

Procedural Noise using Sparse Gabor Convolution

Auxiliary Material

*Ares Lagae
Sylvain Lefebvre
George Drettakis
Philip Dutré*

Report CW545, May 2009



Katholieke Universiteit Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

Procedural Noise using Sparse Gabor Convolution

Auxiliary Material

*Ares Lagae
Sylvain Lefebvre
George Drettakis
Philip Dutré*

Report CW545, May 2009

Department of Computer Science, K.U.Leuven

Abstract

This technical report contains auxiliary material for the article
LAGAE, A., LEFEBVRE, S., DRETTAKIS, G., AND DUTRÉ, P. 2009.
Procedural noise using sparse Gabor convolution. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2009)* 28, 3.

Keywords : noise, procedural texture, rendering, shading.
CR Subject Classification : I.3.7.

Procedural Noise using Sparse Gabor Convolution Auxiliary Material

Ares Lagae* Sylvain Lefebvre†

Katholieke Universiteit Leuven*

George Drettakis† Philip Dutré*

REVES / INRIA Sophia-Antipolis†

Abstract

This technical report contains auxiliary material for the article LAGAE, A., LEFEBVRE, S., DRETTAKIS, G., AND DUTRÉ, P. 2009. Procedural noise using sparse Gabor convolution. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2009)* 28, 3.

1 Countour Lines of the Gabor Kernel

In section 2.3 of the paper we give the full width at half maximum of the Gaussian envelop of the Gabor kernel and the Gabor kernel in the frequency domain. In this section we provide equations for the contour lines of the Gabor kernel.

The contour of the Gaussian envelop for a value x is a circle with radius

$$\frac{1}{a} \sqrt{\frac{-\ln(x/K)}{\pi}}. \quad (1)$$

The full width at half maximum of the Gaussian envelop is the contour for the value $K/2$ multiplied by two, and can be approximated by $0.9394/a$ or $1/a$.

The contour of the Fourier transform of the Gabor kernel for a value x is a circle with radius

$$a \sqrt{\frac{-\ln(2a^2x/K)}{\pi}}. \quad (2)$$

The full width at half maximum of the Fourier transform of the Gabor kernel is the contour for the value $K/4a^2$ multiplied by two, and can be approximated by $0.9394a$ or a .

2 Approximation of the Power Spectrum of Bandlimited Isotropic Noise

In section 3.1 of the paper we approximate the power spectrum of bandlimited isotropic noise. In this section we provide details about this approximation.

The power spectrum of bandlimited isotropic noise in terms of radial frequency is

$$S_{NN}(f_r) = \lambda E[W^2] \frac{K^2}{2a^4} e^{-\frac{2\pi}{a^2}(f_r^2 + F_0^2)} \left[1 + I_0\left(\frac{4\pi F_0}{a^2} f_r\right) \right], \quad (3)$$

where I_0 is the modified Bessel function of the first kind. The asymptotic behavior of the modified Bessel function of the first kind is exponential, it can therefore be approximated by

$$I_0(x) \approx \frac{e^x}{\sqrt{2\pi x}}, \quad (4)$$

for large $|x|$ [Abramowitz and Stegun 1964, 9.7.1]. The power spectrum of the noise can therefore be approximated by

$$S_{NN}(f_r) \approx \lambda E[W^2] \frac{K^2}{4\sqrt{2\pi} \sqrt{f_r} \sqrt{F_0} a^3} e^{-\frac{2\pi}{a^2}(f_r - F_0)^2}, \quad (5)$$

for large $|4\pi F_0 f_r/a^2|$. The most interesting region in the power spectrum of bandlimited isotropic noise is the neighborhood of the



Figure 1: Two procedural textures generated using spatially varying noises, obtained by spatially varying the random sampling of the parameters of the Gabor kernels.

principal frequency F_0 . The power spectrum of the noise can therefore be approximated by

$$S_{NN}(f_r) \approx \lambda E[W^2] \frac{K^2}{4\sqrt{2\pi} F_0 a^3} e^{-\frac{2\pi}{a^2}(f_r - F_0)^2}, \quad (6)$$

for large $|4\pi F_0 f_r/a^2|$ and for $f_r \approx F_0$. Similar to the power spectrum of bandlimited anisotropic noise, this is a Gaussian with a principal frequency of F_0 and a width of a .

Note that we only use this approximation to show that the power spectrum of bandlimited isotropic noise has an intuitive shape related to the parameters of the Gabor kernel, and that we do not use it in computations.

The approximation of the power spectrum is illustrated in Figure 2 (page 3). The figure shows that the approximation is reasonably accurate, and only breaks down when the power spectrum has a significant contribution at zero frequency. However, in general this does not occur for bandlimited isotropic noise.

3 Number of Impulses

In the paper we do not illustrate the effect of the number of impulses on the resulting noise. In this section we illustrate this effect.

The effect of the number of impulses for bandlimited anisotropic and isotropic noise is illustrated in figure 3 (page 4) and figure 4 (page 4). The figures show that noise with a good quality is obtained starting from roughly 16 to 32 number of impulses, and that a higher number of impulses does not significantly improve the quality of the noise.

Although a theoretical method for choosing the number of impulses in terms of the desired noise quality would be interesting, most users will most likely empirically choose the number of impulses as a compromise between speed and quality.

4 Validation of the Procedural Evaluation

In section 4 of the paper we state that we have validated our procedural implementation by comparing the results to those of a non-procedural CPU implementation based on frequency domain filtering. In this section we provide details about the validation of our noise.

*e-mail: {ares.lagae, philip.dutre}@cs.kuleuven.be

†e-mail: {sylvain.lefebvre, george.drettakis}@sophia.inria.fr

We validate the procedural CPU and GPU implementations of our noise using a reference implementation. The reference implementation synthesizes noise by filtering white noise, similar to the anisotropic noise of Goldberg et al. This is done by multiplying white noise with the Fourier transform of the Gabor kernel, followed by an inverse Fourier transform. The noise produced by the reference implementation can be considered as the ground truth.

The validation is illustrated in Figure 5 (page 5) and figure 6 (page 5). These figures show that the noises produced by the procedural implementation and the reference implementation and their power spectrum are virtually indistinguishable. This means that the procedural evaluation does not introduce undesired artifacts, and that the number of impulses per kernel is sufficient.

5 Noise Analysis

Throughout the paper and in section 6 of the paper we perform noise analysis. In this section we provide details about noise analysis.

We estimate the power spectrum using Bartlett's method of averaging periodograms [Ulichney 1988]. The periodogram is a simple estimator for the power spectrum, defined as the magnitude squared of the Fourier transform [Press et al. 2002]. Averaging periodograms produces an estimator with a lower variance.

We obtain an estimate \tilde{S}_{NN} of the true power spectrum S_{NN} of the noise N by averaging T periodograms $|\mathcal{F}(N_i)|^2$ of different instances of the noise

$$\tilde{S}_{NN} = \frac{1}{T} \sum_{i=1}^T |\mathcal{F}(N_i)|^2. \quad (7)$$

We typically use 100 instances.

To our knowledge, previous work analyzes noise using the magnitude of the Fourier transform, which corresponds to a single periodogram. A single periodogram is typically noisy, which makes it difficult to distinguish features from noise, but averaging periodograms can average out features. This means that although Bartlett's method of averaging periodograms produces a better estimate for the power spectrum, a single periodogram can still be useful. For example, the horizontal striation artifacts in Perlin noise are better visible in the magnitude of the Fourier transform than in the estimate for the power spectrum.

We further simplify the estimate of the power spectrum of isotropic noise by radial averaging [Ulichney 1988].

Figure 7 (page 6) and figure 8 (page 6) are extended versions of figures 3 and 4 of the paper that also include the magnitude of the Fourier transform of the noise.

Figure 9 (page 7) is an extended version of figure 8 of the paper that also includes the intensity distribution, the magnitude of the Fourier transform, and the radial power spectrum of the noise.

6 Approximation Errors of a Weighted Sum of Bandlimited Noise

In the paper we state that our noise provides more accurate spectral control than previous noises. In this section we provide details about the approximation errors of a weighted sum of bandlimited noise.

A weighted sum of bandlimited noise offers limited spectral control because of the discretization of frequency space, and this can result in visible approximation errors. All noises that use a weighted sum to achieve spectral control, including Perlin noise, wavelet noise, the noise of Goldberg et al., and the noise of Kensler et al., are subject to this problem. Our noise is not subject to this problem because it does not discretize frequency space.

We illustrate the approximation errors of a weighted sum of bandlimited noise with the anisotropic noise of Goldberg et al. The approximation errors are illustrated in figure 10 (page 8) and figure 11 (page 8). These figures show an anisotropic and an isotropic target noise and the approximation of the target noise with the noise of Goldberg et al. using the least squares and the heuristic method for computing the weights. The target noise cannot be accurately reproduced because of approximation errors due to the discretization of frequency space, and the approximate nature of the heuristic method.

We illustrate that our noise is not subject to this problem by approximating the anisotropic noise of Goldberg et al. using our noise. The approximation is shown in figure 12 (page 8). Our noise can accurately approximate the noise of Goldberg et al.

7 Interference Patterns

In this section we provide details about interference patterns in our noise that might be mistaken for undesired artifacts.

When the bandwidth is very close to zero, our bandlimited anisotropic noise exhibits interference patterns. These interference patterns can be theoretically explained as follows. If the bandwidth is very close to zero, the noise in the spatial domain consist only of harmonics with frequencies very close to the principal frequency. These harmonics slowly transition between perfectly in phase, causing constructive interference, and perfectly out of phase, causing destructive interference. This results in interference patterns. If the bandwidth is zero, the noise consists of a single harmonic with a frequency equal to the principal frequency, and no interference patterns are visible. If the bandwidth is not very close to zero, no interference patterns are visible.

The interference patterns are illustrated in figure 13 (page 9). This figure shows anisotropic noise generated with the procedural implementation and the reference implementation, a non-procedural CPU implementation of our noise based on frequency domain filtering discussed in section 6 of the supplemental material. This figure shows that the interference patterns only occur when the bandwidth is very close to zero, but also that the interference patterns also occur with the reference implementation, confirming the theoretical explanation above. This means that the interference patterns are not undesired artifacts.

References

- ABRAMOWITZ, M., AND STEGUN, I. A. 1964. *Handbook of Mathematical Functions with Formulas, Graphs and Mathematical Tables*, tenth printing, december 1974, with corrections ed. Dover.
- PRESS, W. H., VETTERLING, W. T., TEUKOLSKY, S. A., AND FLANNERY, B. P. 2002. *Numerical Recipes in C++: the art of scientific computing*, 2nd ed. Cambridge University Press.
- ULICHNEY, R. 1988. Dithering with blue noise. *Proceedings of the IEEE* 76, 1, 56–79.

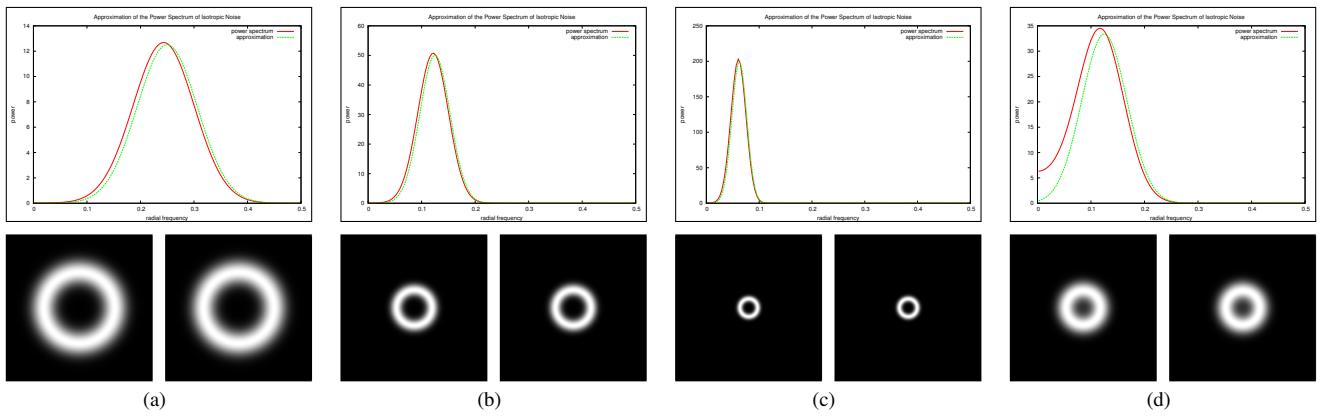


Figure 2: Approximation of the power spectrum of bandlimited isotropic noise. (a-c) The approximation is in general reasonably accurate. (d) The approximation breaks down when the power spectrum has a significant contribution at zero frequency. However, in general this does not occur for bandlimited isotropic noise. Note that we only use this approximation to show that the power spectrum of bandlimited isotropic noise has an intuitive shape related to the parameters of the Gabor kernel, and that we do not use it in computations.

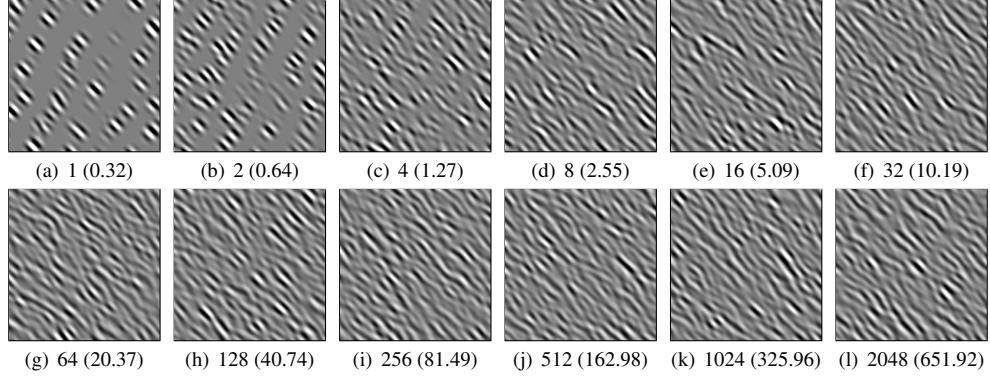


Figure 3: Effect of the number of impulses for bandlimited anisotropic noise. (a-l) Bandlimited anisotropic noise with an increasing number of impulses. The number of impulses per kernel and the corresponding number of impulses per cell is given in the caption of each subfigure. Note that the noises look very similar starting from roughly 16 to 32 impulses per kernel. We typically use 25 to 50 impulses per kernel in our GPU implementation and 100 impulses per kernel in our CPU implementation.

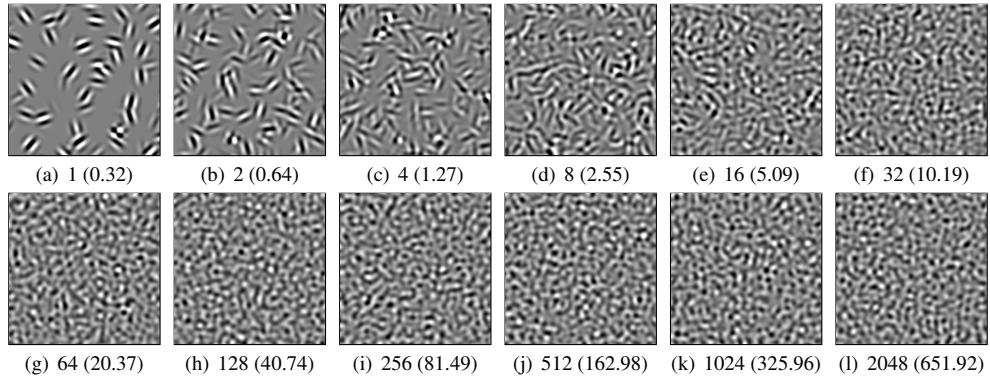


Figure 4: Effect of the number of impulses for bandlimited isotropic noise. (a-l) Bandlimited isotropic noise with an increasing number of impulses. The number of impulses per kernel and the corresponding number of impulses per cell is given in the caption of each subfigure. Note that the noises look very similar starting from roughly 16 to 32 impulses per kernel. We typically use 25 to 50 impulses per kernel in our GPU implementation and 100 impulses per kernel in our CPU implementation.

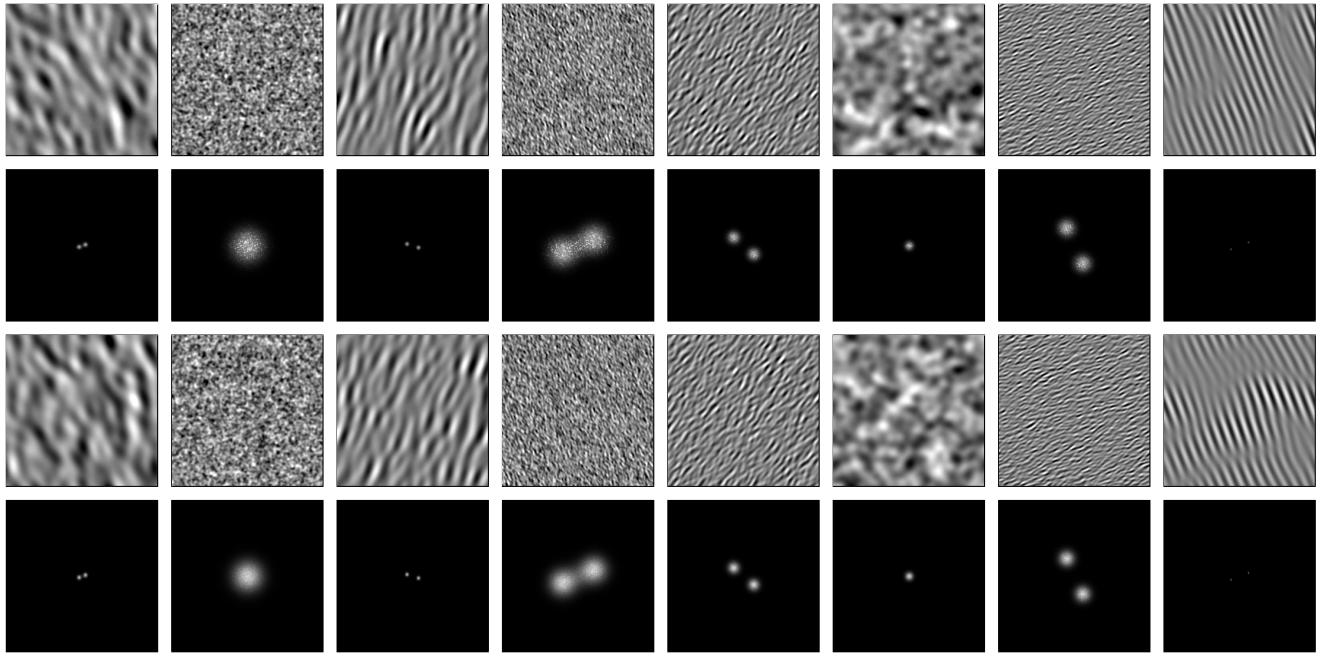


Figure 5: Validation of bandlimited anisotropic noise. Bandlimited anisotropic noise and corresponding power spectrum generated with (rows 1 and 2) the procedural implementation and (rows 3 and 4) the validation implementation. Every column corresponds to a different set of parameters. The noises produced by both implementations and their power spectra are virtually indistinguishable.

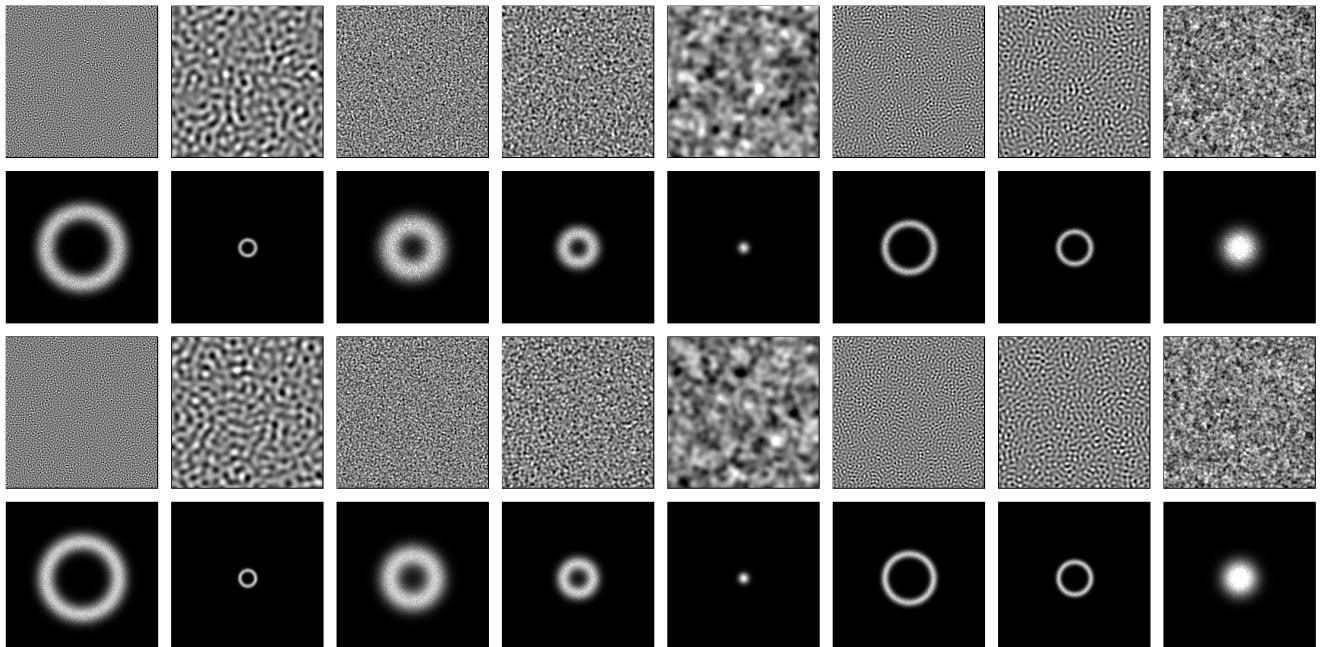


Figure 6: Validation of bandlimited isotropic noise. Bandlimited isotropic noise and corresponding power spectrum generated with (rows 1 and 2) the procedural implementation and (rows 3 and 4) the validation implementation. Every column corresponds to a different set of parameters. The noises produced by both implementations and their power spectra are virtually indistinguishable.

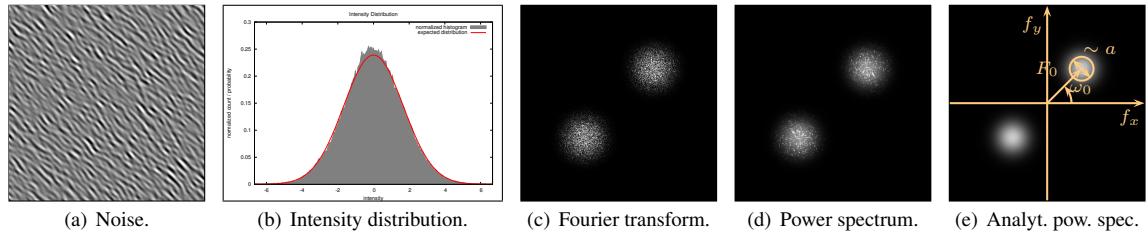


Figure 7: Bandlimited anisotropic noise. (a) Bandlimited anisotropic noise. (b) intensity histogram (gray) and expected intensity distribution (red). (b) Fourier transform. (c) Power spectrum. (d) Analytical power spectrum.

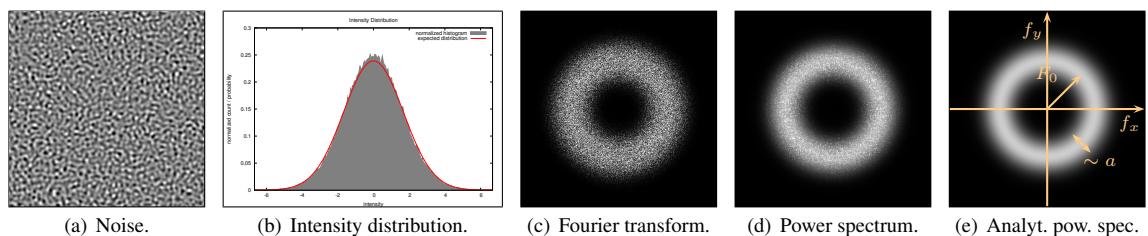


Figure 8: Bandlimited isotropic noise. (a) Bandlimited isotropic noise. (b) intensity histogram (gray) and expected intensity distribution (red). (b) Fourier transform. (c) Power spectrum. (d) Analytical power spectrum.

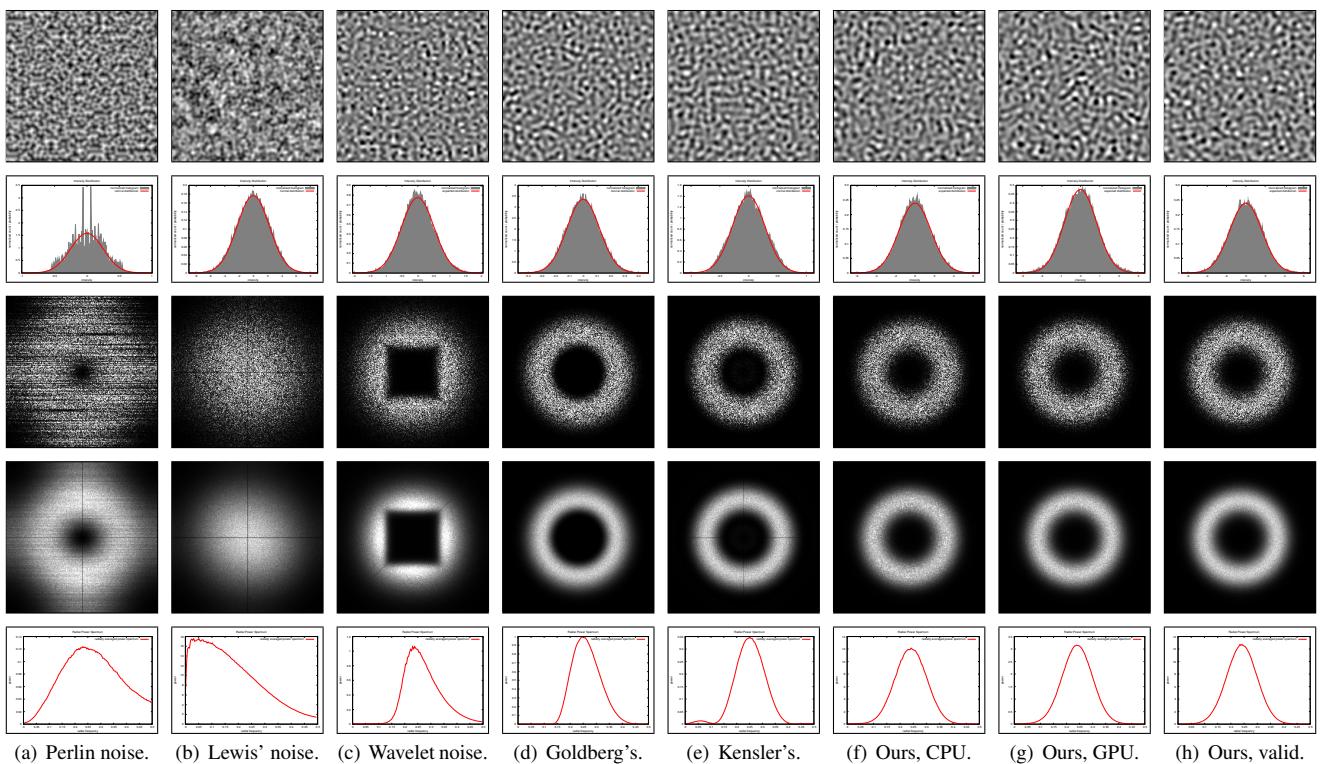


Figure 9: Comparison of isotropic noises. (a) Perlin noise. (b) Sparse convolution noise. (c) Wavelet noise. (d) The noise of Goldberg et al. (e) The noise of Kensler et al. (f) Our noise, CPU implementation. (g) Our noise, GPU implementation. (h) Our noise, validation. (row 1) Noise. (row 2) Intensity distribution. (row 3) Fourier transform. (row 4) Power spectrum. (row 5) Radial power spectrum.

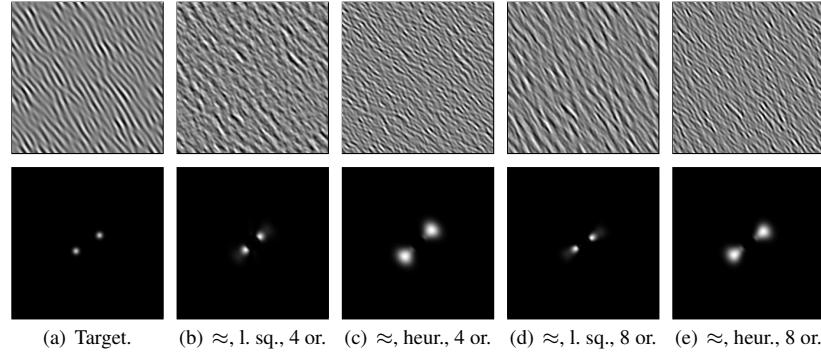


Figure 10: Approximation errors of Goldberg et al., anisotropic case. (a) Target noise. (b-e) Approximations of (a) with the noise of Goldberg et al. using (b) the least squares method and 4 orientations, (c) the heuristic method and 4 orientations, (d) the least squares method and 8 orientations, and (e) the heuristic method and 8 orientations. The target noise cannot be accurately reproduced with the noise of Goldberg et al. because of discretization errors due to the discretization of the frequency domain, and approximation errors made by the heuristic method.

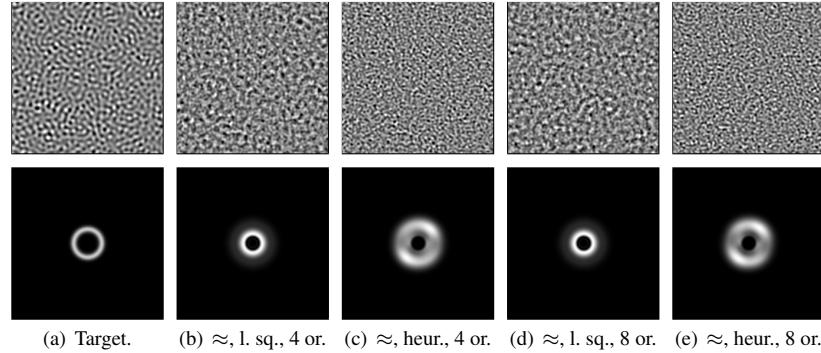


Figure 11: Approximation errors of Goldberg et al., isotropic case. (a) Target noise. (b-e) Approximations of (a) with the noise of Goldberg et al. using (b) the least squares method and 4 orientations, (c) the heuristic method and 4 orientations, (d) the least squares method and 8 orientations, and (e) the heuristic method and 8 orientations. The target noise cannot be accurately reproduced with the noise of Goldberg et al. because of discretization errors due to the discretization of the frequency domain, and approximation errors made by the heuristic method.

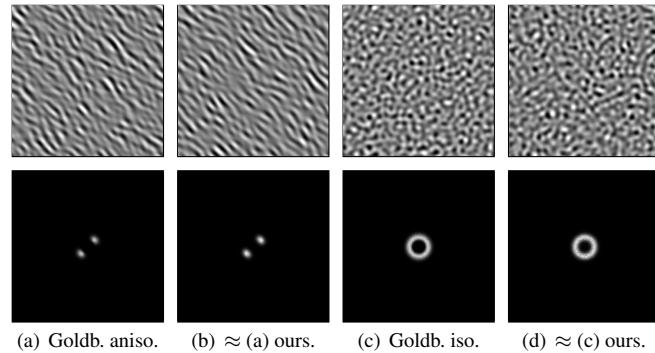


Figure 12: Approximation of the noise of Goldberg et al. with our noise. (a) Anisotropic noise of Goldberg et al. (b) Approximation of (a) with our noise using noise controllable bandlimits. (c) Isotropic noise of Goldberg et al. (d) Approximation of (c) with our noise using bandlimited isotropic noise. The noises and their power spectra are virtually indistinguishable.

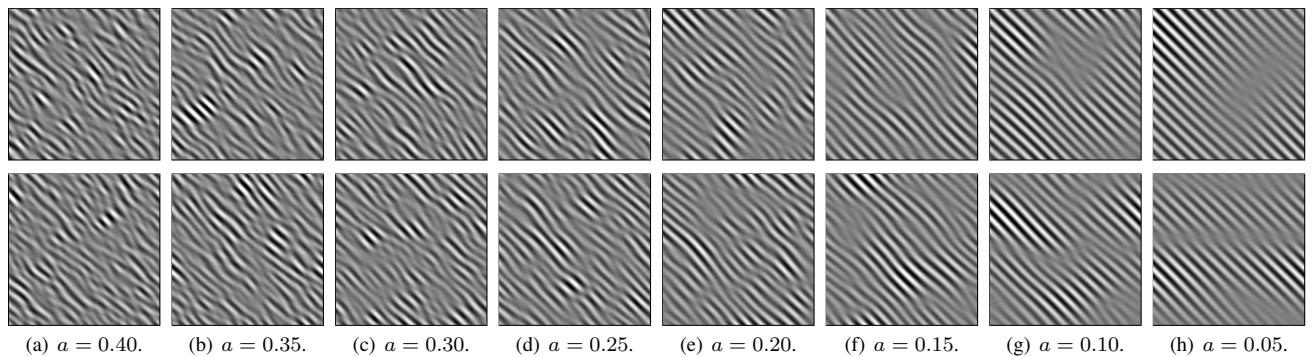


Figure 13: Interference patterns anisotropic noise. (a-h) Bandlimited anisotropic noise with a decreasing bandwidth. The bandwidth is given in the caption of each subfigure. (top row) Procedural implementation. (bottom row) reference implementation. The interference patterns are only visible for a bandwidth very close to zero. The interference patterns are also visible in the reference implementation.