USC–SIPI REPORT #259

An Improved Method for Two Dimensional Self-Similar Image Synthesis

Lance M. Kaplan and C.-C. Jay Kuo

June 1994

Signal and Image Processing Institute
UNIVERSITY OF SOUTHERN CALIFORNIA
Department of Electrical Engineering-Systems
3740 McClintock Avenue, Room 404
Los Angeles, CA 90089-2564 U.S.A.

An Improved Method for Two Dimensional Self-Similar Image Synthesis*

Lance M. Kaplan† and C.-C. Jay Kuo†

June 15, 1994; EDICS: IP 3.4

Abstract

In this work, we propose a new method called incremental Fourier synthesis to generate 2D self-similar images based on a 2D fBm model. With this method, the stationary increments of fBm are created by a Fourier synthesis method and the increments are added up to generate the nonstationary 2D fBm process. Since the new method takes advantage of the FFT, its computational complexity is only $O(N^2 \log_2(N))$, and its memory requirement is only $O(N^2)$ for a self-similar image of size $N \times N$.

1 Introduction

Fractional Brownian motion (fBm) is a useful stochastic model for describing many natural phenomena with a self-similar property [7]. In computer graphic applications, the generation of 2D fBm realizations is used to create natural looking landscapes and clouds [6], [8]. For example, the artist can create natural landscapes by considering the value of 2D fBm as an elevation value and use ray tracing techniques to draw the 2D surface. Another example is that the artist uses the values of 2D fBm as gray scale values to generate cloudy looking textures. Furthermore, the artist can consider the values of 2D fBm to represent certain colors to generate natural looking contour maps or coastlines.

The statistical self-similar property of fBm is a key to the natural looking textured surface where the "roughness" of the surface is invariant to the scale. For some computer graphic applications, the *exact* roughness-invariant property is not crucial in generating naturally appearing images and, as a result, methods loosely based on the statistics of fBm can be used to generate images with an approximate self-similar property. The two most common methods are midpoint displacement and Fourier synthesis [8]. Other methods include more sophisticated variants of midpoint displacement and linear filtering [2], [5], [9]. However, for some other applications, it is important for the textures to be synthesized based on the exact statistics of fBm. For instance, the artist may want the roughness-invariant property of the texture to hold as a user zooms into a texture. In other cases, one needs to use 2D fBm images as test images. One such example is that true fBm images are useful to test algorithms which measure the fractal dimension.

The midpoint displacement is the fastest fBm synthesis method with a complexity of $O(N^2)$, where N^2 is the size of the image. The Fourier synthesis method tries to recreate the $1/f$ spectrum via fast Fourier

*This work was supported by the National Science Foundation Presidential Faculty Fellow (PFF) Award ASC-9350309.

†The authors are with the Signal and Image Processing Institute and the Department of Electrical Engineering-Systems, University of Southern California, Los Angeles, California 90089-2564. E-mail: lancekap@sipi.usc.edu and cckuo@sipi.usc.edu.

transform (FFT) with a computational complexity of $O(N^2 \log_2(N))$. However, the main drawback of these methods is that they fail to create statistical self-similar processes so that the persistence of fBm is not reflected properly in every sample of the realization. The linear filtering method is also unable to capture the persistence of fBm entirely.

When the generation of true fBm is important, one could resort to brute force methods. Hoefer [3] *et. al* used the Cholesky decomposition to generate 2D fBm realizations. This algorithm, however, has a complexity of $O(N^6)$ to generate an image of size $N \times N$. Additionally, the Cholesky method requires that one stores an $N^2 \times N^2$ matrix in memory. Thus, the memory requirement is $O(N^4)$. By using the fact that the second order increments of 2D fBm are stationary and the correlation matrix of the increments can be organized in a block Toeplitz matrix with Toeplitz blocks (i.e. BTTB matrix), an $O(N^5)$ algorithm using a 2D generalization of Levinson recursion is possible [1]. The memory requirement of the 2D Levinson method is $O(N^3)$. Obviously, the above two fBm synthesis methods using exact statistics are expensive both in computation and in memory.

In this work, we propose a new method called incremental Fourier synthesis to generate 2D fBm where the stationary increments of fBm are created by a Fourier synthesis method. The increments are added up to generate the nonstationary 2D fBm process. Since the new method takes advantage of the FFT, its computational complexity is only $O(N^2 \log_2(N))$, and its memory requirement is only $O(N^2)$. Moreover, the method uses statistics which are as close as possible to the exact fBm statistics.

2 Fractional Brownian Motion

FBm is a random process that was popularized by Mandelbrot and Van Ness [7]. In this paper, we consider the straight forward generalization on 1D fBm signals to 2D fBm images. A 2D fBm process is a mean zero Gaussian process $B(t_x, t_y)$ satisfying

$$B(0,0) = 0, \tag{2.1}$$

$$\mathrm{VAR}[B(t_x + r_x, t_y + r_y) - B(t_x, t_y)] = f(r_x, r_y)\sigma^2, \tag{2.2}$$

where $\sigma^2 = \mathrm{VAR}[B(s_x + 1, s_y) - B(s_x, s_y)]$,

$$f(r_x, r_y) = \tilde{f}(\|(r_x, r_y)^T\|^{2H}) = (\sqrt{r_x^2 + r_y^2})^{2H}, \tag{2.3}$$

and $0 < H < 1$. Equation (2.2) is called the 2D self-similarity condition. It implies that the variance of any increments is independent of orientation and dependent only on the length of the increment. The parameter H is known as the Hurst parameter, and each realization of 2D fBm is a fractal with a dimension of $D = 3 - H$ [8]. In this work, we want to generate samples of 2D fBm on a discrete grid.

One can show from (2.1) and (2.2) that 2D fBm is a nonstationary isotropic process whose correlation function for a sampling period of Δx in both x and y directions is

$$\begin{aligned} r_B(m_x, m_y; n_x, n_y) &= E[B(\Delta x m_x, \Delta x m_y) B(\Delta x n_x, \Delta x n_y)] \\ &= \frac{\sigma^2}{2} |\Delta x|^{2H} [f(m_x, m_y) + f(n_x, n_y) - f(n_x - m_x, n_y - m_y)]. \end{aligned} \tag{2.4}$$

Note that the shape of the correlation of discrete 2D fBm is invariant to the chosen sampling rate or scale. This form of scale invariance is a direct result of the self-similarity condition. As a result, the textured appearance of 2D fBm sampled every Δx units and scaled by a factor or $\|\Delta x\|^{2H}$ is identical to the appearance of fBm sampled every one unit. Without loss of generality, we set $\Delta x = 1$. We define the 1st-order discrete increments of 2D fBm as

$$I_x(m_x, m_y) = B(m_x + 1, m_y) - B(m_x, m_y), \text{ and } I_y(m_x, m_y) = B(m_x, m_y + 1) - B(m_x, m_y).$$

The 2nd-order increments of 2D fBm is defined as

$$\begin{aligned} I_2(m_x, m_y) &= I_x(m_x, m_y + 1) - I_x(m_x, m_y) \\ &= I_y(m_x + 1, m_y) - I_y(m_x, m_y) \\ &= B(m_x + 1, m_y + 1) + B(m_x, m_y) - B(m_x + 1, m_y) - B(m_x, m_y + 1). \end{aligned} \tag{2.5}$$

Note that samples of fBm $B(m_x, m_y)$ can be calculated for $m_x \geq 0$ and $m_y \geq 0$ by using the values of the 2nd-order increments and the 1st-order increments along either the x or the y axis. The 1st- and 2nd-order increments are stationary. The correlation functions of these increments are

$$r_x(m_x, m_y) = \frac{\sigma^2}{2}[f(m_x + 1, m_y) + f(m_x - 1, m_y) - 2f(m_x, m_y)] \tag{2.6}$$

$$r_y(m_x, m_y) = \frac{\sigma^2}{2} f(m_x, m_y + 1) + f(m_x, m_y - 1) - 2f(m_x, m_y)] \tag{2.7}$$

$$\begin{aligned} r_2(m_x, m_y) &= \frac{\sigma^2}{2}[2(f(m_x + 1, m_y) + f(m_x - 1, m_y) + f(m_x, m_y + 1) + f(m_x, m_y - 1)) \\ &\quad -(f(m_x + 1, m_y + 1) + f(m_x + 1, m_y - 1) + f(m_x - 1, m_y + 1) \\ &\quad +f(m_x - 1, m_y - 1)) - 4f(m_x, m_y)]. \end{aligned} \tag{2.8}$$

The above correlations are discrete functions. Their Fourier transforms are periodic and it is in general difficult to compute these transforms analytically. However, by treating each correlation to be a continuous function, we can compute the Fourier transform more easily and observe some important properties of the actual periodic spectrum. By using the fact [4] that

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \sqrt{t_x^2 + t_y^2}^{2H} e^{-j[t_x\omega_x + t_y\omega_y]} dt_x dt_y = -\frac{2\sqrt{\pi}\Gamma(2H + 2)\sin(\pi H)}{\sqrt{\omega_x^2 + \omega_y^2}^{2H+2}},$$

and the "shifting" property of the Fourier transform, one can derive the Fourier transform of (2.8) (when integers m_x and m_y are replaced by real variables t_x and t_y) to be

$$S_2(\omega_x, \omega_y) = \frac{32\sqrt{\pi}\sin^2(\omega_x/2)\sin^2(\omega_y/2)\Gamma(2H + 2)\sin(\pi H)}{\sqrt{\omega_x^2 + \omega_y^2}^{2H+2}}. \tag{2.9}$$

One must consider an aliased version of (2.9) due to the sampling of the 2nd order increment. As ω_x and ω_y go to zero at the same rate, (2.9) can be written as

$$S_2(\omega, \omega) \approx C\frac{\omega^4}{\omega^{2H+2}}$$

and since $0 < H < 1$, one can say that the spectrum of the continuous increments has a value of zero at the origin (i.e. $\omega_x = \omega_y = 0$). Moreover,

$$S_2(2\pi k_x, 2\pi k_y) = 0, \qquad \forall k_x, k_y \in \mathbf{Z}. \tag{2.10}$$

Equations (2.9) and (2.10) indicate that the periodic spectrum of the sampled increments (sampled at intervals of one unit) has values of zeros whenever $\omega_x = 0$ or $\omega_y = 0$ and does not approach zero at any other points. The importance of this spectral property will be seen in Section 4.

3 Periodic Random Fields

Consider a periodic stationary Gaussian random field whose correlation function satisfies

$$R(m_x + kN, m_y + lN) = R(m_x, m_y), \qquad \forall k, l \in \mathbf{Z}. \tag{3.1}$$

Each realization of this random field is also periodic with a period of N in both the x and y directions. Thus, it is only necessary to know the values of the field over an $N \times N$ lattice of points, and the correlation function only needs to be considered for time lags which lie on an $N \times N$ grid. Due to symmetry of the correlation function about the origin, we have

$$R(m_x, m_y) = R(N - m_x, N - m_y). \tag{3.2}$$

For cases where the correlation function is symmetric around both x and y axes, i.e.

$$R(m_x, m_y) = R(N - m_x, m_y), \text{ and } R(m_x, m_y) = R(m_x, N - m_y), \tag{3.3}$$

the correlation function can be uniquely defined by lags with $m_x, m_y = 0, \ldots, N/2$. The other values of the correlation function can be determined through (3.2) and (3.3).

The importance of periodic random fields is due to the fact that 2D DFT is the Karhunen-Loève transform (KLT) for such fields. A nice result of this property is that realizations of periodic random fields are easy to generate because one just needs to scale white noise by the square-root of the field's power spectrum and then calculate the inverse 2D DFT. In fact, this generation procedure is used in normal Fourier synthesis of fBm where the power spectrum is assumed to be

$$\hat{R}(k_x, k_y) = C/\sqrt{k_x^2 + k_y^2}^{2H+2} \qquad \forall k_x, k_y = 0, \ldots, N/2. \tag{3.4}$$

The other values of the power spectrum are determined by symmetrically expanding $\hat{R}(k_x, k_y)$. Usually, the first $N/2 \times N/2$ values of the generated field are taken as the fBm image in order to avoid artifacts from the periodicity of the field.

4 Incremental Fourier Synthesis

The idea to create samples of fBm over an $(M+1) \times (M+1)$ grid is to generate the stationary increments over an $M \times M$ grid. We attempt to create periodic random fields of size $N \times N$ (where $N = 2M$)

whose correlation function $R(m_x, m_y)$ matches the correlation function of the nonperiodic increments for $(m_x, m_y) \in [-M, M] \times [-M, M]$. The other values of the correlation function for the periodic field can be determined via symmetries. Then the increments can be synthesized by using the corresponding power spectrum to scale white noise.

Before, we describe the new synthesis algorithm, it is worthwhile to point out two issues. One problem to consider is that the target periodic correlation function may not be positive definite. This could happen because periodically extending a time limited correlation function is equivalent to convolving the actual power spectrum by a sinc and then sampling to create the power spectrum of the periodic random field. Due to the Gibbs phenomenon, some of the values of the DFT of the periodic correlation function may be negative. By considering these bad values to be zero, we create the actual power spectrum which generates the increments. Because the negative values will occur near frequencies where the original power spectrum is zero, the difference between actual and target correlation functions will be small. Another point to consider is that the 1st and 2nd order increments cannot be generated independently or else major creasing will appear. The dependence of $I_2(m_x, m_y)$, $I_x(m_x, m_y)$, and $I_y(m_x, m_y)$ is due to (2.5) where the subtraction is taken modulo N. In fact, by taking the DFT of (2.5), we see that the DFT coefficients of the 1st order increments when the frequencies are nonzero (i.e. $k_x > 0$, $k_y > 0$) are completely determined by the corresponding DFT coefficients of the 2nd order increments. Moreover, the DFT coefficients for $I_2(m_x, m_y)$ must be zero for the zero frequencies (i.e. $k_x = 0$ or $k_y = 0$), and thus the actual power spectrum of $I_2(m_x, m_y)$ is forced to zero at the zero frequencies. Because the power spectrum of the nonperiodic field is zero at the zero frequencies (see Section 2), the change to the power spectrum will not greatly affect the difference between the actual and target correlation functions. Now, we describe the new algorithm in detail below.

Algorithm: Incremental Fourier Synthesis Method

1. Create white noise processes such that for $k_x = 0, \ldots, N$, and $k_y = 0, \ldots, N/2$, $\hat{W}(k_x, k_y) \sim N(0, 1)$, $\hat{\phi}(k_x, k_y) \sim \text{Uniform}[0, 2\pi)$, and $\phi(0,0) = \phi(N/2, N/2) = \phi(N/2, 0) = \phi(0, N/2) = 0$.
2. Calculate $R_2(m_x, m_y)$ (the desired correlation function of $I_2(m_x, m_y)$) by (2.8) for $m_x, m_y = 0, \ldots, N/2$, and symmetrically expand the correlation function via (3.2) and (3.3).
3. Calculate the power spectrum by

$$\hat{R}_2(k_x, k_y) = \sum_{m_x=0}^{N-1} \sum_{m_y=0}^{N-1} R_2(m_x, m_y) e^{-j\frac{2\pi k_x m_x}{N}} e^{-j\frac{2\pi k_y m_y}{N}}. \tag{4.1}$$

4. Define the actual positive semidefinite power spectrum as

$$\hat{S}_2(k_x, k_y) = \begin{cases} 0, & k_x = 0 \text{ or } k_y = 0, \\ 0, & \text{if } \hat{R}_2(k_x, k_y) < 0, \\ \hat{R}_2(k_x, k_y), & \text{otherwise.} \end{cases}$$

5. Synthesize the DFT coefficients of $I_2(m_x, m_y)$:

$$\hat{I}_2(k_x, k_y) = \begin{cases} N\sqrt{\hat{S}_2(k_x, k_y)}\hat{W}(k_x, k_y) e^{j\hat{\phi}(k_x, k_y)}, & \text{for } k_x = 0, \ldots, N-1 \text{ and } k_y = 0, \ldots, N/2, \\ \hat{I}_2^*(N - k_x, N - k_y), & \text{for } k_x = 0, \ldots, N-1 \text{ and } k_y = N/2+1, \ldots, N-1. \end{cases}$$

6. Calculate the 2nd order increments for $m_x, m_y = 0, \ldots, M-1$:

$$I_2(m_x, m_y) = \frac{1}{N^2} \sum_{k_x=0}^{N-1} \sum_{k_y=0}^{N-1} \hat{I}_2(k_x, k_y) e^{j\frac{2\pi k_x m_x}{N}} e^{j\frac{2\pi k_y m_y}{N}}. \tag{4.2}$$

7. Create white noise processes such that for $k_x, k_y = 0, \ldots, N/2$, $W_x(k_x) \sim N(0,1)$, $W_y(k_y) \sim N(0,1)$, $\phi_x(k_x) \sim \text{Uniform}[0, 2\pi)$, $\phi_y(k_y) \sim \text{Uniform}[0, 2\pi)$, and $\phi_x(0) = \phi_y(0) = \phi_x(N/2) = \phi_y(N/2)$.
8. Calculate $R_x(k_x, k_y)$ and $R_y(k_x, k_y)$ for $k_x, k_y = 0, \ldots, N/2$ using (2.6) and (2.7). Symmetrically expand the correlation functions using (3.2) and (3.3).
9. Compute the desired power spectrum of the 1st order increments at the zero frequencies via

$$\hat{R}_x(k_x, 0) = \sum_{m_x=0}^{N-1} \sum_{m_y=0}^{N-1} R_x(m_x, m_y) e^{-j\frac{2\pi k_x m_x}{N}}, \tag{4.3}$$

$$\hat{R}_y(0, k_y) = \sum_{m_x=0}^{N-1} \sum_{m_y=0}^{N-1} R_y(m_x, m_y) e^{-j\frac{2\pi k_y m_y}{N}}. \tag{4.4}$$

10. Define the actual positive semidefinite power spectrum of the 1st order increments at the zero frequencies via

$$\hat{S}_x(k_x, 0) = \begin{cases} 0, & \text{if } \hat{R}_x(k_x, 0) < 0, \\ \hat{R}_x(k_x, 0), & \text{otherwise,} \end{cases}$$

$$\hat{S}_y(0, k_y) = \begin{cases} 0, & \text{if } \hat{R}_y(0, k_y) < 0, \\ \hat{R}_x(0, k_y), & \text{otherwise.} \end{cases}$$

11. Synthesize the DFT coefficients of the the 1st order increments:

$$\hat{I}_x(k_x, k_y) = \begin{cases} -j\frac{\hat{I}_2(k_x, k_y)}{2\sin(2\pi k_y/N)} e^{-j2\pi k_y/N}, & \text{for } k_x, k_y = 1, \ldots, N-1, \\ N\sqrt{\hat{S}_x(k_x, 0)}\hat{W}_x(k_x) e^{j\phi_x(k_x)}, & \text{for } k_x = 0, \ldots, N/2, \text{ and } k_y = 0, \\ \hat{I}_x^*(N - k_x, 0), & \text{for } k_x = N/2+1, \ldots, N-1, \text{ and } k_y = 0, \\ 0, & \text{otherwise.} \end{cases}$$

$$\hat{I}_y(k_x, k_y) = \begin{cases} -j\frac{\hat{I}_2(k_x, k_y)}{2\sin(2\pi k_x/N)} e^{-j2\pi k_x/N}, & \text{for } k_x, k_y = 1, \ldots, N-1, \\ N\sqrt{\hat{S}_y(0, k_y)}\hat{W}_y(k_y) e^{j\phi_y(k_y)}, & \text{for } k_x = 0 \text{ and } k_y = 0, \ldots, N/2, \\ \hat{I}_y^*(0, N - k_y), & \text{for } k_x = 0 \text{ and } k_y = N/2+1, \ldots, N-1,, \\ 0, & \text{otherwise.} \end{cases}$$

12. Compute the 1st order increments along the image boundaries for $m_x, m_y = 0, \ldots, M-1$:

$$I_x(m_x, 0) = \frac{1}{N^2} \sum_{m_x=0}^{N-1} \sum_{m_y=0}^{N-1} \hat{I}_x(k_x, k_y) e^{-j\frac{2\pi m_x k_x}{N}}, \tag{4.5}$$

$$I_y(0, m_y) = \frac{1}{N^2} \sum_{m_x=0}^{N-1} \sum_{m_y=0}^{N-1} \hat{I}_x(k_x, k_y) e^{-j\frac{2\pi m_y k_y}{N}}. \tag{4.6}$$

13. Add up the increments to calculate the fBm field for $m_x, m_y = 0, \ldots, M$ via

$$\begin{aligned} B(0,0) &= 0, \\ B(m_x,0) &= B(m_x-1,0) + I_x(m_x-1,0), \\ B(0,m_y) &= B(0,m_y-1) + I_y(0,m_y-1), \\ B(m_x,m_y) &= B(m_x,m_y-1) + B(m_x-1,m_y) - B(m_x-1,m_y-1) + I_2(m_x-1,m_y-1). \end{aligned}$$

When implementing the algorithm, the expressions (4.1), (4.2), (4.3), (4.4), (4.5), and (4.6) can be implemented using FFTs. To generate an $(M+1) \times (M+1)$ image, the algorithm uses $4(M+1)$ 1D FFTs of size $2M$ to compute (4.1) and (4.2) and another 4 FFTs of size $2M$ to compute (4.3), (4.4), (4.5), and (4.6). In total, the algorithms requires $4M+8$ 1D FFTs of size $2M$. In contrast, the usual Fourier synthesis takes one 2D inverse DFT which requires $2(M+1)$ 1D FFTs. In other words, the incremental Fourier synthesis of fBm uses close to exact statistics of fBm with only twice the computational cost of standard Fourier synthesis.

5 Experimental Results

We use the proposed incremental Fourier synthesis method to generated 256 realizations of 17×17 fBm processes with $H = 0.8$ and $H = 0.2$ where $\sigma^2 = 1$ for both cases. We calculated the variance at all pixels over the 256 independent images. Based on (2.4), the variance of pixel location (m_x, m_y) should $\sqrt{m_x^2 + m_y^2}^{2H}$. The theoretical and experimental variances are displayed in Figs. 1 and 2. Note that the usual Fourier synthesis method generates a stationary process and, consequently, the variance is constant over all pixel values.

To compare the actual statistics of images created by the standard and incremental Fourier synthesis method, we calculate the inverse FFT of the power spectrums that were used to scale white noise. One can calculate the normalized variance as the displacement size of the x directed increments of the generated picture grows, i.e.

$$\tilde{f}(d) = \frac{\mathrm{VAR}[B(m_x+d, m_y) - B(m_x, m_y)]}{\mathrm{VAR}[B(m_x+1, m_y) - B(m_x, m_y)]}, \quad d \in \mathbf{Z}^+.$$

The function $\tilde{f}(d)$ is known as the structure function. For true fBm, the structure function is a hyperbolic function with respect to d due to (2.2). To get a local measurement of the rate $\tilde{f}(d)$ is increasing with respect to scale we define a generalized Hurst parameter as,

$$\tilde{H}(s) = \frac{1}{2}\log_2(\tilde{f}(2^{s+1})/\tilde{f}(2^s)). \tag{5.1}$$

Obviously for true fBm statistics, the value of $\tilde{H}(s)$ is equal to a constant H for all scales s. The value $\tilde{H}(s)$ for our incremental method with H set to 0.2 and the image size set to 512×512 is shown in Fig. 3(a). The figure shows that the actual process generated by incremental Fourier synthesis is nearly constant, i.e. virtually self-similar. Note that an analysis of the y directed increments will yield the same results due to the isotropy of the generating algorithm. The generalized Hurst parameter $\tilde{H}(s)$ for standard Fourier

synthesis is plotted in Fig. 3(b) for comparison. It is clear that images generated by standard Fourier synthesis are not statistically self-similar. In fact, the figure suggests that the generated images will be smoother at finer scales since the value of $\tilde{H}(s)$ becomes larger.

To demonstrate the drawback of standard Fourier synthesis, we generated two realizations of 2D fBm of size 512×512 using the two Fourier methods with $H = 0.2$. Figs. 4 and 5 show the images generated by the standard and incremental Fourier methods, respectively, at different scales. At each scale, the resolution of the picture is 64×64, and each picture is scaled so that the dynamic range of the pixel values cover all 64 gray level values. The statistical self-similarity is evident for the fBm realization created by our new method. As predicted by the generalized Hurst parameters, the fBm realization generated by traditional Fourier synthesis is smoother at finer scales.

6 Conclusions

A new method called incremental Fourier synthesis was proposed to synthesize self-similar images based on a 2D fBm model. The advantage of the method is that it is a relatively fast algorithm while it generates processes whose statistics virtually match those of true fBm. Some interesting topics arise along this research direction. For example, with the incremental Fourier method, one can also choose an arbitrary structure function and substitute the new $\tilde{f}(d)$ in (2.8), (2.6), and (2.7). By choosing alternative forms of the structure function, an artist has precise control of the "roughness" of the texture with respect to scale. Furthermore, the algorithm can be extended to generate 3D (video) and even higher dimension fBm at the expense of $O(N^d \log_2(N))$ computations where d is the dimension.

References

[1] H. Akaike, "Block Toeplitz matrix inversion," *SIAM J. Appl. Math*, Vol. 24, pp. 234–241, Mar. 1973.

[2] L. T. Bruton and N. R. Bartley, "Simulation of fractal multidimensional images using multidimensional recursive filters," *IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Processing*, Vol. 41, pp. 181–188, Mar. 1994.

[3] S. Hoefer, H. Hannachi, M. Pandit, and R. Kumaresan, "Isotropic two-dimensional fractional Brownian motion and its application in ultrasonic analysis," in *Proc. of the 14th IEEE Engineering in Medicine and Biology Society Conference*, pp. 1267–1269, 1992.

[4] S. Hoefer, F. Heil, M. Pandit, and R. Kumaresan, "Segmentation of textures with different roughness using the model of isotropic two-dimensional fractional Brownian motion," in *IEEE ICASSP-93*, vol. 5, pp. 53–56, Apr. 1993.

[5] J. P. Lewis, "Generalized stochastic subdivision," *ACM Trans. on Graphics*, Vol. 6, pp. 167–190, July 1987.

[6] B. B. Mandelbrot, *The Fractal Geometry of Nature*, San Francisco: Freeman, 1982.

[7] B. B. Mandelbrot and J. W. V. Ness, "Fractional Brownian motions, fractional noises and applications," *SIAM Review*, Vol. 10, pp. 422–437, Oct. 1968.

[8] H. O. Peitgen and D. Saupe, eds., *The Science of Fractal Images*, New York: Springer-Verlag, 1988.

[9] W. Rűmelin, "Fractal interpolation of random fields of fractional Brownian motion," in *Fractal Geometry and Computer Graphics* (J. L. Encarnacao, ed.), New York: Springer-Verlag, 1992.

Figure Captions

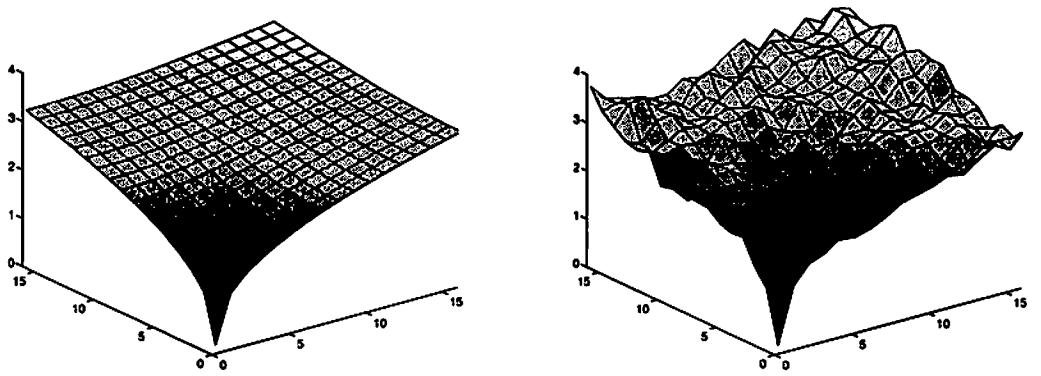
Figure 1: The variance of the generated fBm images for $H = 0.2$.

Figure 2: The variance of the generated fBm images for $H = 0.8$.

Figure 3: Theoretical values of $\tilde{H}$ for the 512×512 realizations of the two Fourier methods when $H = 0.2$.

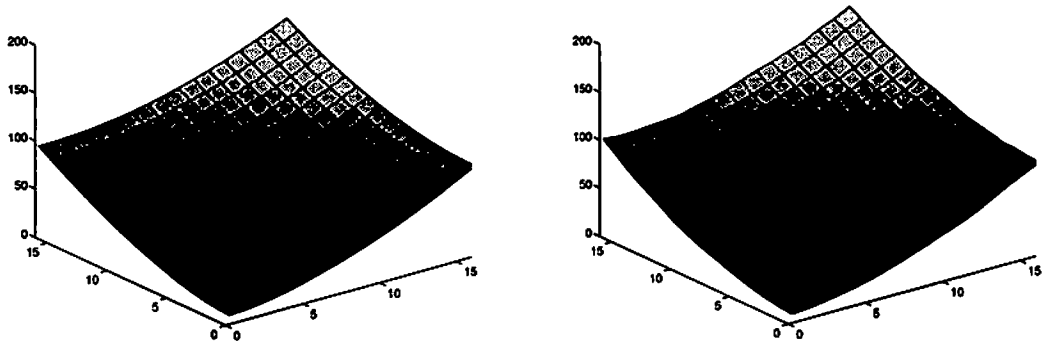
Figure 4: Zooming into a texture generated by standard Fourier synthesis with $H = 0.2$.

Figure 5: Zooming into a texture generated by incremental Fourier synthesis with $H = 0.2$.



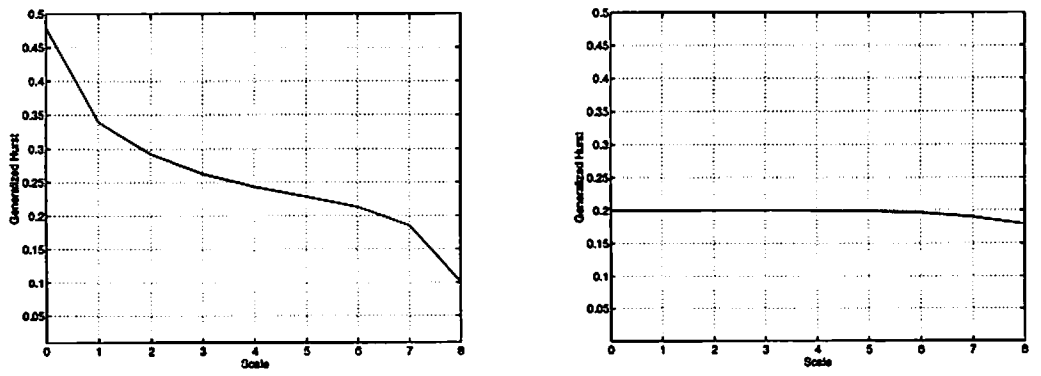
(a) Theoretical (b) Experimental

Figure 1: The variance of the generated fBm images for $H = 0.2$.



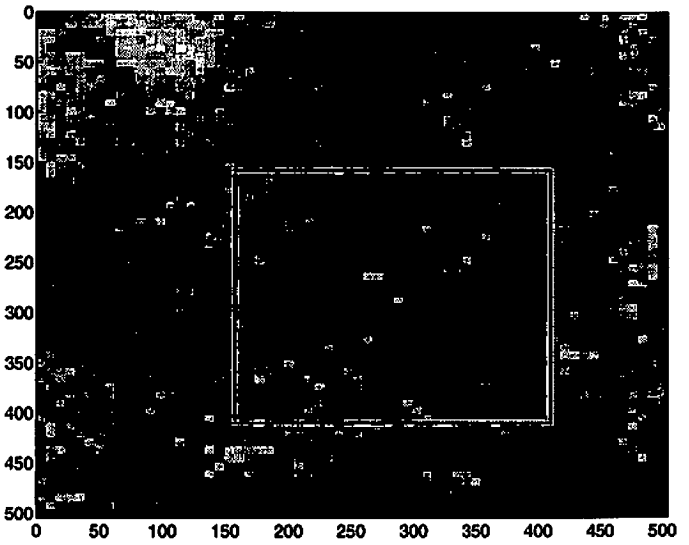
(a) Theoretical (b) Experimental

Figure 2: The variance of the generated fBm images for $H = 0.8$.

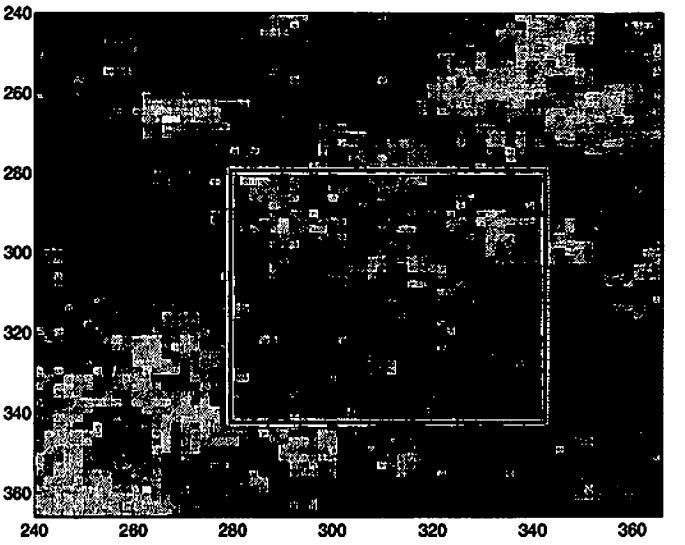


(a) Standard (b) Incremental

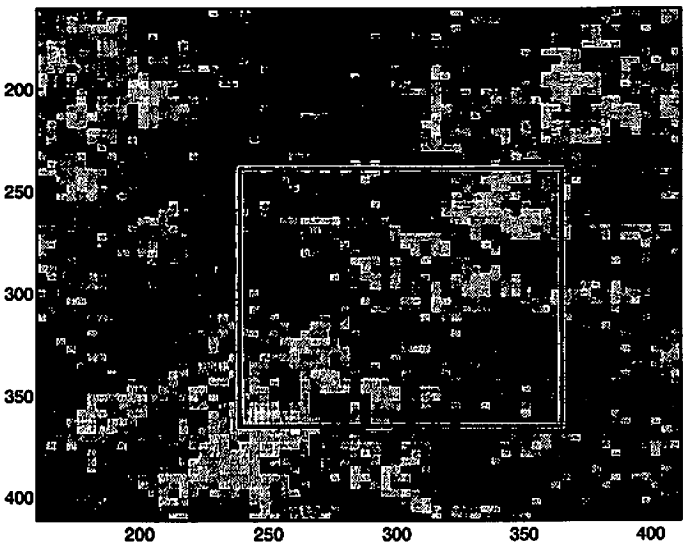
Figure 3: Theoretical values of $\tilde{H}$ for the 512×512 realizations of the two Fourier methods when $H = 0.2$.



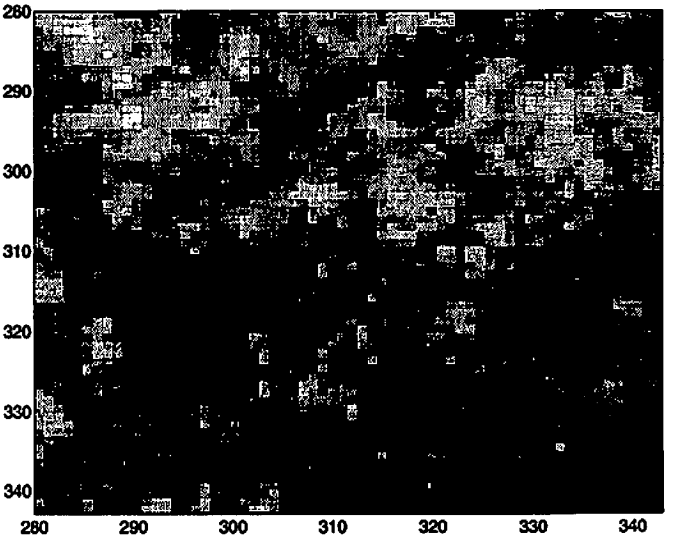
(a) Sampled every 8 units.



(c) Sampled every 2 units.

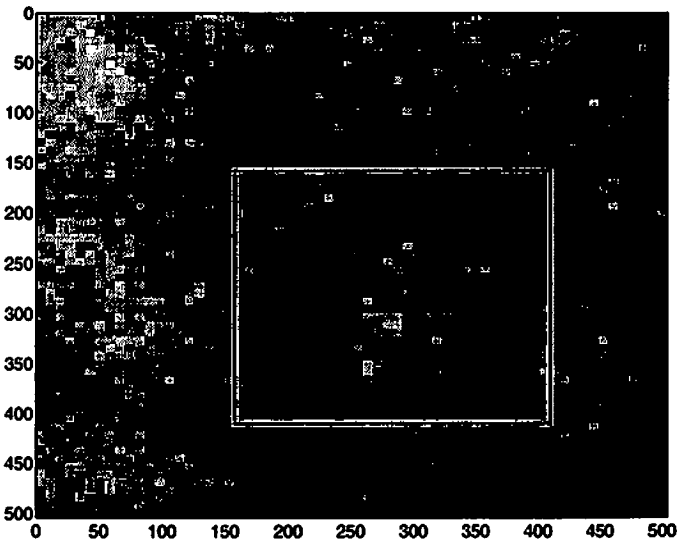


(b) Sampled every 4 units.

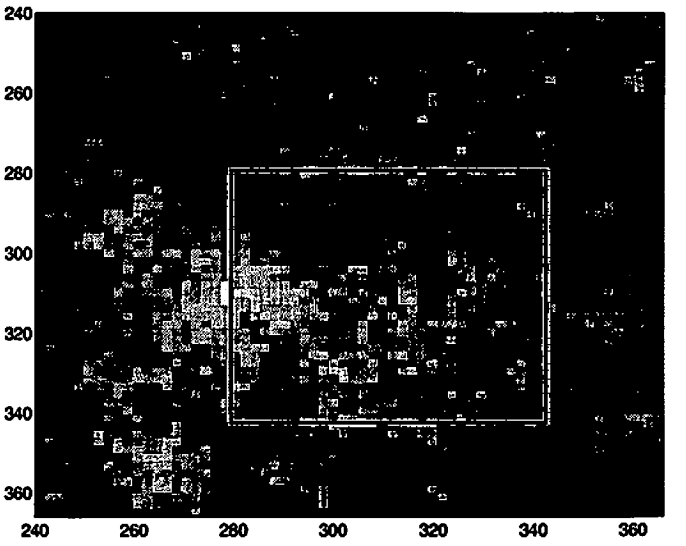


(d) Sampled every unit.

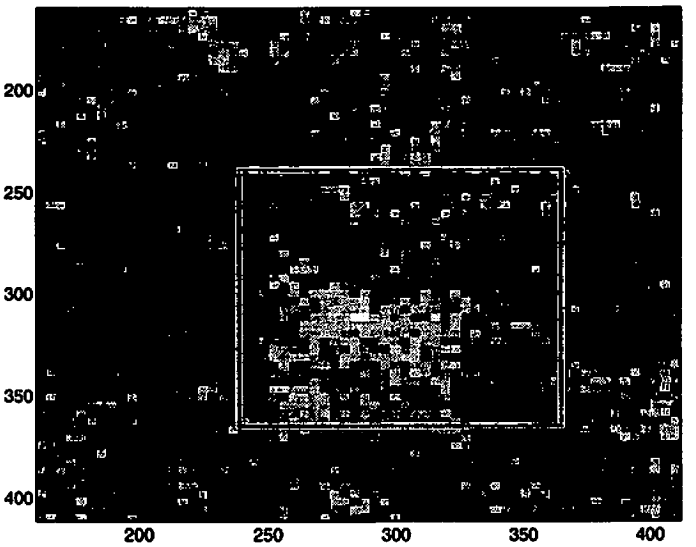
Figure 4: Zooming into a texture generated by standard Fourier synthesis with $H = 0.2$.



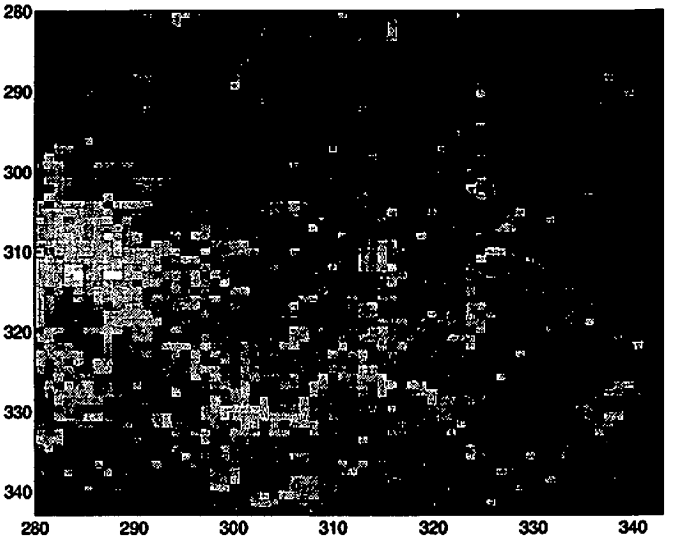
(a) Sampled every 8 units.



(c) Sampled every 2 units.



(b) Sampled every 4 units.



(d) Sampled every unit.

Figure 5: Zooming into a texture generated by incremental Fourier synthesis with $H = 0.2$.