A Texton for Fast and Flexible Gaussian Texture Synthesis

{Bruno.Galerne, Arthur.Leclaire, Lionel.Moisan}
@parisdescartes.fr

Laboratoire MAP5 — Université Paris Descartes

Introduction

Gaussian textures allow for

- **by-example synthesis** [4],
- **dynamic texture synthesis** [8],
- **texture mixing** [8].

The classical FFT-based synthesis algorithm has some limitations:
1) The underlying model is implicitly periodic;
2) It does not allow for local variations of the kernel or the grid.

We propose here

- to approximate a Gaussian texture by a **Discrete Spot Noise**
- to compute a **Synthesis-Oriented Texton** which can be used for DSN synthesis.

Gaussian textures can then be generated **on-demand** in a **faster, simpler, and more flexible way**.

Spot Noise Model

Let $h : \mathbb{Z}^2 \to \mathbb{R}^d$ be a function with finite support S_h and let us define $\tilde{h}(x) = h(-x)$.

The **Discrete Spot Noise (DSN)** on $\mathbb{Z}^2$ is

$$\forall \mathbf{x} \in \mathbb{Z}^2, \quad F_{\lambda,h}(\mathbf{x}) = \sum_{\mathbf{y}\in\mathbb{Z}^2} P_\lambda(\mathbf{y}) h(\mathbf{x}-\mathbf{y}) \,,$$

where P_λ is a Poisson white noise with intensity λ.

The **renormalized DSN**

$$G_{\lambda,h} = \frac{F_{\lambda,h} - \mathbb{E}(F_{\lambda,h})}{\sqrt{\lambda}} = \frac{1}{\sqrt{\lambda}}\Big(h * P_\lambda - \lambda \sum_{\mathbf{y}\in\mathbb{Z}^2} h(\mathbf{y})\Big)$$

has zero-mean and covariance function $h * \tilde{h}^T$, and

$$G_{\lambda,h} \xrightarrow[\lambda\to\infty]{(d)} \mathrm{ADSN}(h) = h * W$$

where W is a normalized Gaussian white noise on $\mathbb{Z}^2$.

One can also define a **circular ADSN** on a $M \times N$ rectangle Ω with periodic boundary conditions. **Circular convolutions** can then be computed using the FFT. In particular, we can compute the **L^2 optimal transport distance** between circular ADSN μ_0, μ_1 associated to h_0, h_1:

$$d_{OT}^2(\mu_0,\mu_1) = \sum_{\boldsymbol{\xi}\in\Omega} \Big(\|\widehat{h}_0\|^2 + \|\widehat{h}_1\|^2 - 2|\widehat{h}_0^*\widehat{h}_1| \Big)(\boldsymbol{\xi}).$$

We define the (squared) **relative model error**

$$\mathrm{RME}(h,h_0)^2 = \frac{\sum_{\boldsymbol{\xi}} \Big(\|\widehat{h}_0\|^2 + \|\widehat{h}\|^2 - |\widehat{h}_0^*\widehat{h}| \Big)(\boldsymbol{\xi})}{\sum_{\boldsymbol{\xi}} \|\widehat{h}_0\|^2(\boldsymbol{\xi})}.$$

Synthesis-Oriented Texton

A **Synthesis-Oriented Texton (SOT)** for the model $\mathrm{ADSN}(h)$ is any kernel k such that

- $\mathbf{Supp(k) \subset S}$ (with prescribed S)
- $\mathbf{k} * \tilde{\mathbf{k}}^\mathbf{T} \approx \mathbf{h} * \tilde{\mathbf{h}}^\mathbf{T}$,
- $\mathbf{G_{\lambda,k}} \overset{\text{(visual)}}{\approx} \mathbf{ADSN(k)}$ even for low λ.

Algorithm

Let $u : \Omega \to \mathbb{R}^d$ be an exemplar texture.

Empirical mean:

$$\bar{u} = \frac{1}{|\Omega|}\sum_{\mathbf{x}\in\Omega} u(\mathbf{x}) \,.$$

Circular autocorrelation:

$$c_u = t_u \odot \tilde{t}_u^T \,, \quad \text{where} \quad t_u = \frac{1}{\sqrt{|\Omega|}}(u - \bar{u}) \,.$$

The algorithm **alternates** between

$$q_S(h) = h\,\mathbf{1}_S \quad \textbf{(support projection)},$$

$$\widehat{p_{t_u}(h)} = \frac{\widehat{t}_u \widehat{t_u}^* \widehat{h}}{|\widehat{t_u}^* \widehat{h}|}\mathbf{1}_{\widehat{t}_u^*\widehat{h}\neq 0} \quad \textbf{(spectral projection)}.$$

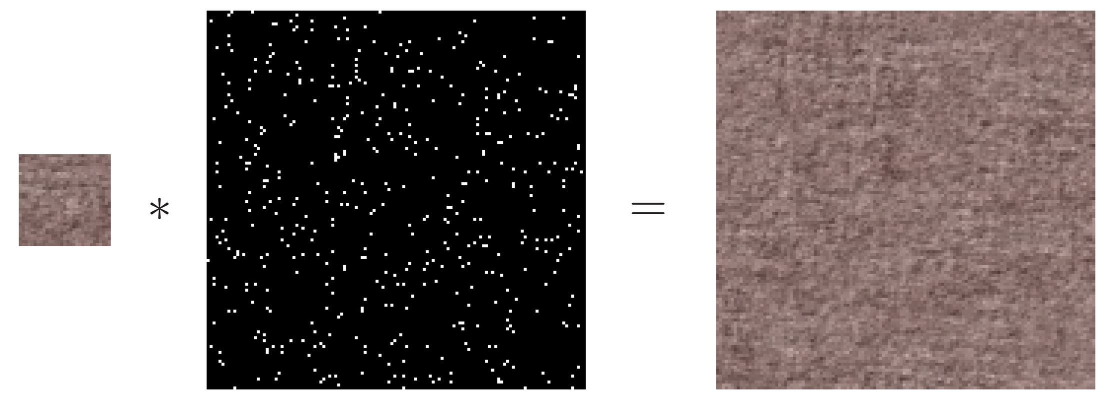
Algorithm: SOT computation

- Initialization: $\widehat{t} \leftarrow \widehat{t}_u e^{i\psi}$ where ψ is a uniform random phase function.
- Repeat (n times) $t \leftarrow q_S(p_{t_u}(t))$.

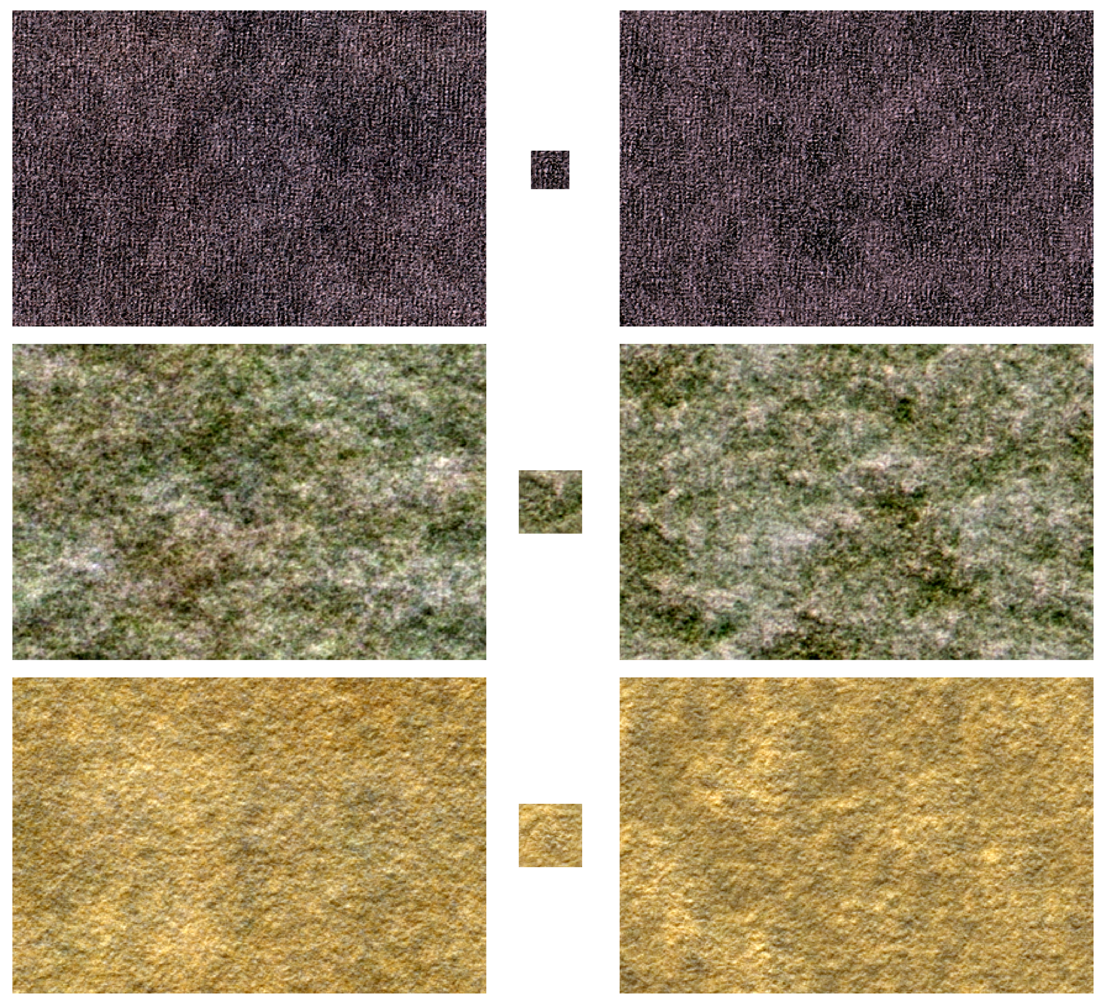
Convergence: Since p_{t_u} is not the projection on a convex set, the convergence is not proved. In practice, the iterates stabilize after 50 iterations.

Fast Gaussian Texture Synthesis

Using the SOT t, the **ADSN** $t_u * W$ can be approximated by the **DSN** $t * P_\lambda$ with low intensity λ, which can be computed by **direct summation** instead of a FFT-based algorithm. The mean complexity is then $\mathscr{O}(\lambda|\Omega|)$ instead of $\mathscr{O}(|\Omega|\log|\Omega|)$.

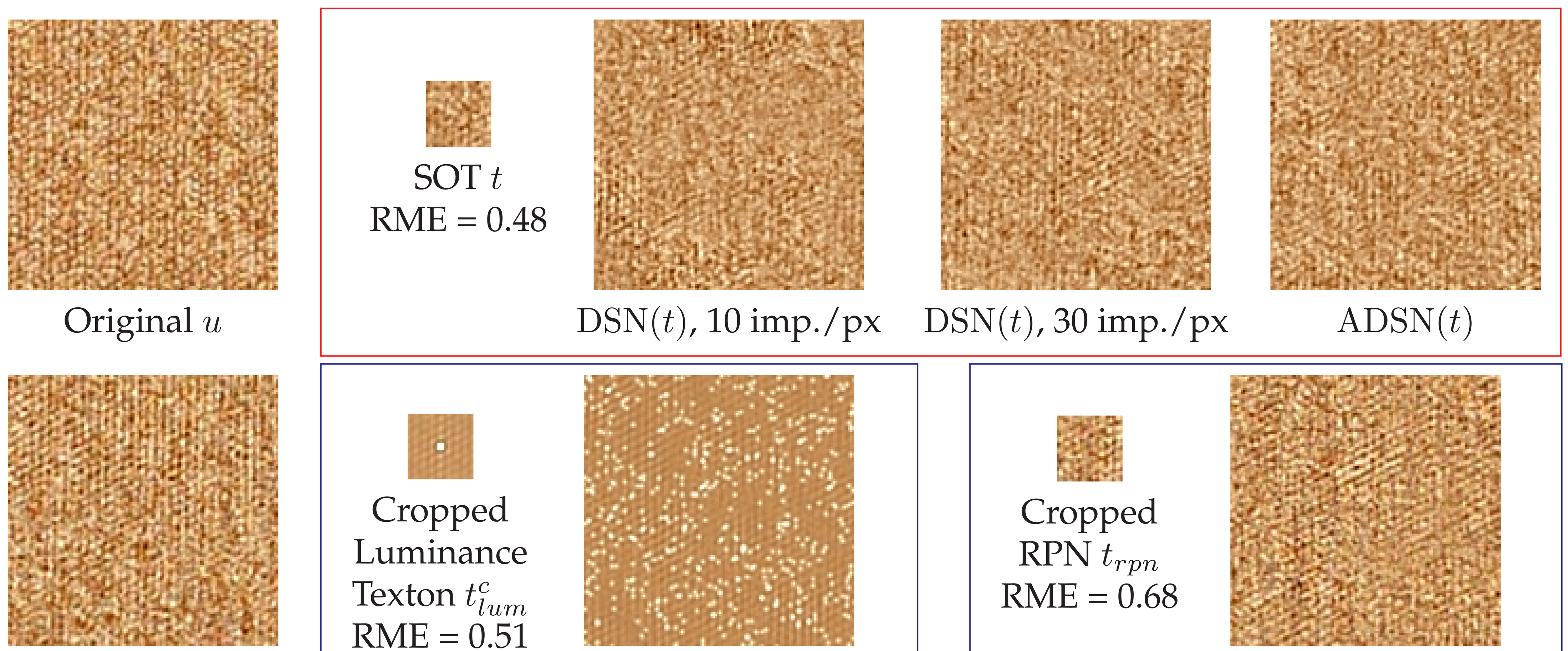


Spot noise synthesis at low intensity.



Exemplar — SOT — DSN (50 imp./pix.)

Comparison



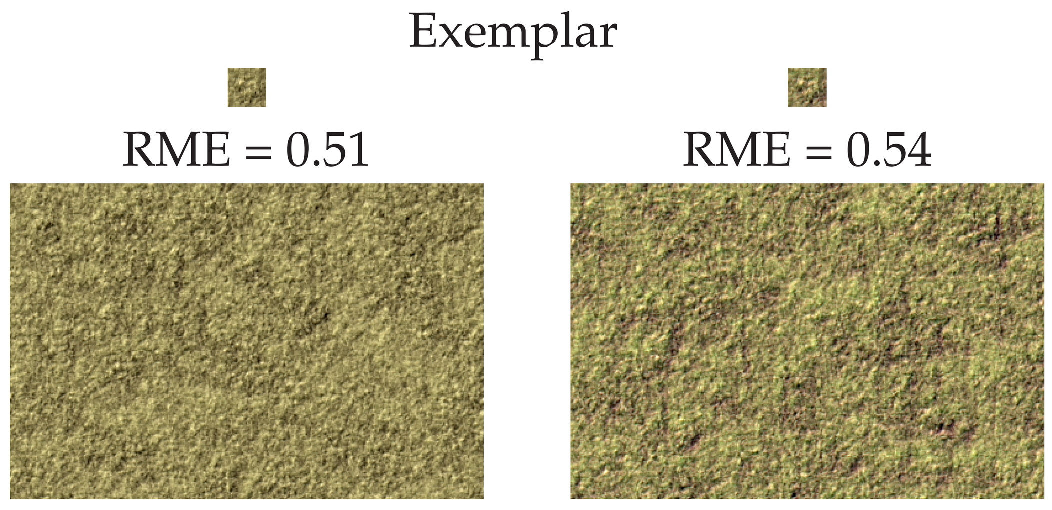
Original u — SOT t, RME = 0.48 — DSN(t), 10 imp./px — DSN(t), 30 imp./px — ADSN(t)

ADSN(t_u) — Cropped Luminance Texton t_{lum}^c, RME = 0.51 — DSN(t_{lum}^c), 30 imp./px — Cropped RPN t_{rpn}, RME = 0.68 — DSN(t_{rpn}), 30 imp./px

Color Correction

One can better **preserve the color distribution** by reimposing the color covariance of the exemplar texture (which amounts to apply a 3×3 transformation in the color space).

Exemplar



RME = 0.51 — RME = 0.54

Without color correction — With color correction

Conclusion

Given an exemplar texture image u, the proposed algorithm computes a synthesis-oriented texton having a **prescribed small support** and for which the associated **DSN is close to the Gaussian texture** associated with u, **even for a low intensity λ**. This SOT can be considered as an **inverse texture synthesis solution** [7] for the Gaussian model.

For an average number of **30 impacts per pixels**, the DSN associated with the SOT produces visually satisfying results, and is thus **more competitive than the spectral simulation** algorithm. The direct simulation of the DSN is simple and allows **parallel local evaluation** using standard computer graphics techniques for the Poisson process simulation [5] (a GPU implementation can **produce 80fps** for a 1024×1024 image on a CUDA server, using a DSN with 30 impacts per pixel).

An interesting perspective would be to extend this procedure to **a continuous framework** for **procedural texture synthesis**.

References

[1] A. Desolneux, L. Moisan, S. Ronsin, "A compact representation of random phase and Gaussian textures", *proc. ICASSP*, pp. 1381–1384, 2012.

[2] A. Desolneux, L. Moisan, S. Ronsin, "A texton for Random Phase and Gaussian textures", in preparation.

[3] J.R. Fienup, "Phase retrieval algorithms: a comparison", *Applied Optics* 21(15), pp. 2758–2769, 1982.

[4] B. Galerne, Y. Gousseau, J.-M. Morel, "Random Phase Textures: Theory and Synthesis", *IEEE Trans. on Image Processing* 20(1), pp. 257–267, 2011.

[5] B. Galerne, A. Lagae, S. Lefebvre, G. Drettakis, "Gabor noise by example", *proc. SIGGRAPH* 31(4), pp. 73:1-73:9, 2012.

[6] J.J. van Wijk, "Spot noise texture synthesis for data visualization", *proc. SIGGRAPH* 25, pp. 309–318, 1991.

[7] L.Y. Wei, J. Han, K. Zhou, H. Bao, B. Guo, H.Y. Shum, "Inverse texture synthesis", *ACM TOG* 27, 2008.

[8] G.-S. Xia, S. Ferradans, G. Peyré, J.-F. Aujol, "Synthesizing and Mixing Stationary Gaussian Texture Models", *SIAM J. Imaging Sci.* 8(1), pp. 476–508, 2014.

[9] `http://www.math-info.univ-paris5.fr/~aleclair/sot/`