

Gabor Noise 的生成

摘要

噪声对于建模与材质都十分重要，而一个好的噪声需要能满足分辨率可调，参数可调，生成方便快捷，具有各向同性或异性的性质。一般的噪声生成很难满足这些条件，但是 Gabor Noise 具有上述所有性质。本文介绍了如何在 Sparse 卷积以及 Gabor Kernel 的基础上生成 Gabor Noise，以及 Gabor Kernel 与 Gabor Noise 在频域上的表现。

1. Gabor Noise 综述

1.1 Sparse White Noise

Sparse White Noise 是一系列随机放置并带有权重的点。利用 Sparse White Noise 作为基础，在每个点的位置放置一个随机权重的径向对称的余弦核（radially symmetric smooth cosine kernel），即将余弦核作为脉冲与 Sparse White Noise 做卷积，就可以得到 Sparse Convolution Noise。Sparse White Noise 的点分布遵从泊松分布，每个点的权重是一个限定范围内的随机数，所有点的权重的平均数为 0。

1.2 Gabor Kernel

在上述过程中提到，使用余弦核作为脉冲来生成 Sparse Convolution Noise。但是为了更精确的控制随机脉冲过程的功率谱，控制生成的噪声的性质，应当选择能够参数化控制，并且在空间域与频域都有精确表现的脉冲。因此选择了 Gabor Kernel。Gabor Kernel 是一个高斯波包和一个谐波的乘积，我们可以使用一个二维的高斯核和余弦波图像来得到它。

$$g(x, y) = K e^{-\pi a^2(x^2+y^2)} \cos[2\pi F_0(x \cos \omega_0 + y \sin \omega_0)] \quad (1)$$

其中 K 为高斯核的量级，a 为高斯核的宽度， F_0 为余弦波的频率， ω_0 为余弦波的朝向。用这四个参数可以很好地控制 Gabor Kernel 的性质，从而控制最终生成的噪音的性质。

经过傅里叶变换后，Gabor Kernel 在频域上的表现为

$$G(f_x, f_y) = \frac{K}{2a^2} \left\{ e^{-\frac{\pi}{a^2}[(f_x - F_0 \cos \omega_0)^2 + (f_y - F_0 \sin \omega_0)^2]} + e^{-\frac{\pi}{a^2}[(f_x + F_0 \cos \omega_0)^2 + (f_y + F_0 \sin \omega_0)^2]} \right\} \quad (2)$$

1.3 Gabor Noise

仿照 Sparse Convolution Noise 的生成，Gabor Noise 是 Gabor Kernel 与 Sparse White Noise 的卷积。对于各向异性的 Gabor Noise，我们使用相同参数的 Gabor kernel。

$$N(x, y) = \sum_i w_i g(x - x_i, y - y_i) \quad (3)$$

而对于各向同性的 Gabor Noise，我们将每次脉冲的 Gabor kernel 的方向随机化，这样叠加出来的噪声在各个方向上的性质相同。

$$N(x, y) = \sum_i w_i g(x - x_i, y - y_i; \omega_{0,i}) \quad (4)$$

Gabor Noise 有如下性质：

过程化：由于噪声可以过程化生成，所以我们不需要很大的空间来储存噪声图像，同时该噪声也不受分辨率的限制，可以无限拓展。该噪声也是参数化的，可以随时调整生成参数来控制噪声的性质。

各向异性：由于 Gabor Kernel 引入了谐波，所以是有方向性的。因此 Gabor Noise 也可以是有特定方向的。即该噪声能够同时支持各向同性与各向异性。

光谱控制：能够描述每个频带的贡献的功率谱是控制噪声的一个强大手段。Gabor Noise 实现了对功率谱的控制，从而能够得到多种噪声样式。

表面噪声：在不经预处理的情况下可以直接应用到固体纹理上。

快速评估：通过不同的参数调整，Gabor Noise 可以通过交互的方式控制生成质量与生成速度之间的平衡，从而可以应用到不同的领域。

2. Gabor Noise 生成

2.1 空间域的生成

我们不使用直接卷积的方式来生成噪声，而是通过逐步生成随机脉冲，逐像素点填充的方式。对于分辨率为 h 的图像，令 (x,y) 的取值范围为 $[-\frac{h}{2}, \frac{h}{2}]$ 。将图像平面划分出网格，网格的宽度为 Gabor Kernel 的半径，则每一个确定的 (x,y) 对应一个确定的网格。如图 1 所示，由于网格的宽度等于 Gabor Kernel 的半径，所以一个 Gabor Kernel 最多可以影响到 9 个网格中的点，同时也就意味着任意一点 (x,y) 的值会受到其所在的以及周围的一共 9 个网格中所有的脉冲的影响。所以计算某一点的值时，只需要计算周围所有脉冲在该点的影响，然后叠加起来即可。

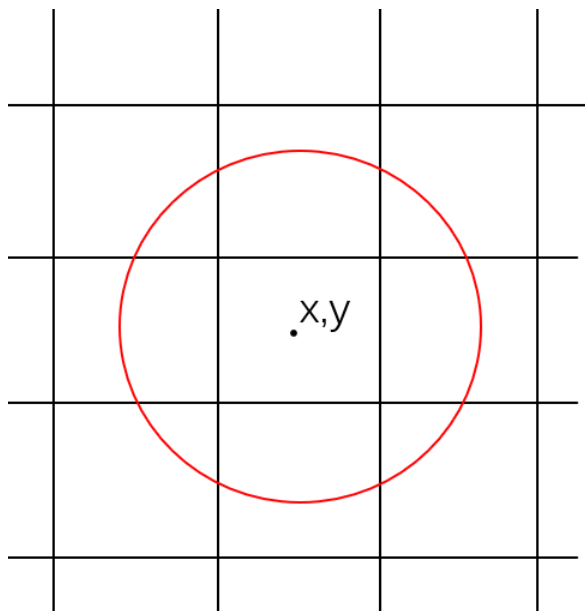


图 1

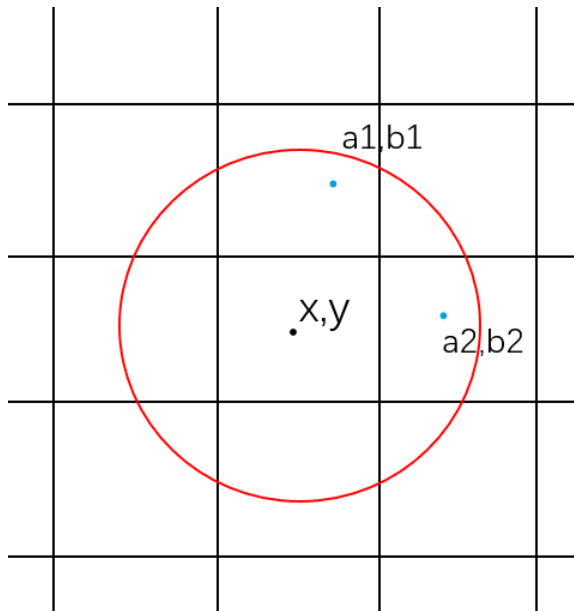


图 2

上述说到噪声图片是逐像素填充的，如图 2，当点 (a_1, b_1) 与 (a_2, b_2) 都受到中心在 (x,y) 的脉冲的影响时，需要确保两次计算中 (x,y) 的位置不会变化，否则我们会生成错误的图像。为了保证噪声的随机性，脉冲的位置和幅值又应当是随机的，所以我们使用了一个伪随机数生成器来控制。通过控制伪随机数生成器

的种子的值,保证了整体上所有脉冲是随机的,满足泊松分布,但每个脉冲的位置和幅值又是可计算的,可重复生成的。

对于空间域噪声图像上的一点 (x_0, y_0) ,该点的像素值计算流程如下:

- (1) 确定 (x_0, y_0) 所属的网格, 以及其周围的八个网格, 记为 (g_x, g_y) 。
- (2) 对于上述网格 (g_x, g_y) , 使用 Morton 编码方式来得到与该网格对应的唯一编码 s 。
- (3) 将上述编码 s 加上一个全局的随机偏移, 作为随机数生成器的种子。记当前的随机数生成器为 p 。全局的随机偏移量是每次噪声图片生成前确定的, 这保证了在相同的参数下, 每次生成的噪声有些许差异, 提高了噪声的随机性。
- (4) 使用 p 来生成网格 (g_x, g_y) 中的所有脉冲, 包括其位置与幅值。由于编码 s 与全局偏移都是确定值, 所以对于某确定的网格, p 也是确定的。这就保证了每次使用到网格 (g_x, g_y) 中的脉冲时, 其属性都是一样的。
- (5) 对于上述九个网格中的所有的脉冲, 使用公式 (1) (见 1.2) 计算其在 (x_0, y_0) 的值, 然后做叠加。叠加的结果就是 (x_0, y_0) 处的最终像素值。
- (6) 注意到对于各向同性的噪声, 参数 ω_0 为根据 p 随机生成的朝向。

2.2 频域的生成

首先根据公式 (2) (见 1.2) 生成单个 Gabor Kernel 的频域图像。

对于噪声的频域图像, 其生成与空间域类似。

将 2.1 中计算流程的 (5) 修改为使用公式 (2) (见 1.2) 计算所有脉冲在频域上的叠加即可。

3. 结果

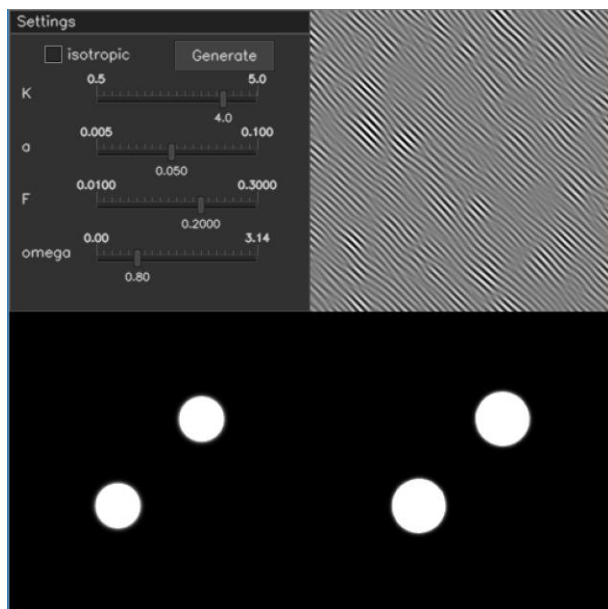
3.1 软件使用方法

使用选择框 (选中与否) 来选择生成各向异性/同性的噪声, 使用滑动条来调整参数的大小, 调整完后点击 Generate 按钮生成。

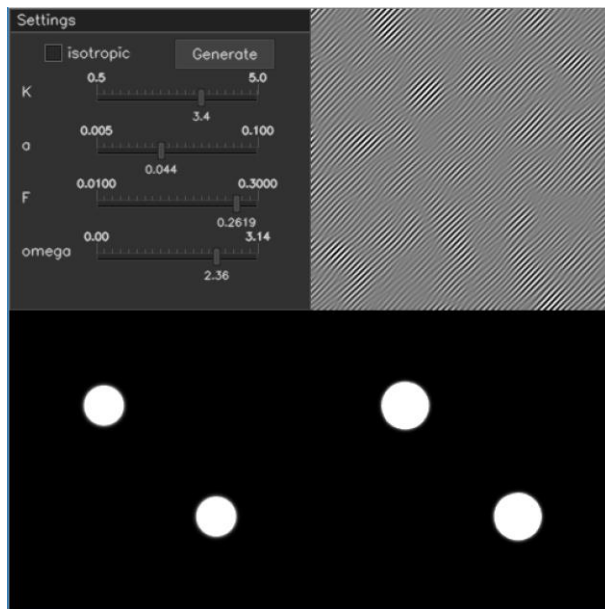
生成一张图片的时间约为 3 秒。

3.1 各向异性

对于各向异性的噪声，由于其是相同的参数脉冲的叠加，其在频域上的表现与单个 Gabor Kernel 在空间域上的表现相同。



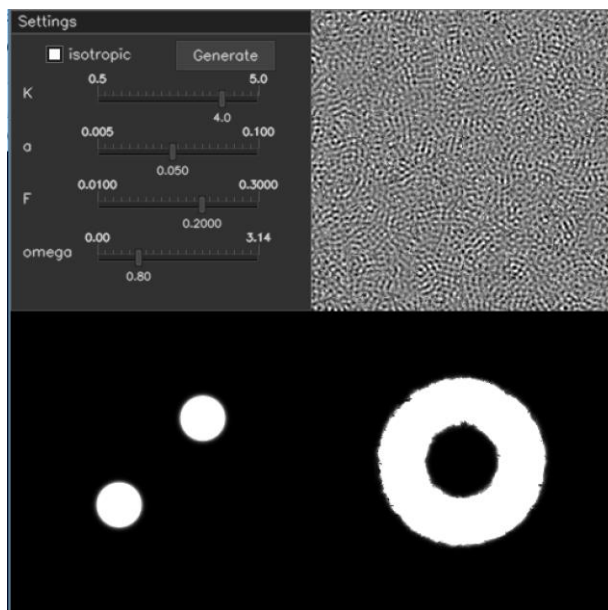
$K=4.0$, $a=0.05$, $F=0.2$, $\omega=0.8$, 64 脉冲/网格
频域：左下为单个 kernel，右下为噪声图片



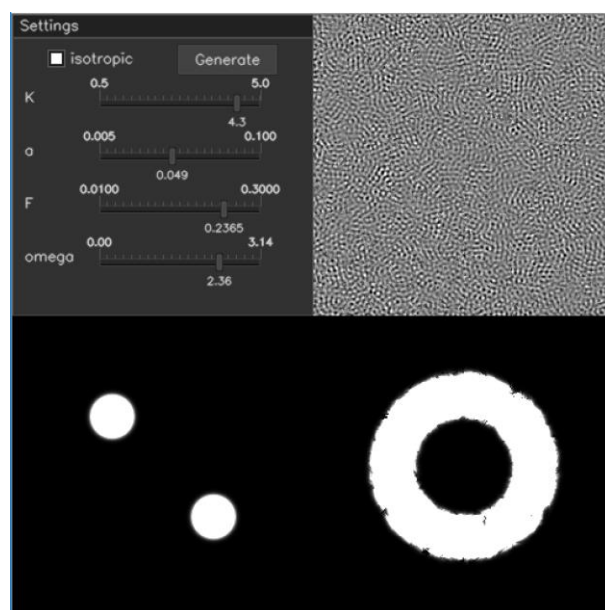
$K=3.4$, $a=0.044$, $F=0.2619$, $\omega=2.36$, 64 脉冲/网格
频域：左下为单个 kernel，右下为噪声图片

3.2 各向同性

于各向同性的噪声，由于脉冲的方向是随机的，其在频域上的表现为一个圆环，即不同方向的 kernel 在频谱上的叠加。



$K=4.0$, $a=0.05$, $F=0.2$, $\omega=0.8$, 64 脉冲/网格
频域：左下为单个 kernel，右下为噪声图



$K=4.3$, $a=0.049$, $F=0.2365$, $\omega=2.36$, 64 脉冲/网格
频域：左下为单个 kernel，右下为噪声图片

参考资料

- Ares Lagae, Sylvain Lefebvre. 2009. *Procedural Noise using Sparse Gabor Convolution*
- Victor Charpenay, Bernhard Steiner. 2014. *Sampling Gabor Noise in the Spatial Domain*
- Bruno Galerne, Ares Lagae. 2012. *Gabor Noise by Example*