1. **滤波器：Gaussian & Butterworth**

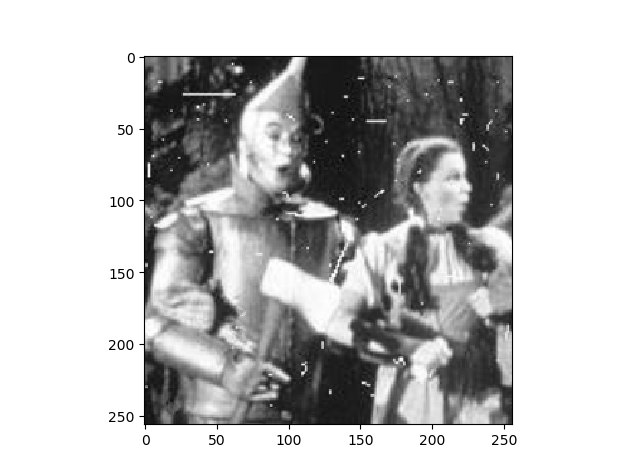
 在频域对图像进行滤波所采取的方法是用滤波器卷积图像的频谱，这个过程不会改变图像的相位，这是因为滤波器中的数值均为实数，相角为0.  
 在频域中常用的低通滤波器主要为高斯滤波器与巴特沃斯滤波器，它们都是基于像素点距离频谱中心的距离远近而定义的。频谱的中心点为直流分量，距离中心点越远，说明谐波阶数越高，也就是高通分量。因此，在频域中进行滤波实际上就是衰减某个频率半径以外的频谱数值。

确定了滤波器后，最重要的一步就是对图像进行二维傅里叶变换与傅里叶反变换，由于FFT的使用，我们可以很快的完成上述转换，这里，根据DFT的性质，可以得到以下简便方法：

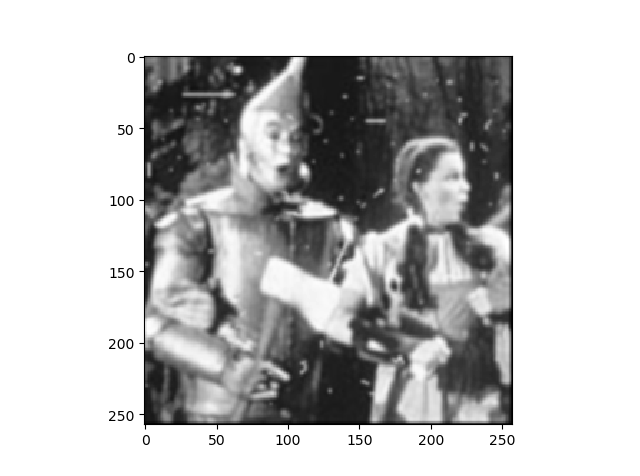
* 对于二维傅里叶变换，只需要对图像的行与列分别进行傅里叶变换即可，这意味着我们可以先对图像的每一行进行DFT，然后对得到的图像在按列进行DFT，当然，这里的顺序也是改变的。
* 为了加快DFT的运算过程，做DFT的点数应当为[https://github.com/szxhcl/hw5/raw/492a0b14f3cbdc9b49ecaf0e4a3264e81b989519/eq4.gif](https://github.com/szxhcl/hw5/blob/492a0b14f3cbdc9b49ecaf0e4a3264e81b989519/eq4.gif)。因此，在对行或列进行padding操作时要注意这一点。
* 从二维傅里叶变换返回时空域，只需要对频域结果再进行一次二维傅里叶变换，然后取实部并除以MN，这里MN是频域中图像的点数。

根据上面的步骤，可以很容易的在频域对图像进行滤波，为了便于观察，将频谱的数值取对数后再画出是一种非常有效的方法。这是因为零频点的幅值往往超出其其它点若干数量级，用取对数的方式可以减小这种差异。

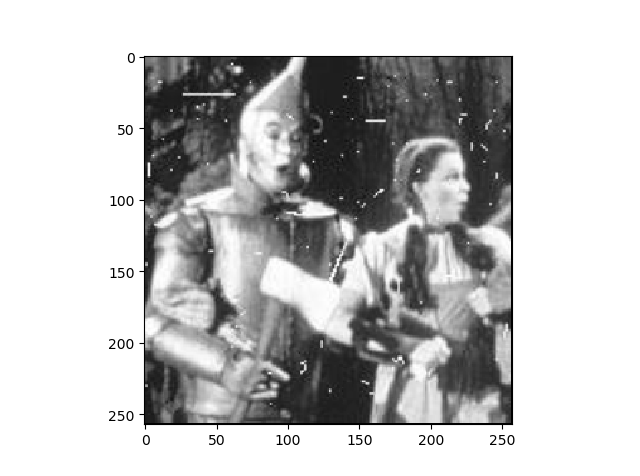
频域Butterworth低通(D = 100, N = 2) &&Gaussian低通（D = 100）



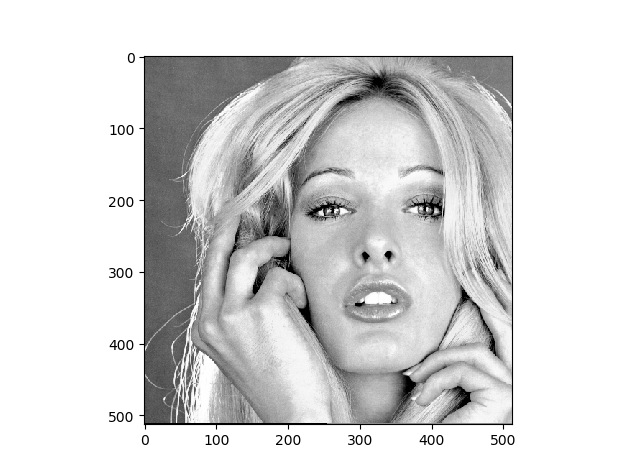
Test1 original



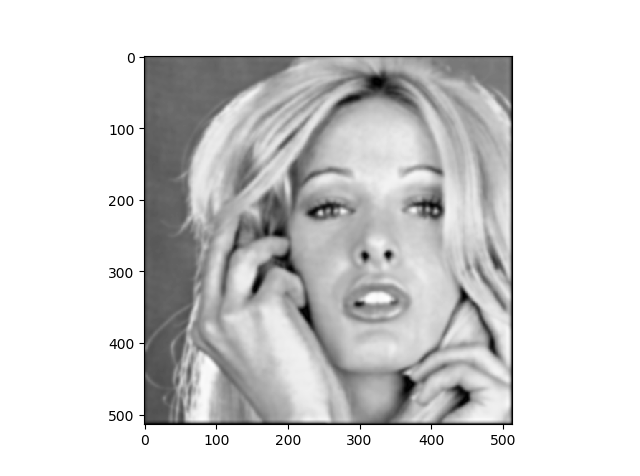
Butterworth\_LP



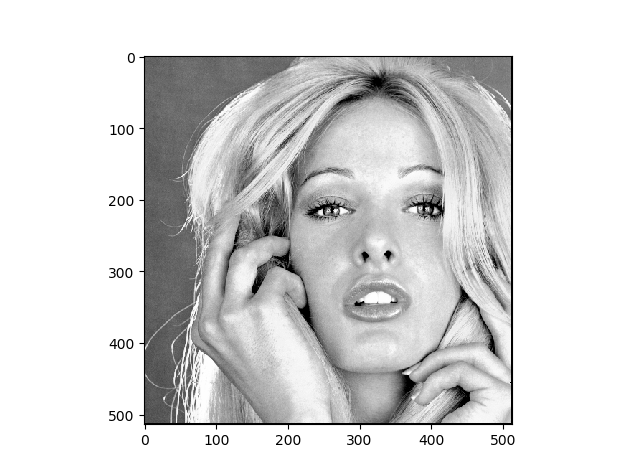
Gaussian\_LP



Test2 original



Butterworth\_LP

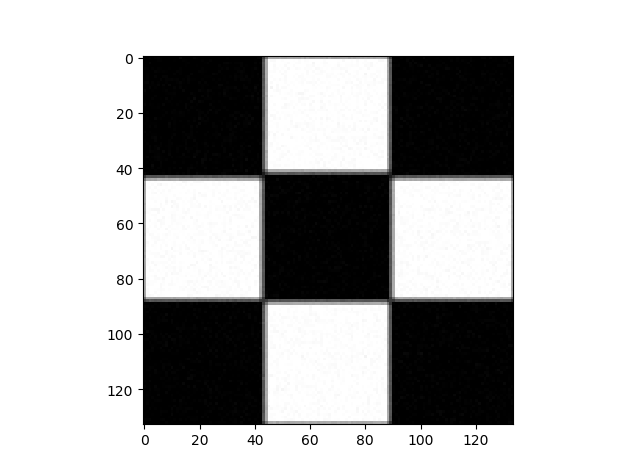


Gaussian\_LP

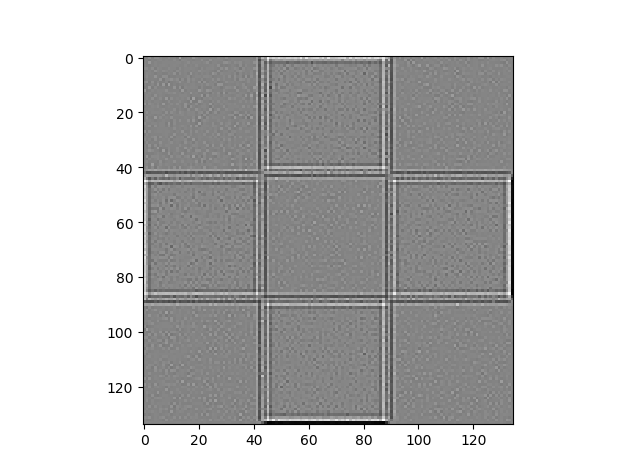
**二．频域高通通滤波器：Gaussian & Butterworth**

频域高通滤波器就是让高频，也就是距离中心点距离远的频谱分量通过。

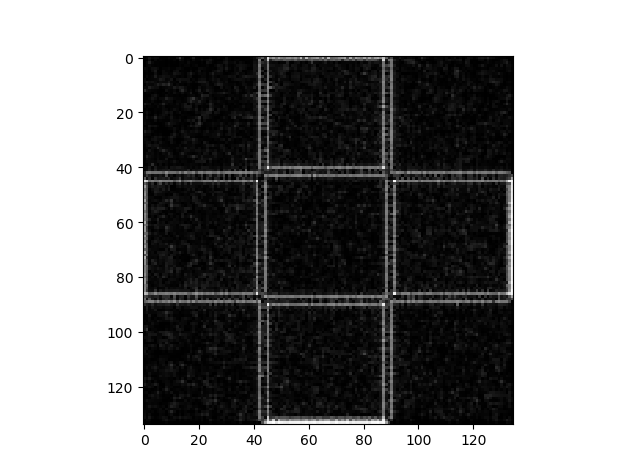
频域Butterworth高通(D = 100, N = 2) &&Gaussian高通（D = 100），但是对于图四的高斯高通滤波效果不理想



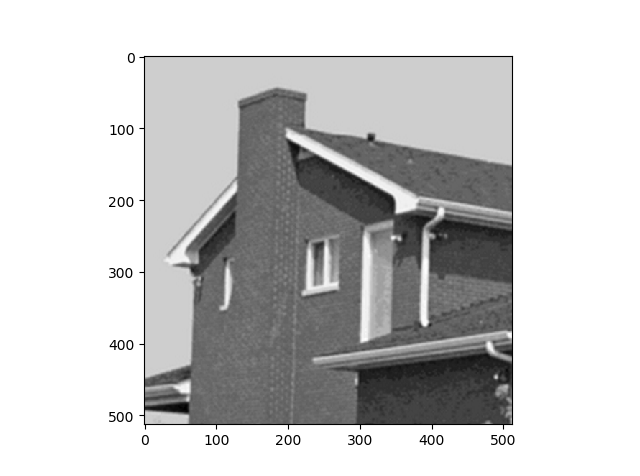
Test3 original



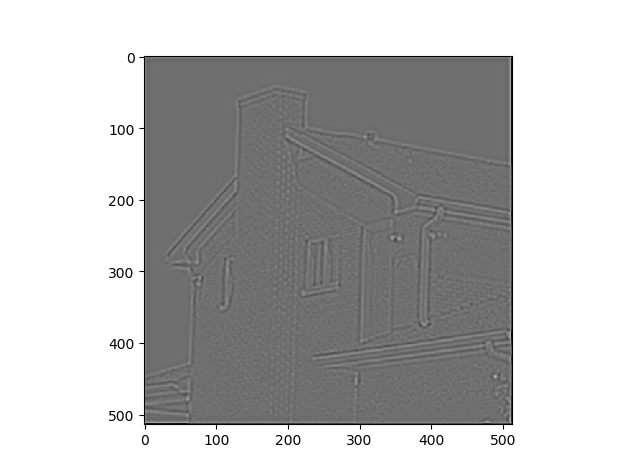
Butterwroth\_HP



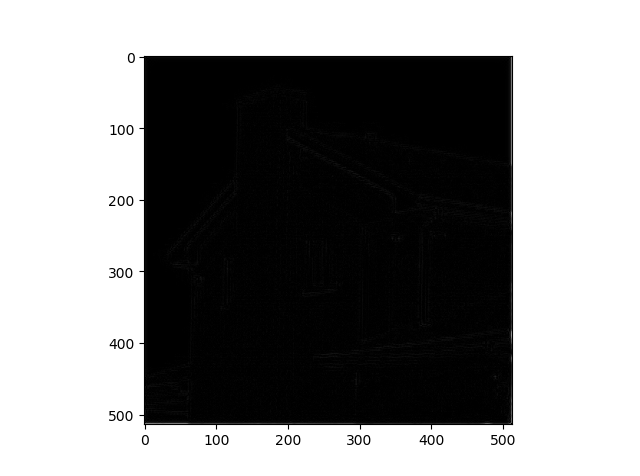
Gaussian\_HP



Test4 original



Butterworth\_HP



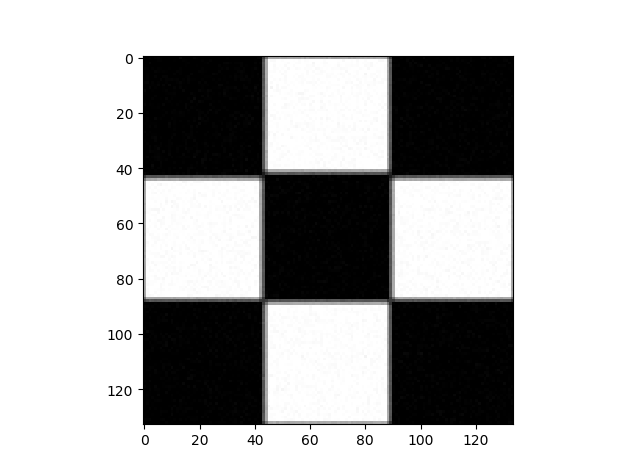
Gaussian\_HP

**三．频域高通通滤波器：Laplace & Unmask**

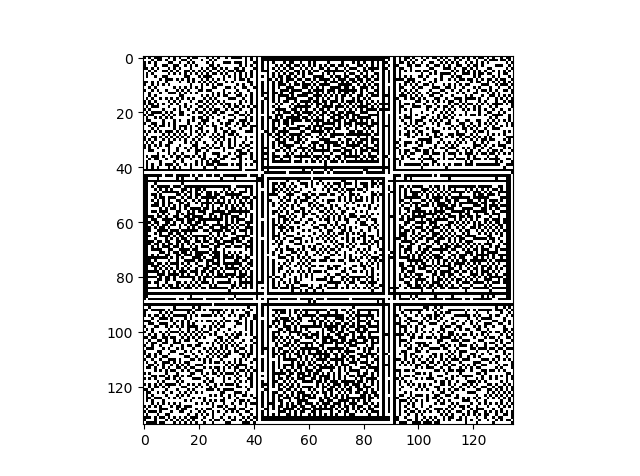
Unsharp Masking 是一种非常容易理解的边缘强化方法，它获取边缘信息的方式是用原始图像减去模糊后的图像。相对而言，laplace 滤波计算要稍微复杂一些，涉及一些数据的尺度变换操作。

在频率域中使用Laplace滤波器要比在时空域使用复杂一些，但是，他们的视觉效果是相同的。

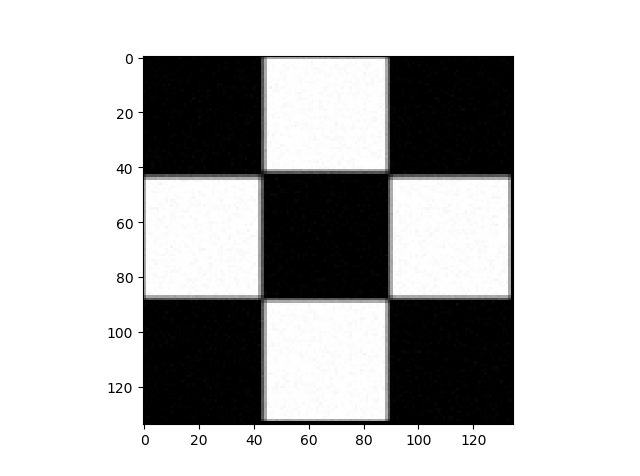
频域Laplace高通(D = 100, N = 2) && Unmask高通（D = 250 k1=1 k2=1）



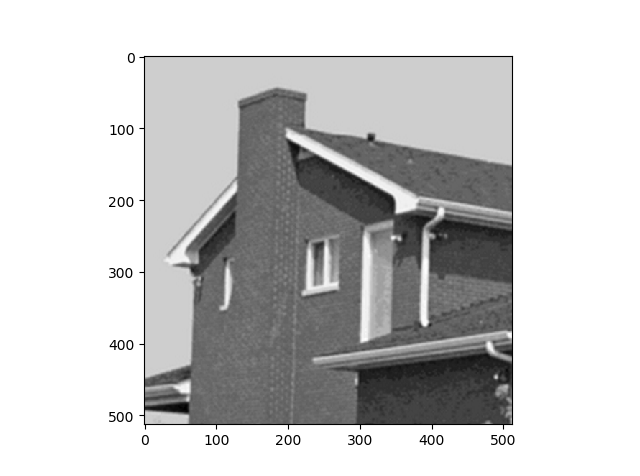
Test3 orginal



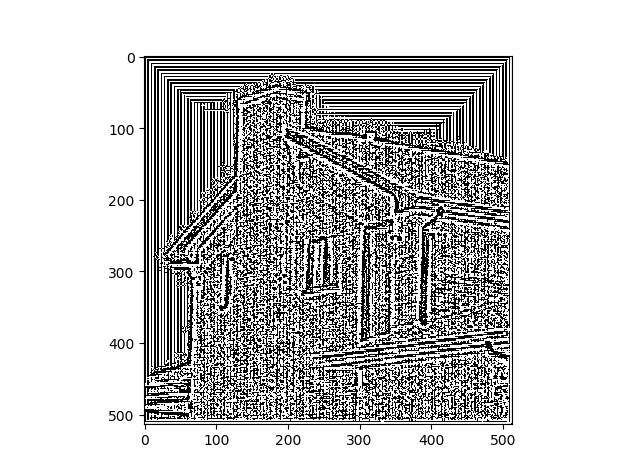
Laplace\_HP

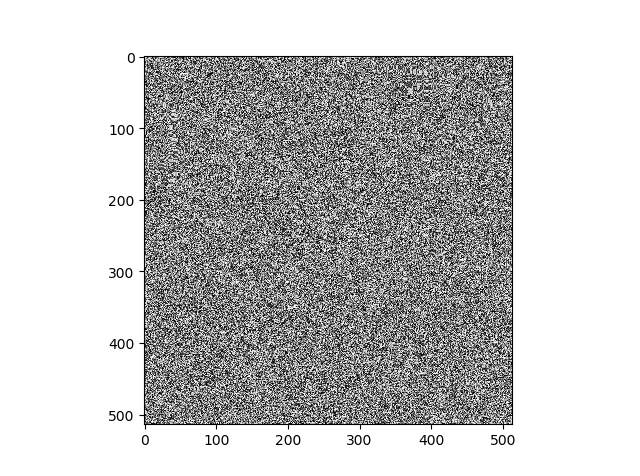


Unmask



Test 4 original

Laplace\_HP



Unmask\_HP