Static and Dynamic Texture Mixing Using Optimal Transport

Sira Ferradans[1,⋆], Gui-Song Xia[2],
Gabriel Peyré[1], and Jean-François Aujol[3]

[1] Ceremade, Univ. Paris-Dauphine
{sira.ferradans,gabriel.peyre}@ceremade.dauphine.fr
[2] LIERSMARS, Wuhan University
guisong.xia@whu.edu.cn
[3] IMB, Université Bordeaux 1
Jean-Francois.Aujol@math.u-bordeaux1.fr

Abstract. This paper tackles the problem of mixing static and dynamic texture by combining the statistical properties of an input set of images or videos. We focus on Spot Noise textures that follow a stationary and Gaussian model which can be learned from the given exemplars. From here, we define, using Optimal Transport, the distance between texture models, derive the geodesic path, and define the barycenter between several texture models. These derivations are useful because they allow the user to navigate inside the set of texture models, interpolating a new one at each element of the set. From these new interpolated models, new textures can be synthesized of arbitrary size in space and time. Numerical results obtained from a library of exemplars show the ability of our method to generate new complex and realistic static and dynamic textures.

1 Introduction

The problem of synthesizing new textures is central in Image Processing and Computer Graphics. In order to render scenes for video games or animation films, a texture is mapped onto a given surface. Because the shape and extension of the surface may vary, the main goal of texture synthesis is to be able to generate as much texture as it is needed in a fast and realistic way. This problem has been addressed since the beginning of Computer Graphics, so we can find many solutions in the literature.

1.1 Previous Works

Copy-Based Methods. These methods are adapted to complicated (not even random) textures. The main assumption is that textures contain repeating local patterns. They synthesize new textures by copying patches or pixels from the original image in a way that preserves local structure. First proposed by Popat

⋆ This work has been supported by the European Research Council (ERC project SIGMA-Vision).

A. Kuijper et al. (Eds.): SSVM 2013, LNCS 7893, pp. 137–148, 2013.

and Picard [1] in the context of clustering, it was simplified and popularized by Efros and Leung [2] for texture synthesis. For a thorough review of copy-based methods we refer the reader to the article [3].

Statistical Texture Models. Statistical parametric models are generally not as good in handling complex texture patterns, but are more flexible and fast, see for instance [4]. The main assumption of these models is that textures are modeled by a probability distribution. Thus, texture analysis consists in estimating the probability function and texture synthesis amounts to generate new realizations of this probability distribution. Many methods have been proposed within this category, specially relevant is the use of Markov random fields (i.e. [5]) which model also copy-based methods (see for instance [6]) or stationary Gaussian random fields [7].

Spot Noise models were first introduced by van Wijk [8] and are stationary models that replicate, in random locations, simple spot images. Galerne et al.[7] analyze the asymptotical behavior of Van Wijk's method to propose a new method (Asymptotic Discrete Spot Noise), which consists in modeling texture with a stationary Gaussian distribution. In this paper, we focus our attention on this texture model and extend this framework to texture mixing.

Dynamic Texture Synthesis. Many methods for static image synthesis have been adapted to the dynamic scenario (see for example [9], [6] in the context of the copy-based methods), but very few have studied the specific dynamics of texture in time. In the context of Gaussian textures, linear dynamical systems [10] and dynamic multiscale autoregressive models [11] have been proposed to model the evolution of texture with time. However, these methods define models that are difficult to manipulate (for instance to achieve model mixing.) Recently, an extension to Galerne et al.'s model [7] for stationary Gaussian dynamic textures has been proposed by Xia et al. [12]. In this paper, Xia et al. model dynamic texture as a 3D Gaussian random field, with stationarity in space *and* time. Here, we take advantage of this extension to generate new mixed models from input dynamic textures.

Texture Mixing. More complex textures can be obtained by texture mixing which extends the traditional texture synthesis by considering the interplay between several texture models. This is a difficult problem since it requires to average very distinct statistical features. Previous works make use of mixture models, see for instance [13]. The use of non-parameteric histogram averaging has also been proposed for grayscale [14] as well as color and wavelets features [15]. We propose here a simpler approach that makes use of a parameterization of the Gaussian texture model. Defining a geodesic path with Optimal Transport (OT) between the original Gaussian models, we can generate new textures sharing the characteristics of the input ones. The proposed method ensures that the new texture model stays Gaussian.

1.2 Contributions

We propose a new framework for texture synthesis based on the definition of geodesic paths between stationary Gaussian texture models. Our first

contribution is the definition of the geodesic distance, according to OT, between texture models, and the geodesic path associated to such distance. The straightforward consequence of having this geodesic path is that we obtain a method for interpolating new texture models with the statistical properties of the input textures. Our second contribution consists in the extension of the interpolation formula between two models to *several* models by defining the OT barycenter. The final algorithm is solidly founded, the texture synthesis is fast, and the obtained results look natural.

2 Spot Noise Texture Model

We model textures as stationary Gaussian random fields. These assumptions allow us to learn the texture model parameters from a single texture exemplar.

2.1 Notations

Deterministic input exemplar textures are represented as $f \in \mathbb{R}^{N\times d}$, where $N = \prod_{j=1}^{k} N_j$ is the number of pixels ($k = 2$ for image and $k = 3$ for videos) and d is the number of channels ($d = 1$ for grayscale and $d = 3$ for color datasets). We refer to $f(x) \in \mathbb{R}^3$ to the color vector at position x, where there are N such positions x. We denote Gaussian distributions as $\mu = \mathcal{N}(m, \Sigma)$ where $m \in \mathbb{R}^{N\times d}$ is the mean of the distribution and $\Sigma \in \mathbb{R}^{Nd\times Nd}$ is a positive semi-definite covariance matrix.

The k-dimensional discrete Fourier transform $\hat{f} \in \mathbb{R}^{N\times d}$ of $f \in \mathbb{R}^{N\times d}$ is defined as

$$\forall\, \omega = (\omega_1, \ldots, \omega_k), \quad \hat{f}(\omega) = \sum_x f(x) e^{-\sum_j \frac{2i\pi}{N_j}\omega_j x_j} \in \mathbb{R}^d.$$

It is computed in $O(Nd\log(Nd))$ operations and it is inverted with the same complexity using the inverse FFT.

Given two periodic images or videos $f, g \in \mathbb{R}^N$, we define the convolution $h = f \star g$ of f and g as

$$h(x) = \sum_y f(x-y)g(y) \quad \Longleftrightarrow \quad \hat{h}(\omega) = \hat{f}(\omega)\hat{g}(\omega). \tag{1}$$

2.2 Stationary Gaussian Models

We model a texture as a random vector X distributed according to some Gaussian distribution μ, which we denote $X \sim \mu$. A random vector X is stationary if the distribution of $X(\cdot)$ and $X(\cdot + \tau)$ are the same, for any translation vector $\tau \in \mathbb{Z}^k$, where we assume periodic boundary conditions. Section 2.4 details how to learn the parameters when the input exemplar is non-periodic.

The fact that X is stationary implies that the mean $m(x) \in \mathbb{R}^d$ is independent of the position x and the covariance operator Σ is block-diagonal over the Fourier domain, thus it can be computed using convolutions, that is to say, the covariance operator $y = \Sigma f$ can be applied over the Fourier domain as $\hat{y}(\omega) = \hat{\Sigma}(\omega)\hat{f}(\omega)$ where $\hat{\Sigma}(\omega) \in \mathbb{C}^{d\times d}$ is a positive Hermitian matrix.

2.3 Spot Noise Model

A Spot Noise (SN) random vector $X = (X_1, \ldots, X_d)$ is a Gaussian texture model obtained from an input texture $f = (f_1, \ldots, f_d)$ by convolving each channel with the same Gaussian white noise [7]. This reads

$$\forall\, j = 1, \ldots, d, \quad X_j = m_j + f_j \star W \tag{2}$$

where $\star$ is the k-dimensional periodic convolution and the W is a white noise $W \sim \mathcal{N}(0, \mathrm{Id}_N/\sqrt{N})$, and m_j is the mean of f_j. We denote $\mu = \mu(f)$ the distribution of this random vector X, which is the SN distribution associated to the exemplar f.

Equivalently, Spot Noise models are the stationary Gaussian vectors for which the matrices $\hat{\Sigma}(\omega)$ are rank one, and can thus be decomposed as

$$\hat{\Sigma}(\omega) = \hat{f}(\omega)\hat{f}(\omega)^*, \tag{3}$$

where $u^* \in \mathbb{C}^d$ is the complex conjugate transpose of $u \in \mathbb{C}^d$.

2.4 Stationary Gaussian Model Synthesis

Once the parameters Σ and m of the Gaussian model $\mu = \mathcal{N}(m, \Sigma)$ have been computed, the synthesis of a texture $g \in \mathbb{R}^{N \times d}$ is obtained using a realization of the Gaussian process.

For a generic stationary model, this is achieved by computing the Cholesky factorization of the frequency covariance $\hat{\Sigma}(\omega) = \hat{A}(\omega)\hat{A}(\omega)^*$ where $A(\omega)^* \in \mathbb{C}^{d \times d}$ is the complex conjugate transpose of the matrix $A(\omega) \in \mathbb{C}^{d \times d}$. Then, we compute $\hat{g}(\omega) = \hat{A}(\omega)\hat{w}(\omega)$ for $\omega \neq 0$ where w is a realization of $\mathcal{N}(0, \mathrm{Id}_{Nd}/\sqrt{N})$ and $\hat{g}(0) = Nm(0)$ is the constant mean of the model.

In the special case where the model is a Spot Noise $\mu(f)$, meaning that $\hat{\Sigma}(\omega) = \hat{f}(\omega)\hat{f}(\omega)^*$, the synthesis is even faster using, for $\omega \neq 0$, $\hat{g}(\omega) = \hat{w}(\omega)\hat{f}(\omega)$, or equivalently using a realization of the convolution formula (2).

Boundary Conditions. Up to now, the image is assumed to be periodic in our texture model. To be able to learn the parameters from a non-periodic image, a preprocessing is required. Symmetrizing the image with respect to the boundaries introduces axis-aligned artifacts. Following [7], we substitute each channel f_j of the input exemplar by its periodic component as defined by Moisan [16].

Extending the Texture Size. In our context, the process of extending the input texture of size $N_1 \times N_2 \times N_3$ to any arbitrary size $M_1 \times M_2 \times M_3$ can be done following the method proposed by Galerne et al. [7]. The periodic component of the original texture is located at the center of a flat new image (or video) of value m and dimensions $M_1 \times M_2 \times M_3$. To avoid the introduction of high frequencies, the new borders are smoothed with a spatial windowing function. This extended image or video is then used to learn a texture model of size $M_1 \times M_2 \times M_3$.

3 Optimal Transport Geodesic of Spot Noise

Now that we have defined our texture model we proceed to the exposition of our model mixing method. It operates using OT geodesic over the set of Gaussian distributions. This is achieved by defining the OT geodesic interpolation [17] over the space of Gaussian models.

3.1 Optimal Transport Geodesics of Gaussian Fields

The first step is to define a geodesic distance between texture models, that is to say, between two arbitrary stationary Gaussian distributions. The L^2 OT distance between $\mu_i = \mathcal{N}(m_i, \Sigma_i)$ reads:

$$d(\mu_0, \mu_1)^2 = \operatorname{tr}(\Sigma_0 + \Sigma_1 - 2\Sigma_{0,1}) + \|m_0 - m_1\|^2,$$

where $\Sigma_{0,1} = (\Sigma_0^{1/2}\Sigma_1\Sigma_0^{1/2})^{1/2}$ (see for instance [18]).

This distance is known to be geodesic, meaning that $d(\mu_0, \mu_1)$ is equal to the length of the shortest path (the so-called geodesic path) $t \in [0,1] \mapsto \mu_t$ between μ_0 and μ_1. This geodesic path satisfies

$$\forall\, t \in [0,1], \mu_t = \operatorname*{argmin}_{\mu} (1-t)d(\mu_0, \mu)^2 + td(\mu_1, \mu)^2,$$

where $t \mapsto \mu_t$ parameterizes the path, so μ_t can also be understood as a weighted barycenter of the input texture models. The following proposition shows that this geodesic path is composed of Gaussian models, so that the set of Gaussian models are geodesically convex for the OT distance [19].

Proposition 1. *If* $\ker(\Sigma_0) \not\subset \ker(\Sigma_1)^\perp$ *and* $\operatorname{rank}(\Sigma_0) \geqslant \operatorname{rank}(\Sigma_1)$*, the unique Gaussian OT-geodesic of Gaussian distributions* $\mu_i = \mathcal{N}(m_i, \Sigma_i)$ *(for* $i = 0, 1$*) is a Gaussian distribution* $\mathcal{N}(m_t, \Sigma_t)$ *where* $m_t = (1-t)m_0 + tm_1$ *and*

$$\Sigma_t = [(1-t)\mathrm{Id} + t\Pi]\Sigma_0[(1-t)\mathrm{Id} + t\Pi] \tag{4}$$

where $\Pi = \Sigma_1^{1/2}\Sigma_{0,1}^{+}\Sigma_1^{1/2}$ *and where* A^+ *is the Moore-Penrose pseudo-inverse and* $A^{1/2}$ *is the unique positive square root of a symmetric semi-definite matrix.*

Proof. The proof follows the one in [19] with the extra care that the covariance can be rank-deficient, hence requiring a pseudo-inverse.

Note that the condition $\operatorname{rank}(\Sigma_0) \geqslant \operatorname{rank}(\Sigma_1)$ is not restrictive since one can otherwise exchange the roles of Σ_0 and Σ_1 and replace t by $1-t$ when computing the geodesic path.

We now show that if the input models μ_0, μ_1 are Spot Noise, then the geodesic interpolation is also Spot Noise. This means that the texture models we consider are geodesically convex.

Theorem 1. *For $i = 0, 1$, let $\mu_i = \mu(f^{[i]})$ be Spot Noise distributions associated with $f^{[0]}, f^{[1]} \in \mathbb{R}^{N\times d}$. The OT geodesic path μ_t defined in equation (4) is a Spot Noise model $\mu_t = \mu(f^{[t]})$ where $f^{[t]} = (1-t)f^{[0]} + tg^{[1]}$ with*

$$\forall\, \omega, \quad \hat{g}^{[1]}(\omega) = \hat{f}^{[1]}(\omega)\frac{\hat{f}^{[1]}(\omega)^*\hat{f}^{[0]}(\omega)}{|\hat{f}^{[1]}(\omega)^*\hat{f}^{[0]}(\omega)|}. \tag{5}$$

Proof. The covariance operator is a matrix-convolution operator, thus we can define in the Fourier domain its associated kernel as $\hat{\Sigma}_i(\omega) = \hat{f}^{[i]}(\omega)\hat{f}^{[i]}(\omega)^* \in \mathbb{C}^{d\times d}$. The symmetric operator Π from equation (4) is also a matrix convolution $\Pi g = \pi \star g$ with kernel whose Fourier transform is

$$\hat{\pi}(\omega) = \hat{\Sigma}_1^{\frac{1}{2}}(\omega)\left(\hat{\Sigma}_1^{\frac{1}{2}}(\omega)\hat{\Sigma}_0(\omega)\hat{\Sigma}_1^{\frac{1}{2}}(\omega)\right)^{-\frac{1}{2}}\hat{\Sigma}_1^{\frac{1}{2}}(\omega).$$

Note that the square root of a rank-1 matrix can be easily computed as

$$\forall\, u \in \mathbb{C}^d, \quad (uu^*)^{1/2} = \frac{1}{|u|}uu^* \in \mathbb{C}^{d\times d}.$$

Using this property, together with the definition of $\hat{\Sigma}$, and denoting $u_i = \hat{f}^{[i]}(\omega)$ one proves that

$$\hat{\pi}(\omega) = \frac{1}{|u_1^*u_0|}u_1u_1^*(u_1u_1^*)^{-1}u_1u_1^* = \frac{u_1u_1^*}{|u_1^*u_0|}. \tag{6}$$

Observe that although the matrix $u_1^*u_0$ is non invertible, the above expression is correct because the mapping $\pi(\omega)$ is zero on the orthogonal of u_1.

The expression (4) of the covariance implies that it is also a matrix-convolution operator with kernel defined over the Fourier domain as

$$\hat{\Sigma}_t(\omega) = \hat{f}^{[t]}(\omega)\hat{f}^{[t]}(\omega)^* \in \mathbb{C}^{d\times d},$$

where

$$\hat{f}^{[t]}(\omega) = [(1-t)\mathrm{Id} + t\hat{\pi}(\omega)]\hat{f}^{[0]} \in \mathbb{C}^d.$$

Using the expression (6) for $\hat{\pi}(\omega)$, one thus has that $\mu_t = \mu(f^{[t]})$ is a Spot Noise model where $f^{[t]}$ is defined as

$$\hat{f}^{[t]}(\omega) = (1-t)\hat{f}^{[0]}(\omega) + t\underbrace{\frac{\hat{f}^{[1]}(\omega)^*\hat{f}^{[0]}(\omega)}{|\hat{f}^{[1]}(\omega)^*\hat{f}^{[0]}(\omega)|}\hat{f}^{[1]}(\omega)}_{\hat{g}^{[1]}(\omega)} \in \mathbb{C}^d.$$

Therefore the new interpolated models μ_t are Gaussian, Spot Noise, and their covariances can be computed by a suitable averaging of the Fourier transforms of the input exemplars. The pseudocode of the proposed method is provided in Fig. 1.

Table 1. Pseudocode for geodesic mixing between two input exemplars

Input: exemplars $(f^{[0]}, f^{[1]})$, weight $t \in [0,1]$.
Output: realization h of the interpolated model μ_t between $\mu_i = \mu(f^{[i]})$ for $i = 0, 1$.
Preprocessing: for $i = 0, 1$,
- Replace $f^{[i]}$ by its periodic component. If needed extend its size by zero padding.
- Compute the mean m_i and subtract it, $\forall\, x, f^{[i]}(x) \leftarrow f^{[i]}(x) - m_i$.
- Computes the Fourier transforms $\hat{f}^{[i]}$ of $f^{[i]}$ using FFTs.

Model mixing:
- Compute $\hat{g}^{[1]}$ with equation (5).
- $\forall\, \omega$, compute $\hat{f}^{[t]}(\omega) = (1-t)\hat{f}^{[0]}(\omega) + t\hat{g}^{[1]}(\omega)$
- Compute $m_t = (1-t)m_0 + tm_1 \in \mathbb{R}^d$.

Spot Noise synthesis:
- Compute a realization $w \in \mathbb{R}^N$ of $\mathcal{N}(0, \mathrm{Id}_N/\sqrt{N})$ (using e.g. Matlab `randn`).
- Compute the Fourier transform $\hat{w}$ of w using FFT.
- $\forall\, \omega \neq 0$, $\forall\, j = 1, \ldots, d$ compute $\hat{h}_j(\omega) = \hat{f}_j^{[t]}(\omega)\hat{w}(\omega)$. Set $\hat{h}(0) = Nm_t$.
- Compute $h \in \mathbb{R}^{N \times d}$ from $\hat{h}$ using the inverse FFT.

3.2 Numerical Results

Let us now show some results obtained with the Spot Noise geodesic mix method explained in this section. Each row of Figure 1 corresponds to a single experiment which consists in learning the Gaussian model of two input textures ($f^{[0]}$ and $f^{[1]}$) and interpolate new Gaussian models following the path between model $f^{[0]}$ and $f^{[1]}$. Note that the images in columns $t = 0, 1$ are instances of the original models. We would like to point out how this instances are perceptually similar to the original input textures.

Regarding the columns for $t = 1/3$ and $2/3$, we would like to point out how the color changes gradually as we move along the geodesic path and that the spacial patterns of the original textures are being mixed also in different proportion.

An example of dynamic texture mixing can be observed in Figure 2. Each row corresponds to a single video, where every image is a single frame, ordered from left to right. The first and last rows are the inputs and the two middle ones where interpolated with the geodesic mix method.

4 Optimal Transport Barycenter of Spot Noise

In the previous section, we explained how to create new texture models by following a geodesic path between the two input models. This section extends this idea to more exemplars using a geodesic barycenter of the models. In the case of 3 exemplars (resp. 4), this can be visualized by locating the input models on the vertices of a 2-D triangle (resp. 3-D tetrahedron). Computing the OT barycenter allows one to navigate inside the triangle (resp. tetrahedron).

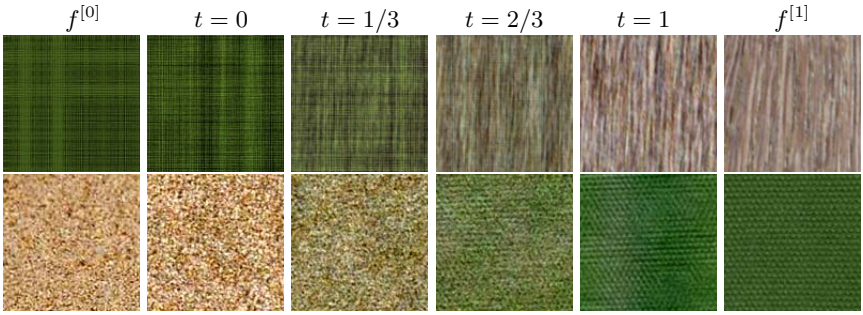

Fig. 1. $f^{[0]}$ and $f^{[1]}$ are the input texture images. After learning the input models, we interpolate new ones ($t = 0, 1/3, 2/3, 1$) along the OT geodesic path from $f^{[0]}$ to $f^{[1]}$.

4.1 Optimal Transport Barycenter

Given a family of Gaussian distributions $(\mu_i)_{i \in I}$ and weights ρ_i with $\sum_i \rho_i = 1$, where $\rho_i \geq 0$, the OT barycenter is defined as

$$\mu^\star = \underset{\mu}{\operatorname{argmin}} \sum_{i \in I} \rho_i d(\mu_i, \mu)^2. \tag{7}$$

Note that for $|I| = 2$ we retrieve the geodesic path by setting $t = \rho_2$. For the special case of a Gaussian distribution $\mu_i = \mathcal{N}(m_i, \Sigma_i)$, there is no close form solution if $|I| > 2$. The barycenter can be shown to be Gaussian [20]

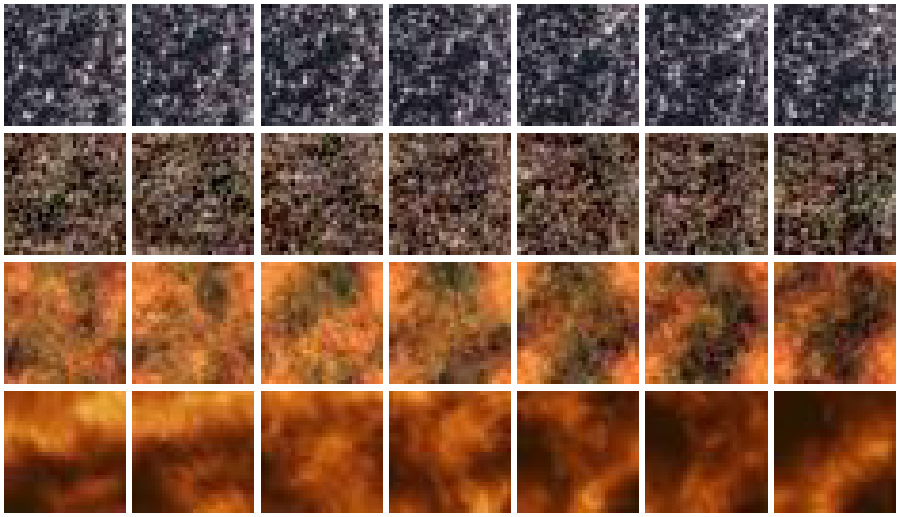

Fig. 2. Example of dynamic textures mixing. The first and last row correspond to $f^{[0]}$ and $f^{[1]}$ respectively, being the order of the frames from left to right. The central rows are instances of the interpolated dynamic texture models.

Table 2. Pseudocode for mixing several input exemplars

Input: exemplars $(f^{[i]})_{i\in I}$, weight $(\rho_i)_i$ with $\sum_{i\in I}\rho_i = 1$.
Output: realization h of the interpolated model $\mu^\star$ between $\mu_i = \mu(f^{[i]})$ for $i \in I$.
Preprocessing: for $i \in I$, apply the pre-processing step of Table 1.
Model mixing: for each ω, do
- $\forall\, i \in I$, compute $\hat{\Sigma}_i(\omega) = (\hat{f}_p^{[i]}(\omega))(\hat{f}_p^{[i]}(\omega))^* \in \mathbb{C}^{d\times d}$
- Initialize $\hat{\Sigma}^{(0)}(\omega) = 0 \in \mathbb{C}^{d\times d}$.
- Repeat until convergence $\hat{\Sigma}^{(k+1)}(\omega) = \Phi_\omega(\hat{\Sigma}^{(k)}(\omega))$ (see (9)), $k \leftarrow k+1$.
- Set $\hat{\Sigma}^\star(\omega) = \hat{\Sigma}^{(k)}(\omega)$. Compute the Cholesky factorization $\hat{\Sigma}^\star(\omega) = \hat{A}(\omega)\hat{A}(\omega)$ of $\hat{\Sigma}^\star(\omega)$.

Gaussian model synthesis:
- Compute $m^\star = \sum_{i\in I}\rho_i m_i \in \mathbb{R}^d$.
- Compute a realization $w \in \mathbb{R}^{N\times d}$ of $\mathcal{N}(0, \mathrm{Id}_{Nd})$.
- Compute the Fourier transform $\hat{w} \in \mathbb{R}^{N\times d}$ of w using FFT.
- $\forall\, \omega \neq 0$, compute $\hat{h}(\omega) = \hat{\Sigma}^\star(\omega)\hat{w}(\omega) \in \mathbb{C}^d$. Set $\hat{h}(0) = m^\star$.
- Compute $h \in \mathbb{R}^{N\times d}$ from $\hat{h}$ using the inverse FFT.

$\mu^\star = \mathcal{N}(m^\star, \Sigma^\star)$, where $m^\star = \sum_{i\in I}\rho_i m_i$ and the covariance matrix is solution of the fixed point equation $\Phi(\Sigma^\star) = \Sigma^\star$ where

$$\Phi(\Sigma) = \sum_{i\in I}\rho_i \left(\Sigma^{1/2}\Sigma_i\Sigma^{1/2}\right)^{1/2}. \tag{8}$$

This barycenter can be shown to be unique if one of the Σ_i is full rank [20]. We leave for future work the theoretical analysis of the uniqueness when all the covariances are rank-deficient.

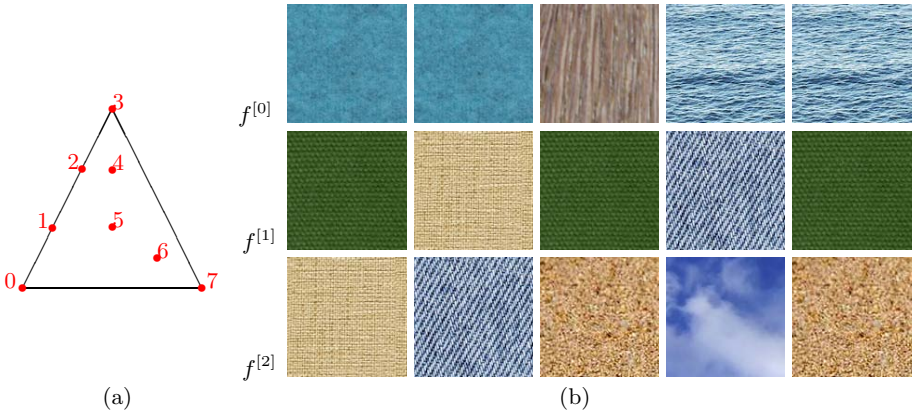

Fig. 3. (a) Spatial location scheme. (b) Each column corresponds to a single experiment, where $f^{[0]}, f^{[1]}, f^{[2]}$ are the original textures located at the vertices of the triangle in positions $0, 3, 7$, respectively. The other numbers correspond to the interpolated Gaussian models. Instances of all of these models can be observed in Figure 4 (a)-(e).

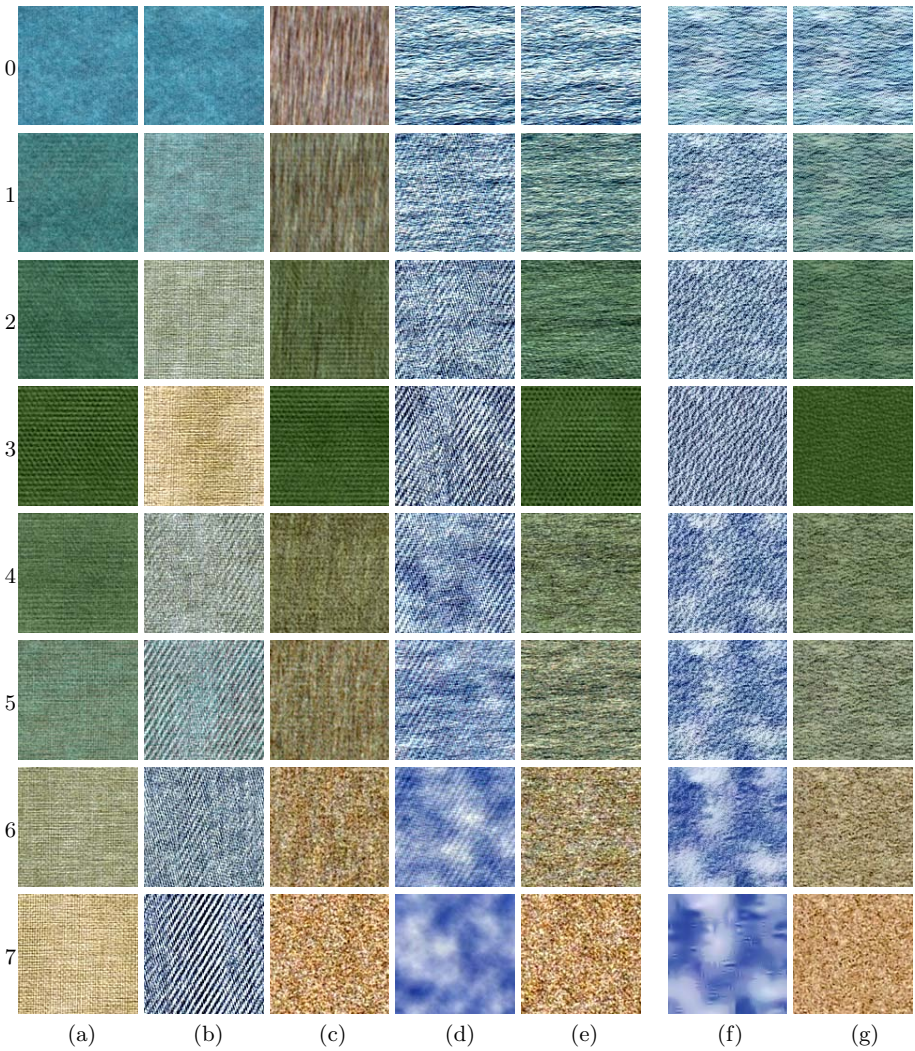

Fig. 4. Each column corresponds to a single experiment. The parameter $\rho = (\rho_1, \rho_2, \rho_3)$ of equation 7 is defined according to the triangle coordinates of the points in Figure 3(a). **(a)-(e)** Images obtained with the barycenter mix method whose input textures are shown as columns in Figure 3, respectively. **(f)(g)** Results obtained by the first method proposed by Rabin et al. [15].

4.2 Spot Noise Barycenter

When μ_i are Spot Noises, the covariance $\Sigma^\star$ of the barycenter is block diagonal over the Fourier domain, and the blocks $\hat{\Sigma}^\star(\omega)$ satisfy the fixed point equation $\hat{\Sigma}^\star(\omega) = \Phi_\omega(\hat{\Sigma}^\star(\omega))$ with

$$\Phi_\omega(\Sigma) = \sum_{i \in I} \rho_i \left(\Sigma^{1/2} \hat{\Sigma}_i(\omega) \Sigma^{1/2} \right)^{1/2}. \tag{9}$$

We note that in general, $\mu^\star$ is not Spot Noise because $\hat{\Sigma}^\star(\omega)$ is not necessarily rank one.

Numerical Computation. Following [21], we propose to compute $\hat{\Sigma}^\star(\omega)$ by iterating the mapping Φ_ω, i.e. compute the sequence $\hat{\Sigma}^{(k+1)}(\omega) = \Phi_\omega(\hat{\Sigma}^{(k)}(\omega))$. Although the mapping Φ_ω is not strictly contracting, we observe numerically the convergence $\hat{\Sigma}^{(k)}(\omega) \to \hat{\Sigma}^\star(\omega)$ when $k \to +\infty$. The numerical computation of Φ_ω in the case $d = 3$ requires the computation of the square root of 3×3 matrices, which is performed explicitly by computing the eigenvalue of the symmetric matrix as the root of a third order polynomial. The pseudocode of the method is detailed in Table 2.

4.3 Numerical Examples

Given three input textures, $f^{[0]}, f^{[1]}, f^{[2]}$, and the path defined in Figure 3(a) by the red numbers in increasing order, we generate the Gaussian models associated to each point. A realization of each of these models can be observed in Figure 4 (a)-(e) using as input textures the columns of Figure 3, respectively. Note how, as we approach an input model, the features of it tend to predominate in the synthesized texture and how the color and the texture patterns are smoothly interpolated along the geodesic path. We would also like to note that this method is also able to reproduce small periodic patterns. Finally, in Figure 4 (f) (g) we show the results obtained with the method by Rabin et al. [15], to be compared with the columns Figure 4 (d) (e), respectively.

5 Conclusion

We have presented a new method for texture mixing that enables the creation of new complex textures from a set of exemplars.

Given two texture models, we used the OT geodesic path over Gaussian distributions to interpolate new texture models. The numerical results show how the method is able to merge the visual features of the original images into new complex textures. We also generalized this OT geodesic method to the mixing of an arbitrary number of models using OT barycenters. We postpone for later research a thorough perceptual evaluation of the output textures.

References

1. Popat, K., Picard, R.W.: Novel cluster-based probability model for texture synthesis, classification, and compression. In: Visual Communications and Image Processing, pp. 756–768 (1993)

2. Efros, A.A., Leung, T.K.: Texture synthesis by non-parametric sampling. In: Proc. of ICCV 1999, p. 1033 (1999)
3. Wei, L.Y., Lefebvre, S., Kwatra, V., Turk, G.: State of the art in example-based texture synthesis. In: Eurographics 2009, State of the Art Report, EG-STAR. Eurographics Association (2009)
4. Portilla, J., Simoncelli, E.P.: A parametric texture model based on joint statistics of complex wavelet coefficients. Int. Journal of Computer Vision 40, 49–70 (2000)
5. Zhu, S.C., Wu, Y., Mumford, D.: Filters, random fields and maximum entropy (FRAME): Towards a unified theory for texture modeling. International Journal of Computer Vision 27, 107–126 (1998)
6. Kwatra, V., Schödl, A., Essa, I., Turk, G., Bobick, A.: Graphcut textures: image and video synthesis using graph cuts. ACM Trans. Graph. 22, 277–286 (2003)
7. Galerne, B., Gousseau, Y., Morel, J.M.: Random phase textures: Theory and synthesis. IEEE Transactions on Image Processing 20, 257–267 (2011)
8. van Wijk, J.J.: Spot noise texture synthesis for data visualization. In: Proceedings of the 18th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1991, pp. 309–318. ACM, New York (1991)
9. Wei, L.Y., Levoy, M.: Fast texture synthesis using tree-structured vector quantization. In: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2000, pp. 479–488. ACM Press/Addison-Wesley Publishing Co., New York (2000)
10. Doretto, G., Chiuso, A., Wu, Y., Soatto, S.: Dynamic textures. International Journal of Computer Vision 51, 91–109 (2003)
11. Doretto, G., Jones, E., Soatto, S.: Spatially homogeneous dynamic textures. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004, Part II. LNCS, vol. 3022, pp. 591–602. Springer, Heidelberg (2004)
12. Xia, G.S., Ferradans, S., Peyré, G., Aujol, J.F.: Compact representations of stationary dynamic textures. In: Proc. ICIP 2012 (2012)
13. Bar-Joseph, Z., El-Yaniv, R., Lischinski, D., Werman, M.: Texture mixing and texture movie synthesis using statistical learning. IEEE Tr. on Vis. and Comp. Graph. 7, 120–135 (2001)
14. Matusik, W., Zwicker, M., Durand, F.: Texture design using a simplicial complex of morphable textures. ACM Transactions on Graphics 24, 787–794 (2005)
15. Rabin, J., Peyré, G., Delon, J., Bernot, M.: Wasserstein barycenter and its application to texture mixing. In: Bruckstein, A.M., ter Haar Romeny, B.M., Bronstein, A.M., Bronstein, M.M. (eds.) SSVM 2011. LNCS, vol. 6667, pp. 435–446. Springer, Heidelberg (2012)
16. Moisan, L.: Periodic plus smooth image decomposition. Journal of Mathematical Imaging and Vision 39, 161–179 (2011)
17. Villani, C.: Topics in Optimal Transportation. American Mathematical Society (2003)
18. Dowson, D.C., Landau, B.V.: The fréchet distance between multivariate normal distributions. J. Multivariate Anal. 3, 450–455 (1982)
19. Takatsu, A.: Wasserstein geometry of gaussian measures. Osaka J. Math (2011)
20. Agueh, M., Carlier, G.: Barycenters in the wasserstein space. SIAM J. on Mathematical Analysis 43, 904–924 (2011)
21. Knott, M., Smith, C.S.: On a generalization of cyclic monotonicity and distances among random vectors. Linear Algebra and its Applications 199, 363–371 (1994)