

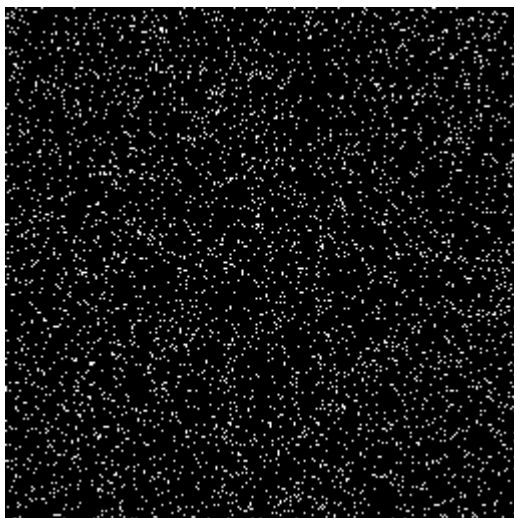
采样

介绍

这个代码题中，你将使用 `python` 实现三种图像点噪声生成算法，它们算法结果的频谱特征各不相同。你可以通过运行 `python main.py` 来生成算法结果。

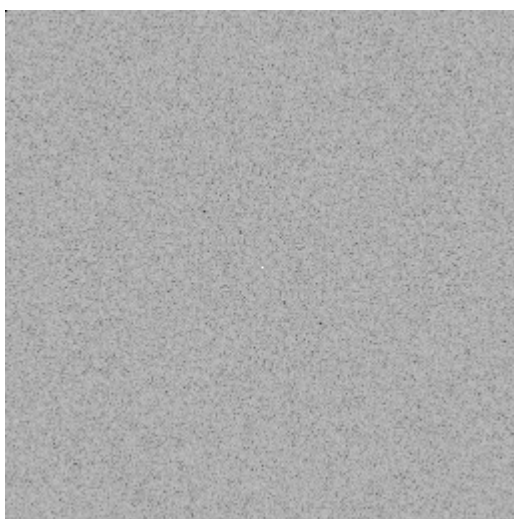
Threshold

生成平面的点噪声，最简单的方式是先生成一张均匀采样的白噪声灰度图片，然后设置`threshold`，令灰度值大于`threshold`的像素取255；灰度值小于`threshold`的像素取0。请完成 `utils.py` 中的 `SampleByThreshold` 函数。效果如下：



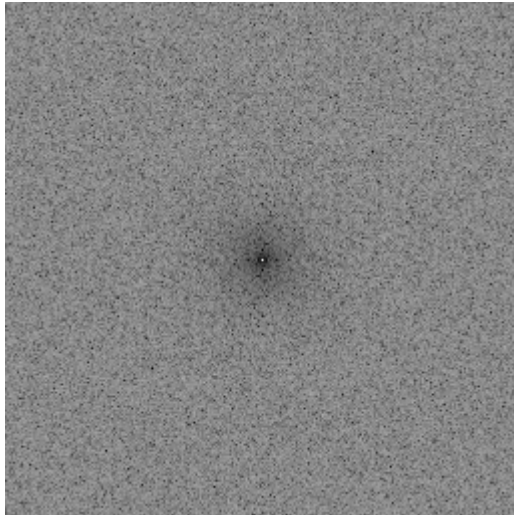
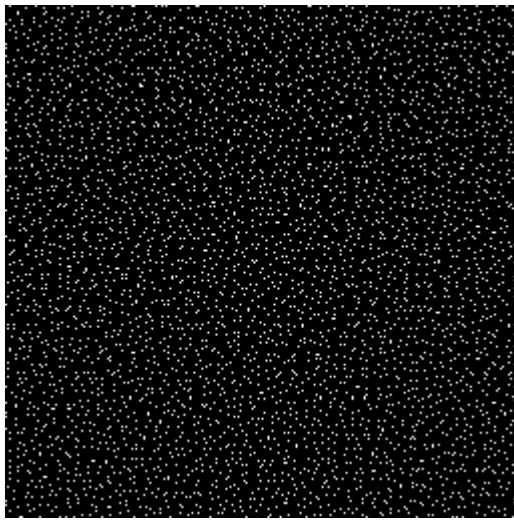
频谱分析

`numpy`提供有快速傅里叶变换(FFT)的算法接口。接下来请在 `utils.py` 中的 `ImageToFFT` 函数对刚才生成的点采样图片作频谱分析。具体地，请首先对图片进行快速傅里叶变换，然后将零频率分量移至频谱中心，返回 `log(1 + abs(result))` 用以可视化。效果如下：



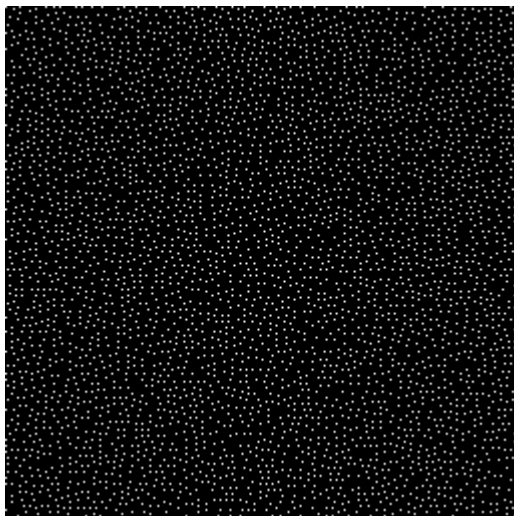
带抖动的分层采样

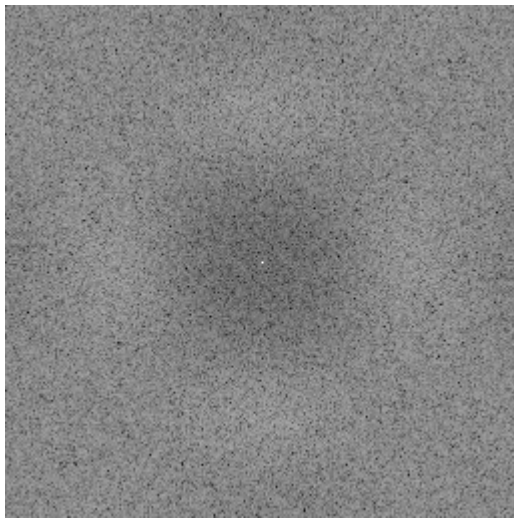
这里提供一种以简单方式获得弱低频的点采样算法：将图像划分为小尺寸的子网格（如 4×4 ），在每个子网格内随机选取一个像素点着色。请完成 `utils.py` 中的 `SampleByWindow` 函数。对生成的采样图像作频谱分析，你会发现其低频区域更暗。效果如下：



泊松圆盘采样

这是图形学常用的一种生成蓝噪声点采样的算法。请自行查阅算法并完成 `utils.py` 中的 `PoissonDiskSampling` 函数。效果如下：





差异计算 (discrepancy)

`scipy` 提供了计算差异值的接口。请对于以上运行结果计算对应的差异值，并将差异值存储于 `discrepancy.txt` 中。

提交

将完成以后的 `utils.py` 提交即可。