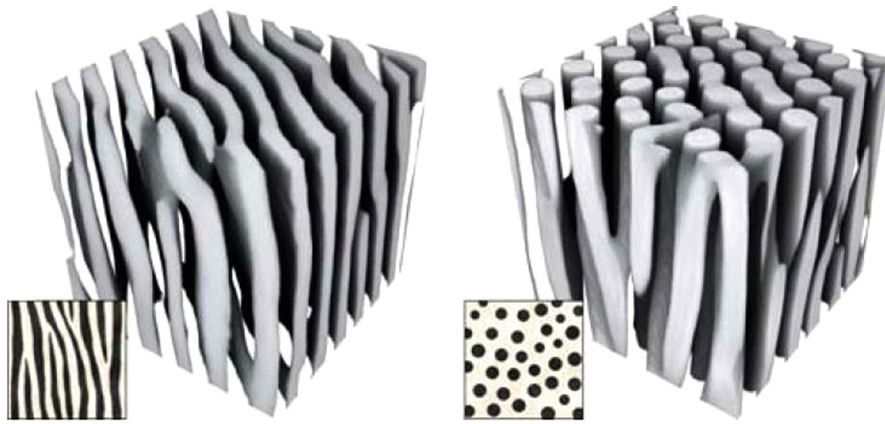
Fig. 11. Two examples of 3D texture synthesis, with the corresponding 2D exemplars shown in inserts. Figure reproduced from Kopf et al. (2007).

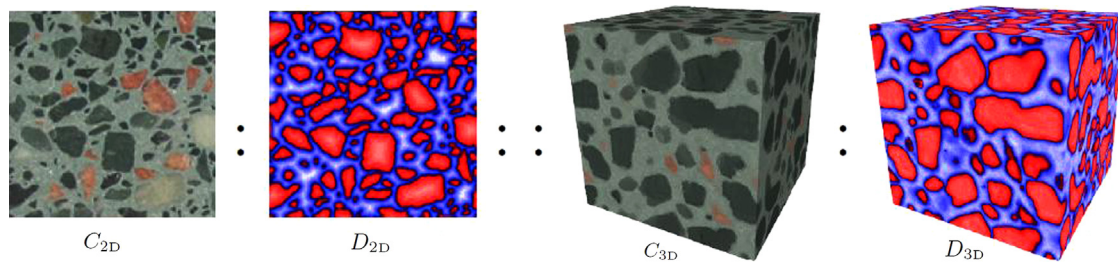C_{2D} D_{2D} C_{3D} D_{3D}

Fig. 12. Segmentation of 3D images seen as a filtering problem. Figure reproduced from Wang et al. (2010). The notations of Hertzmann et al. (2001) are used.

given neighborhoods. During this preliminary analysis, for each TI node a similarity-set is built that contains a list of other TI nodes with similar neighborhoods. During synthesis, the algorithm copies pixels values from the input to the output, but in addition to the value, the source pixel location is also copied. To synthesize a particular output pixel, the algorithm builds a candidate set by taking the ensemble of all similarity-sets of the neighborhood, and then searches through this limited candidate-set to find the best match. The size of the similarity-set, k, is a user-controllable parameter, usually between 2 and 11, that determines the overall speed/quality trade-off. It is noted that k-coherence provides a constant time complexity, as opposed to a logarithmic time complexity with tree-based algorithms. Another advantage is that the k-coherence search is very easy to parallelize.

Related to k-coherence, jump maps (Zelinka and Garland, 2002) store for each input pixel a set of matching training image pixels (jumps). Texture synthesis proceeds as a random walk through the jump map. Another similar acceleration method is the approximate nearest-neighbor algorithm of Barnes et al. (2009). It can be seen as a randomized version of k-coherence. Instead of precomputing sets of best matching neighborhoods, it relies on a random search during synthesis. It alternates between selecting candidates at random, and selecting candidates using coherence from neighboring pixels. This effectively propagates good matches found by the random search, hence quickly forming coherent patches. This scheme is both simple to implement and very effective.

7.2. PCA

Dimensionality reduction of the patterns used for simulation has been recently investigated in the context of MPS (Abdollahifard and Faez, 2012; Honarkhah and Caers, 2010). This avenue has also been explored in texture synthesis, the most frequently used tool in this context being Principal Component Analysis (PCA) applied on the training image patterns. In the transformed space, the same patterns can be represented using smaller vectors, therefore resulting in improvements in both computational time and memory use (Lefebvre and Hoppe, 2005; Liang et al., 2001). Hertzmann et al. (2001) report that keeping 99% of the variance can lead to a reduction of the patterns storage by an order of magnitude. However, they also note that PCA can degrade the quality of the simulated patterns, especially when the training image is made of simple patterns, and therefore further simplification through PCA erases the small amount of information present. For complex cases where memory and CPU time are critical, significant gains are reported. An important aspect is that there is no need to back-transform the synthesized texture to a high-dimensional space because dimensionality reduction methods are applied in conjunction with coherence techniques, hence for each simulated value the original location in the training image is known. Using this location information allows mapping the synthesized high-dimensional variable by simple look-up in the training image.

7.3. GPU

Although Graphical Processing Units (GPU) implementations of geostatistical methods have started to appear recently (Huang et al., 2013a, 2013b; Tahmasebi et al., 2012b), the algorithms developed in the texture synthesis community have already been implemented on GPU for a number of years. In traditional geostatistics, the use of parallelization is hampered by the sequential nature of MPS algorithms. Each pixel depends on each previously simulated pixel and therefore they cannot be generated simultaneously. In contrast, iterative methods (i.e. order-independent simulation) are more amenable to parallel implementations.

One of the most efficient texture synthesis methods is the parallel texture synthesis algorithm of Lefebvre and Hoppe (2005), whose guiding principle is to manipulate pixel coordinates rather than values. The resulting simulations can be represented as a map corresponding to the location in the training image of the

simulated values (Zelinka and Garland, 2002). The simulation algorithm is based on manipulating coordinates, with three main steps:

1. The Upsampling step increases the resolution of the previous pyramid level. Since the algorithm manipulates coordinates, this is done trivially through simple coordinate inheritance instead of neighborhood matching.
2. The Jitter step introduces variability explicitly in the result, adding an offset to pixel coordinates during the multi-resolution synthesis process. This visually displaces blocks of texture in the result. In absence of Jitter a simple tiling of the exemplar is produced.
3. The last step is the correction step done by a few iterations of order-independent neighborhood matching. The search space is kept small by using k-coherence candidates.

Each of these operations is amenable to GPU implementation and the scheme has been extended to 3D models (Dong et al., 2008). A corollary observation is that the coordinate map of a simulation directly reflects its variability, with large patches of identical colors reflecting zones of low variability with respect to the training image. Such coordinates maps could be used in geostatistical applications as a validation tool indicator of the variability between simulations.

8. Conclusion

The fields of MPS and texture synthesis have evolved separately during the last decade, and there is a large innovation potential in bringing together the advances made in both domains. This study has identified areas in both geostatistics and texture synthesis that can contribute to improve methods and applications in the other discipline. Some important areas for future research in that have been identified include:

- Optimization-based simulation methods (Kopf et al., 2007; Kwatra et al., 2005) for hard and soft conditioning, and also to constrain models to equivalent parameters such as hydraulic conductivity over a given volume;
- The integration of texton maps (auxiliary variables) for the simulation of continuous fields (Zhang et al., 2003);
- Faster search in the training image using k-coherence (Barnes et al., 2009; Busto et al., 2010; Tong et al., 2002) seems to combine good computational efficiency with low memory needs;
- Accelerated implementations using GPU (Lefebvre and Hoppe, 2005);
- Applying PCA and appearance vectors for reduction of the training image data events (Lefebvre and Hoppe, 2006);
- Adaptive simulation path to improve the connectivity properties of MPS simulations (Criminisi et al., 2004);
- Conditioning to sparse data in texture synthesis and statistical coherence between data and exemplar (Toftaker and Tjelmeland, 2013).
- Developing new types of constraints for texture models such as connectivity (Renard et al., 2011) or physical rules through inverse problems (Hansen et al., 2012).

Traditionally, texture generation studies have been seen as a 2D problem. Moreover, they have not concentrated much on conditioning, for the simple reason that conditioning – especially to sparse hard data – is not needed in most graphical applications. Therefore it has often been stated that texture synthesis methods are not applicable in geostatistics. The emergence of solid (or 3D)

texture synthesis may change these preconceptions. This study has shown that the generation of 3D stochastic models is possible and efficient with texture synthesis methods, and that the results can compete with the latest MPS algorithms in terms of patterns quality and speed.

It is shown that a number of texture synthesis methods do allow for local conditioning, however statistical consistency between data and model is not investigated. Most texture generation algorithms use unilateral paths that have traditionally presented problems for conditioning. However, adapting these methods for random paths does not seem to present major difficulties. Even in cases where this may prove difficult, it has been showed that conditioning is possible with unilateral paths (Parra and Ortiz, 2011). In addition, soft conditioning is often present in texture applications, from the user-constrained textures of Ashikhmin (2001) to the image analogies of Hertzmann et al. (2001). It was also showed that optimization-based methods can simultaneously integrate both soft and hard conditioning with 3D textures Kopf et al. (2007).

References

Abdollahifard, M.J., Faez, K., 2012. Stochastic simulation of patterns using Bayesian pattern modeling. Comput. Geosci., 1–18

Allard, D., Comunian, A., Renard, P., 2012. Probability aggregation methods in geoscience. Math. Geosci. 44 (5), 545–581.

Arpat, B., Caers, J., 2007. Conditional simulations with patterns. Math. Geol. 39 (2), 177–203.

Ashikhmin, M., 2001, Synthesizing natural textures. Paper presented at Proceedings of the ACM Symposium on Interactive 3D Graphics.

Atkinson, P., Pardo-Igúzquiza, E., Chica-Olmo, M., 2008. Downscaling cokriging for super-resolution mapping of continua in remotely sensed images. IEEE Trans. Geosci. Rem. Sens. 46 (2), 573–580.

Bárdossy, A., Li, J., 2008. Geostatistical interpolation using copulas. Water Resour. Res. 44(W07412).

Barnes, C., Shechtman, E., Finkelstein, A., Goldman, D.B., 2009. PatchMatch: a randomized correspondence algorithm for structural image editing. ACM Trans. Gr. 28 (3).

Besag, J., York, J., Mollié, A., 1991. Bayesian image restoration, with two applications in spatial statistics. Ann. Inst. Stat. Math. 43 (1), 1–20.

Boisvert, J.B., Pyrcz, M.J., Deutsch, C.V., 2010. Multiple point metrics to assess categorical variable models. Nat. Resour. Res. 19 (3), 165–175.

Boucher, A., 2009. Considering complex training images with search tree partitioning. Comput. Geosci. 35 (6), 1151–1158.

Brooks, S. and N. Dodgson (2002). Self-similarity based texture editing. ACM transactions on Graphics 21 (3), 653-656.

Busto, P., Eisenacher, C., Lefebvre, S., Stamminger, M., 2010. Instant texture synthesis by numbers. Vis. Model. Vis. 2010, 81–85.

Caers, J. 2005. Petroleum Geostatistics. Society of Petroleum Engineers. publication place: Richardson.

Caers, J., 2011. Modeling Uncertainty in the Earth Sciences. Wiley, New Jersey.

Chang, T., Kuo, C.C. J., 1993. Texture analysis and classification with tree-structured wavelet transform. IEEE Trans. Image Process. 2 (4), 429–441.

Chatterjee, S., Dimitrakopoulos, R., 2012. Multi-scale stochastic simulation with a wavelet-based approach. Comput. Geosci. 45, 177–189.

Chilès, J.-P., Delfiner, P., 1999. Geostatistics - Modeling Spatial Uncertainty. John Wiley & Sons, Inc., New Jersey.

Chugunova, T., Hu, L., 2008. Multiple-point simulations constrained by continuous auxiliary data. Math. Geosci. 40 (2), 133–146.

Ciannelli, L., Fauchald, P., Chan, K., Agostini, V., Dingsor, G., 2008. Spatial fisheries ecology: recent progress and future prospects. J. Mar. Syst. 3–4 (71), 223–236.

Comunian, A., Renard, P., Straubhaar, J., 2012. 3D multiple-point statistics simulation using 2D training images. Comput. Geosci. 40, 49–65.

Cressie, N., Wikle, K., 2011. Statistics for Spatio-Temporal Data. Wiley, New Jersey.

Criminisi, A., Pérez, P., Toyama, K., 2004. Region filling and object removal by exemplar-based image inpainting. IEEE Trans. Image Process. 13 (9), 1200–1212.

Cross, G.R., Jain, A.K., 1983. Markov random field texture models. IEEE Trans. Pattern Anal. Mach. Intell. PAMI-5 (1), 25–39.

Daly, C., 2004. Higher order models using entropy, Markov random fields and sequential simulation. Paper presented at Geostatistics Banff 2004. Kluwer Academic Publisher, Banff, Alberta

Darabi, S., Shechtman, E., Barnes, C., Goldman, D., 2012. Image melding: combining inconsistent images using patch-based synthesis. In: Proceedings of SIGGRAPH 2012, ACM Transactions on Graphics, vol. 31 (4), pp. 82:81–82:10.

de Bonet, J.S., 1997. Multiresolution sampling procedure for analysis and synthesis of texture images. Paper presented at Proceedings of the ACM SIGGRAPH Conference on Computer Graphics.

de Iaco, S., Maggio, S., 2011. Validation techniques for geological patterns simulations based on variogram and multiple-point statistics. Math. Geosci. 43 (4), 483–500.

de Marsily, G., Delay, F., Gonçalvès, J., Renard, P., Teles, V., Violette, S., 2005. Dealing with spatial heterogenity. Hydrogeol. J. 13 (1), 161–183.

de Vries, L., Carrera, J., Falivene, O., Gratacos, O., Slooten, L., 2009. Application of multiple point geostatistics to non-stationary images. Math. Geosci. 41 (1), 29–42.

Deutsch, C., Journel, A., 1992. GSLIB: Geostatistical Software Library. Oxford University Press, New York

Deutsch, C., Tran, T., 2002. FLUVSIM: a program for object-based stochastic modeling of fluvial depositional systems. Comput. Geosci. 28 (4), 525–535.

Deutsch, C.V., Journel, A.G., 1991. Application of simulated annealing to stochastic reservoir modeling. Society of Petroleum Engineers of AIME, (Paper) SPE.

Dijkstra, E., 1959. A note on two problems in connexion with graphs. Numer. Math. 1, 269–271.

Dimitrakopoulos, R., Mustapha, H., Gloaguen, E., 2010. High-order statistics of spatial random fields: exploring spatial cumulants for modeling complex non-Gaussian and non-linear phenomena. Math. Geosci. 42 (1), 65–99.

Dong, Y., Lefebvre, S., Tong, X., Drettakis, G., 2008. Lazy solid texture synthesis. Comput. Graph. Forum 27 (4), 1165–1174.

Drori, I., Cohen-Or, D., Yeshurun, H., 2003. Fragment-based image completion. ACM Trans. Graph. 22 (3), 303–312.

Efros, A. and T. Leung (1999). Texture synthesis by non-parametric sampling. the Seventh IEEE International Conference on Computer Vision, 1999, Kerkyra, Greece, volume 2, pp. 1033-1038.

Efros, A.A., Freeman, W.T., 2001. Image quilting for texture synthesis and transfer. Paper presented at Proceedings of the ACM SIGGRAPH Conference on Computer Graphics.

El Ouassini, A., Saucier, A., Marcotte, D., Favis, B., 2008. A patchwork approach to stochastic simulation: a route towards the analysis of morphology in multiphase systems. Chaos Solitons Fract. 36, 418–436.

Emery, X., Lantuéjoul, C., 2014. Can a Training Image Be a Substitute for a Random Field Model? Mathematical Geosciences, 46 (2), 133-147. doi: 10.1007/s11004-013-9492-z.

Frodeman, R., 1995. Geological reasoning: geology as an interpretive and historical science. Geol. Soc. Am. Bull. 107 (8), 960–968.

Galerne, B., Lagae, A., Lefebvre, S., Drettakis, G., 2012. Gabor noise by example. ACM Trans. Graph. 31, 4.

Gloaguen, E., Dimitrakopoulos, R., 2009. Two-dimensional conditional simulations based on the wavelet decomposition of training images. Math. Geosci. 41 (6), 679–701 (special issue).

Gómez-Hernández, J.J., Wen, X.H., 1998. To be or not to be multi-Gaussian? A reflection on stochastic hydrogeology. Adv. Water Resour. 21 (1), 47–61.

Goovaerts, P., 1997. Geostatistics for Natural Resources Evaluation. Oxford University Press, Oxford

Goovaerts, P., 2000. Geostatistical approaches for incorporating elevation into the spatial interpolation of rainfall. J. Hydrol. 228 (1–2), 113–129.

Goovaerts, P., 2010. Visualizing and testing the impact of place on late-stage breast cancer incidence: a non-parametric geostatistical approach. Health Place 16 (2), 321–330.

Guardiano, F., Srivastava, M., 1993. Multivariate geostatistics: beyond bivariate moments. In: Geostatistics-Troia. editor: A. Soares, Kluwer Academic, Dordrecht, the Netherlands, pp. 133–144

Hajizadeh, A., Safekordi, A., Farhadpour, F.A., 2011. A multiple-point statistics algorithm for 3D pore space reconstruction from 2D images. Adv. Water Resour. 34 (10), 1256–1267.

Haldorsen, H.H., Chang, D.M., 1986. Notes on stochastic shales from out crop to simulation models. In: Reservoir Characterization. editors: L. W. Lake and H. B. Carrol. Academic, New York, pp. 152–167

Han, C., Risser, E., Ramamoorthi, R., Grinspun, E., 2008. Multiscale texture synthesis. ACM Trans. Graph. 27 (3).

Hansen, T.M., Cordua, K.S., Mosegaard, K., 2012. Inverse problems with non-trivial priors: efficient solution through sequential Gibbs sampling. Comput. Geosci. 16 (3), 593–611.

Haslauer, C.P., Guthke, P., Brdossy, A., Sudicky, E.A., 2012. Effects of non-Gaussian copula-based hydraulic conductivity fields on macrodispersion. Water Resour. Res. 48 (7).

Heeger, D.J., Bergen, J.R., 1995. Pyramid-based texture analysis/synthesis. Paper presented at Proceedings of the ACM SIGGRAPH Conference on Computer Graphics.

Hendricks Franssen, H.J., Alcolea, A., Riva, M., Bakr, M., van der Wiel, N., Stauffer, F., Guadagnini, A., 2009. A comparison of seven methods for the inverse modelling of groundwater flow. Application to the characterisation of well catchments. Adv. Water Resour. 32 (6), 851–872.

Hertzmann, A., Jacobs, C.E., Oliver, N., Curless, B., Salesin, D.H., 2001. Image analogies. Paper presented at Proceedings of the ACM SIGGRAPH Conference on Computer Graphics (SIGGRAPH 2001), Los Angeles, CA, 12–17 August 2001.

Honarkhah, M., Caers, J., 2010. Stochastic simulation of patterns using distance-based pattern modeling. Math. Geosci. 42 (5), 487–517.

Honarkhah, M., Caers, J., 2012. Direct pattern-based simulation of non-stationary geostatistical models. Math. Geosci. 44 (6), 651–672.

Hu, L., Chugunova, T., 2008. Multiple-point geostatistics for modeling subsurface heterogeneity: a comprehensive review. Water Resour. Res. 44(W11413).

Huang, H.D., Tong, X., Wang, W.C., 2007. Accelerated parallel texture optimization. J. Comput. Sci. Technol. 22 (5), 761–769.

Huang, T., Li, X., Zhang, T., Lu, D.T., 2013a. GPU-accelerated Direct Sampling method for multiple-point statistical simulation. Comput. Geosci. 57, 13–23.

Huang, T., Lu, D.T., Li, X., Wang, L., 2013b. GPU-based SNESIM implementation for multiple-point statistical simulation. Comput. Geosci. 54, 75–87.

Huysmans, M., Dassargues, A., 2011. Direct multiple-point geostatistical simulation of edge properties for modeling thin irregularly shaped surfaces. Math. Geosci. 43 (5), 521–536.

Huysmans, M., Dassargues, A., 2012. Modeling the effect of clay drapes on pumping test response in a cross-bedded aquifer using multiple-point geostatistics. J. Hydrol. 450–451, 159–167.

Isaaks, E., 1984. Indicator simulation: application to the simulation of a high grade uranium mineralization. In: Geostatistics for Natural Resources Characterization, Part2. Editors: G. Verly, D. Reidel Publishing Company, New York, pp. 1057–1069

Jha, S., Mariethoz, G., Evans, J.P., McCabe, M., 2013a. Demonstration of a geostatistical approach to physically-consistent downscaling of climate modeling simulations. Water Resour. Res. 49, 1–14.

Jha, S., Mariethoz, G., Kelly, B.F. J., 2013b. Bathymetry fusion using multiple-point geostatistics: novelty and challenges in representing non-stationary bedforms. Environ. Model. Softw. 50, 66–76.

Journel, A., Huijbregts, C., 1978. Mining Geostatistics. The Blackburn Press, Caldwell, New Jersey.

Journel, A., Alabert, F., 1989. Non-Gaussian data expansion in the earth sciences. Terra Nova 1, 123–134.

Journel, A., Zhang, T., 2006. The necessity of a multiple-point prior model. Math. Geol. 38 (5), 591–610.

Jung, A., Aigner, T., 2012. Carbonate geobodies: Hierarchical classification and database – a new workflow for 3D reservoir modelling. J. Pet. Geol. 35 (1), 49–65.

Kanevski, M., Maignan, M., Pozdnoukhov, A., Timonin, V., 2008. Interest rates mapping. Physica A-Stat. Mech. Appl. 15 (387), 3897–3903.

Kerrou, J., Renard, P., Hendricks Franssen, H.J., Lunati, I., 2008. Issues in characterizing heterogeneity and connectivity in non-multiGaussian media. Adv. Water Resour. 31 (1), 147–159.

Kitanidis, P. (1997). Introduction to Geostatistics: Applications in Hydrogeology. Cambridge, Cambridge University Press.

Kitanidis, P., 1995. Quasi-linear geostatistical theory for inversing. Water Resour. Res. 31 (10), 2411–2419.

Kjønsberg, H., Kolbjørnsen, O., 2008. Markov mesh simulations with data conditioning through indicator kriging. In: Proceedings of the Eighth International Geostatistics Congress 2008, 1–5 December 2008, pp. 257–267.

Kopf, J., Fu, C.W., Cohen-Or, D., Deussen, O., Lischinski, D., Wong, T.T., 2007. Solid texture synthesis from 2D exemplars. ACM Trans. Graph. 26, 3.

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

Kwatra, V., Essa, I., Bobick, A., Kwatra, N., 2005. Texture optimization for example-based synthesis. ACM Trans. Graph. 24 (3), 795–802.

Lagae, A., Lefebvre, S., Cook, R., DeRose, T., Drettakis, G., Ebert, D.S., Lewis, J.P., Perlin, K., Zwicker, M., 2010. A survey of procedural noise functions. Comput. Graph. Forum 29 (8), 2579–2600.

Lange, K., Frydendall, J., Cordua, K.S., Hansen, T.M., Melnikova, Y., Mosegaard, K., 2012. A frequency matching method: solving inverse problems by use of geologically realistic prior information. Math. Geosci. 44 (7), 783–803.

Lantuejoul, C., 2002. Geostatistical Simulation: Models and Algorithms. Springer, Berlin.

Lasram, A., Lefebvre, S., 2012. Parallel patch-based texture synthesis. Paper presented at High Performance Graphics, Paris, June 25–27, 2012.

Lefebvre, S., Hoppe, H., 2005. Parallel controllable texture synthesis. ACM Trans. Graph. 24 (3), 777–786.

Lefebvre, S., Hoppe, H., 2006. Appearance-space texture synthesis. ACM Trans. Graph. 25 (3), 541–548.

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

Liu, Y., Journel, A., 2004. Improving sequential simulation with a structured path guided by information content. Math. Geol. 36 (8), 945–964.

Liu, Y., 2006. Using the Snesim program for multiple-point statistical simulation. Comput. Geosci. 23, 1544–1563.

Lloyd, S., 1982. Least squares quantization in PCM. IEEE Trans. Inform. Theory IT-28 (2 part 1), 129–137.

Lu, J., Georghiades, A.S., Glaser, A., Wu, H., Wei, L.Y., Guo, B., Dorsey, J., Rushmeier, H., 2007. Context-aware textures. ACM Trans. Graph. 26, 1.

Manke, F., Wunsche, B., 2009. Fast spatially controllable 2D/3D texture synthesis and morphing for multiple input textures. Paper presented at GRAPP 2009 – International Conference on Computer Graphics Theory and Applications, 5–8 February 2009, Lisboa, Portugal.

Mariethoz, G., Renard, P., Froidevaux, R., 2009. Integrating collocated auxiliary parameters in geostatistical simulations using joint probability distributions and probability aggregation. Water Resour. Res. 45 (8).

Mariethoz, G., Kelly, B.F. J., 2011. Modeling complex geological structures with elementary training images and transform-invariant distances. Water Resour Res. 47 (7).

Mariethoz, G., Kelly, B.F. J., Baker, A., 2012. Quantifying the value of laminated stalagmites for paleoclimate reconstructions. Geophys. Res. Lett. 39, 5.

Matheron, G., 1965. Les variables régionalisées et leur estimation. Masson, Paris.

Matheron, G., 1967. Eléments pour une théorie des milieux poreux. Masson, Paris.

Meerschman, E., Van Meirvenne, M., Mariethoz, G., Islam, M.M., De Smedt, P., Van De Vijver, E., Saey, T., 2014. Using bivariate multiple-point statistics and proximal soil sensor data to map fossil ice-wedge polygons. Geoderma 213, 571–577.

Meliker, J.R., Goovaerts, P., Jacquez, G.M., AvRuskin, G.A., Copeland, G., 2009. Breast and prostate cancer survival in Michigan: can geographic analyses assist in understanding racial disparities? Cancer 115 (10), 2212–2221.

Mustapha, H., Dimitrakopoulos, R., Chatterjee, S., 2011. Geologic heterogeneity representation using high-order spatial cumulants for subsurface flow and transport simulations. Water Resour. Res. 47 (8).

Neuweiler, I., Vogel, H.J., 2007. Upscaling for unsaturated flow for non-Gaussian heterogeneous porous media. Water Resour. Res. 43, 3.

Neuweiler, I., Papafotiou, A., Class, H., Helmig, R., 2011. Estimation of effective parameters for a two-phase flow problem in non-Gaussian heterogeneous porous media. J. Contam. Hydrol. 120–121 (C), 141–156.

Paget, R., Longstaff, I.D., 1998. Texture synthesis via a noncausal nonparametric multiscale Markov random field. IEEE Trans. Image Process. 7 (6), 925–931.

Parra, A., Ortiz, J.M., 2011. Adapting a texture synthesis algorithm for conditional multiple point geostatistical simulation. Stoch. Environ. Res. Risk Assess. 25 (8), 1101–1111.

Peredo, O., Ortiz, J.M., 2011. Parallel implementation of simulated annealing to reproduce multiple-point statistics. Comput. Geosci. 37 (8), 1110–1121.

Pham, T.D., 2012. Supervised restoration of degraded medical images using multiple-point geostatistics. Comput. Methods Programs Biomed. 106 (3), 201–209.

Popat, K., Picard, R., 1993. Novel cluster-based probability model for texture synthesis, classification, and compression. Paper presented at SPIE Visual Communications and Image Processing '93, Boston, November 8–11, 1993.

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

Relethford, J., 2008. Geostatistics and spatial analysis in biological anthropology. Am. J. Phys. Anthropol. 1 (136), 1–10.

Renard, P., Straubhaar, J., Caers, J., Mariethoz, G., 2011. Conditioning facies simulations with connectivity data. Math. Geosci. 43 (8), 879–903.

Ronayne, M., Gorelick, S., Caers, J., 2008. Identifying discrete geologic structures that produce anomalous hydraulic response: an inverse modeling approach. Water Resour. Res. 44(W08426).

Rufino, M., Stelzenmueller, V., Maynou, F., Zauke, G., 2006. Assessing the performance of linear geostatistical tools applied to artificial fisheries data. Fish. Res. 82 (1–3), 263–279.

Sánchez-Vila, X., Carrera, J., Girardi, J.P., 1996. Scale effects in transmissivity. J. Hydrol. 183 (1–2), 1–22.

Stien, M., Kolbjørnsen, O., 2011. Facies modeling using a Markov mesh model specification. Math. Geosci. 43 (6), 611–624.

Straubhaar, J., Renard, P., Mariethoz, G., Froidevaux, R., Besson, O., 2011. An improved parallel multiple-point algorithm using a list approach. Math. Geosci. 43 (3), 305–328.

Straubhaar, J., Walgenwitz, A., Renard, P., 2013. Parallel multiple-point statistics algorithm based on list and tree structures. Math. Geosci. 45 (2), 131–147.

Strebelle, S., 2002. Conditional simulation of complex geological structures using multiple-point statistics. Math. Geol. 34 (1), 1–22.

Sun, J., Yuan, L., Jia, J., Shum, H.Y., 2005. Image completion with structure propagation. ACM Trans. Graph. 24, 861–868.

Tahmasebi, P., Hezarkhani, A., Sahimi, M., 2012a. Multiple-point geostatistical modeling based on the cross-correlation functions. Comput. Geosci. 16 (3), 779–797.

Tahmasebi, P., Sahimi, M., Mariethoz, G., Hezarkhani, A., 2012b. Accelerating geostatistical simulations using graphics processing units (GPU). Comput. Geosci. 46, 51–59.

Tahmasebi, P., Sahimi, M., 2013. Cross-correlation function for accurate reconstruction of heterogeneous media. Phys. Rev. Lett. 110, 078002.

Tang, Y., Atkinson, P.M., Wardrop, N.A., Zhang, J., 2013. Multiple-point geostatistical simulation for post-processing a remotely sensed land cover classification. Spat. Stat. 5, 69–84.

Tatem, A., Lewis, H., Atkinson, P., Nixon, M., 2002. Super-resolution land cover pattern prediction using a Hopfield neural network. Remote Sens. Environ. 79, 1–14.

Toftaker, H., Tjelmeland, H., 2013. Construction of binary multi-grid Markov random field prior models from training images. Math. Geosci. 45 (4), 383–409.

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

ver Hoef, J., 2008. Spatial methods for plot-based sampling of wildlife populations. Environ. Ecol. Stat. 1 (15), 3–13.

Wang, L., Zhou, K., Yu, Y., Guo, B., 2010. Vector solid textures. ACM Trans. Graph. 29, 4.

Webb, E., Anderson, M., 1996. Simulation of preferential flow in three-dimensional, heterogeneous conductivity fields with realistic internal architecture. Water Resour. Res. 32 (3), 533–546.

Wei, L., Levoy, M., 2000. Fast texture synthesis using tree-structured vector quantization. Paper presented at SIGGRAPH '00: 27th Annual Conference on Computer Graphics and Interactive Techniques. ACM Press/Addison-Wesley, New Orleans.

Wei, L. and M. Levoy (2001). Texture synthesis over arbitrary manifold surfaces. SIGGRAPH '01, 28th annual conference on Computer graphics and interactive techniques, Los Angeles, California, 12-17 August 2001, ACM, New York, pp. 355-360.

Wei, L., 2002. Texture Synthesis by Fixed Neighborhood Searching. Electrical Engineering, Stanford University. PhD thesis.

Wei, L., Levoy, M., 2003. Order-independent texture synthesis. Paper presented at Stanford Computer Science TR-2002-01.

Wei, L., Lefebvre, S., Kwatra, N., Yurk, G., 2009. State of the Art in example-based texture synthesis. Paper presented at EUROGRAPHICS 2009, 30 March–3 April, Munich, Germany.

Welsh, T., M. Ashikhmin, et al. (2002). Transferring color to greyscale images. SIGGRAPH '02, 29th annual conference on Computer graphics and interactive techniques San Antonio, 21-26 July 2002, ACM, New York, pp. 277-280.

Western, A., Blöschl, G., Grayson, R., 2001. Toward capturing hydrologically significant connectivity in spatial patterns. Water Resour. Res. 37 (1), 83–97.

Wu, J., Boucher, A., Zhang, T., 2008. A SGeMS code for pattern simulation of continuous and categorical variables: FILTERSIM. Comput. Geosci. 34 (12), 1863–1876.

Wu, Q. and Y. Yu (2004). "Feature matching and deformation for texture synthesis." ACM transactions on Graphics 23(3): 364-367.

Wu, Y.N., Zhu, S.C., Liu, X., 2000. Equivalence of Julesz ensembles and FRAME models. Int. J. Comput. Vis. 38 (3), 247–265.

Xu, Y.Q., Guo, B., Shum, H., 2000. Chaos mosaic: fast and memory efficient texture synthesis. Microsoft Res.

Zelinka, S., Garland, M., 2002. Towards real-time texture synthesis with the jump map. Paper presented at 13th Eurographics Workshop on Rendering, Pisa, 26–28 June 2002.

Zhang, J., Zhou, K., Velho, L., Guo, B., Shum, H.Y., 2003. Synthesis of progressively-variant textures on arbitrary surfaces. Paper presented at ACM SIGGRAPH 2003, San Diego, 27–30 July 2003.

Zhang, T., Switzer, P., Journel, A., 2006. Filter-based classification of training image patterns for spatial simulation. Math. Geol. 38 (1), 63–80.

Zhang, T., Pedersen, S.I., Knudby, C., McCormick, D., 2012. Memory-efficient categorical multi-point statistics algorithms based on compact search trees. Math. Geosci. 44 (7), 863–879.

Zhang, Z., Switzer, P., 2007. Stochastic space-time regional rainfall modeling adapted to historical rain gauge data. Water Resour. Res. 43, 3.

Zhao, S., Bala, K., Marschner, S., Jakob, W., 2012. Structure-aware synthesis for predictive woven fabric appearance. ACM Trans. Graph. 31 (4) (Article 75).

Zhu, S.C., Wu, Y., Mumford, D., 1998. Filters, Random Fields and Maximum Entropy (FRAME): towards a unified theory for texture modeling. Int. J. Comput. Vis. 27 (2), 107–126.

Zimmerman, D., De Marsily, G., Gotway, C., Marietta, M., Axness, C., Beauheim, R., Bras, R., Carrera, J., Dagan, G., Davies, P., Gallegos, D., Galli, A., Gomez-Hernandez, J., Grindrod, P., Gutjahr, A., Kitanidis, P., Lavenue, A., McLaughlin, D., Neuman, S., RamaRao, B., Ravenne, C., Rubin, Y., 1998. A comparison of seven geostatistically based inverse approaches to estimate transmissivities for modeling advective transport by groundwater flow. Water Resour. Res. 34 (6), 1373–1413.

Zinn, B., Harvey, C., 2003. When good statistical models of aquifer heterogeneity go bad: a comparison of flow, dispersion, and mass transfer in connected and multivariate Gaussian hydraulic conductivity fields. Water Resour. Res. 39 (3), WR001146.