

计算机视觉与模式识别

第一次实验报告



王易博

人工智能82

2186113742

目录

[一、实验内容 2](#_Toc57078122)

[二、实验原理 2](#_Toc57078123)

[2.1 图像的高斯滤波与Padding 2](#_Toc57078124)

[2.2 高斯核与高斯核卷积实验 3](#_Toc57078125)

[2.3 图像锐化滤波器 3](#_Toc57078126)

[2.4双边滤波 4](#_Toc57078127)

[三、实验结果与分析 4](#_Toc57078129)

[3.1图像的高斯滤波与Padding 4](#_Toc57078130)

[3.2 高斯核与高斯核卷积实验 8](#_Toc57078131)

[3.3 图像锐化 9](#_Toc57078132)

[3.4 双边滤波 10](#_Toc57078133)

[3.5 图像的傅里叶变换 11](#_Toc57078134)

[3.6 探索：频率水印 12](#_Toc57078135)

[四、结论与讨论 13](#_Toc57078136)

[4.1图像的高斯滤波与Padding 13](#_Toc57078137)

[4.2高斯核与高斯核的卷积实验 13](#_Toc57078138)

[4.3 图像锐化 13](#_Toc57078139)

[4.4 双边滤波 13](#_Toc57078140)

[4.5 图像的傅里叶变换 13](#_Toc57078141)

[五、代码 14](#_Toc57078142)

[5.1 图像的高斯滤波与Padding 14](#_Toc57078143)

[5.2 高斯核与高斯核的卷积实验 16](#_Toc57078144)

[5.3 图像锐化 16](#_Toc57078145)

[5.4 双边滤波 16](#_Toc57078146)

[5.5 图像的傅里叶变换 17](#_Toc57078147)

[5.6 探索：频率水印 17](#_Toc57078148)

[5.7 作业所有代码及图片地址 17](#_Toc57078149)

# 一、实验内容

1、2D高斯模板的设计（给定方差生成滤波核）；图像的高斯滤波；在高斯滤波中不同边界的处理方法实验

2、高斯核与高斯核的卷积实验；利用两个相同方差的一维行列高斯核卷积生成2D高斯核，利用一维行列高斯对图像进行滤波；不同方差高斯核之差对图像进行滤波；

3、利用两个高斯核设计图像锐化滤波器核；

4、图像的双边滤波实验；

5、图像的Fourier变换，显示幅度谱与相位谱；利用高斯滤波器进行图像的频域滤波；

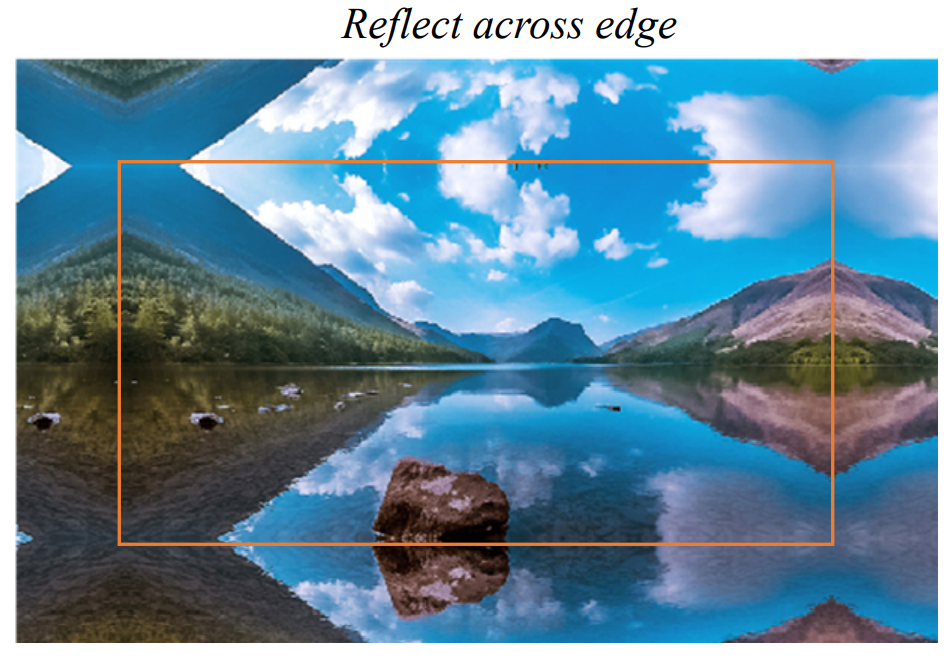
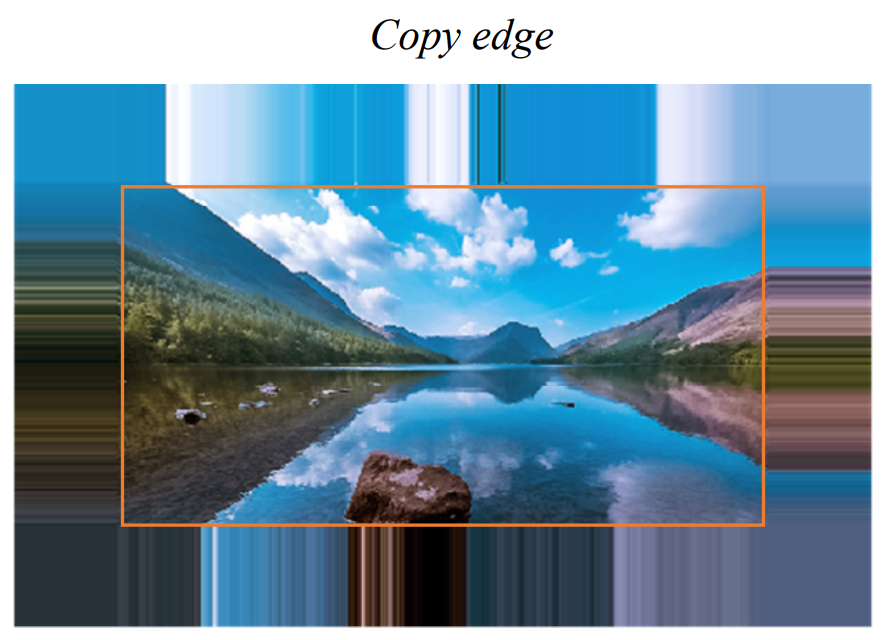
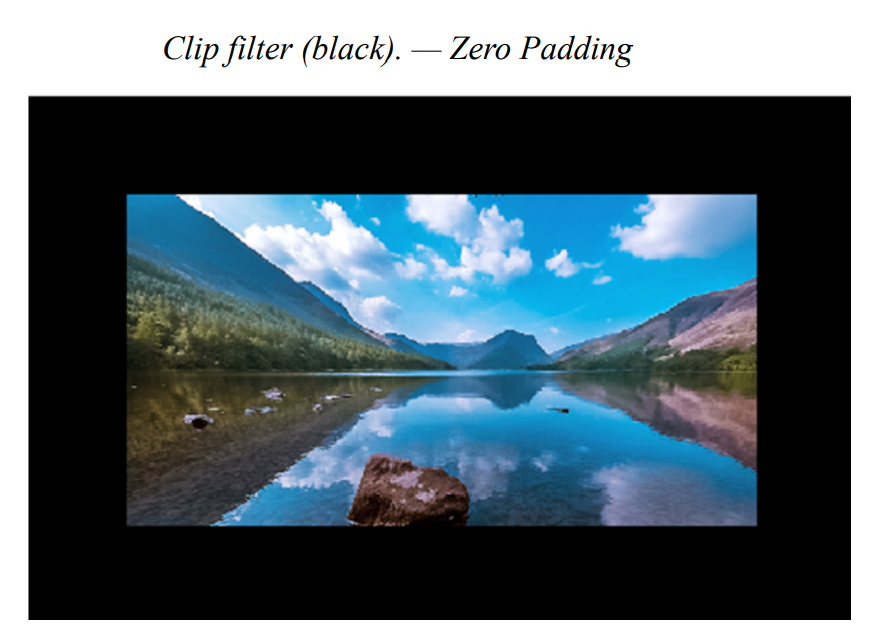
# 二、实验原理

## 2.1 图像的高斯滤波与Padding

高斯核设计原理：设二维高斯分布为两个独立的零均值，方差相同的一维高斯分布组成。以滤波核中心为原点，像素之间距离为1，将高斯核每个点转成坐标形式，带入高斯函数中，算出具体值后，进行归一化处理，使得加和为1。

图像的高斯滤波：将高斯核与图像进行卷积

Padding：具体使用Python的切片操作实现

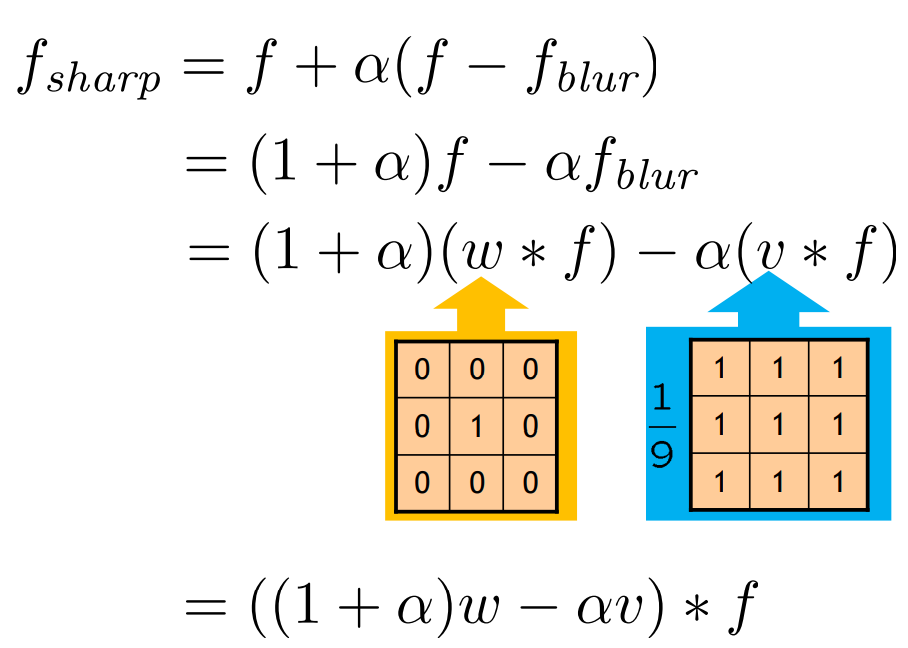


## 2.2 高斯核与高斯核卷积实验

与生成二维高斯核类似，将一维卷积核转换成坐标形式代入一维高斯分布并归一化，生成一维高斯核。将上述一维高斯核转置，生成第二个一维高斯核。利用卷积的交换律，使用两个一维高斯核依次对图像进行卷积。生成两个不同方差的二维卷积核进行相减，得到的新卷积核即可完成不同方差卷积核之差对图像滤波。

## 2.3 图像锐化滤波器

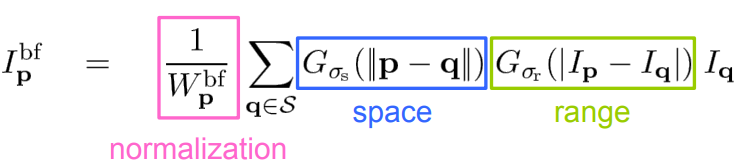
锐化原理为给图像加上α倍的图像与高斯滤波后的图像之差，具体操作如下图所示：



## 2.4双边滤波

由于高斯滤波会破坏边界信息，所以在卷积操作时，加入图像信息作为权重，即将中心点与进行卷积计算点的差值作为高斯函数的自变量，求得高斯函数值作为权重吗，即下图所示：

## 2.5 图像的傅里叶变换

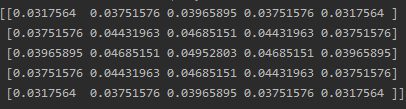


将图片像素值作为信号幅值，进行傅里叶变换。由于图片是二维的离散信号，所以使用（M，N）点的二维快速傅里叶变换。将频谱加上Mask，使其只保留高频成分，完成滤波。

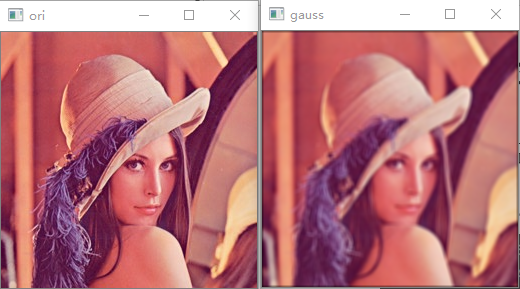
# 三、实验结果与分析

## 3.1图像的高斯滤波与Padding

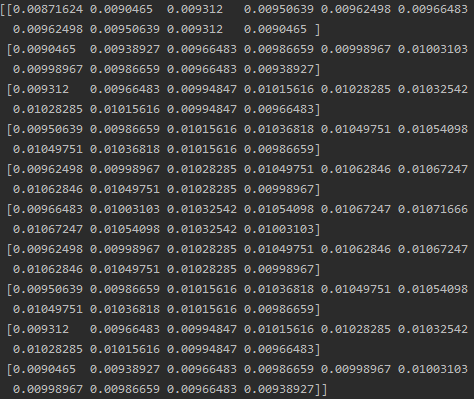
设置size = 5， sigma = 3，生成卷积核如下所示：



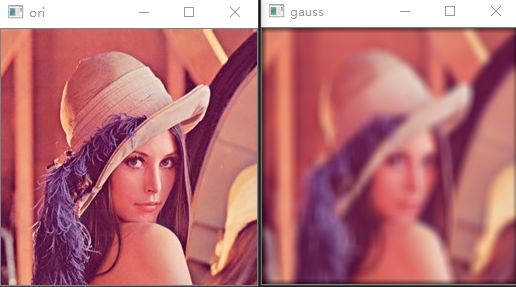
使用该核对图像滤波，卷积模式为same，padding为zero padding：



设置size = 11， sigma = 10，生成卷积核如下所示

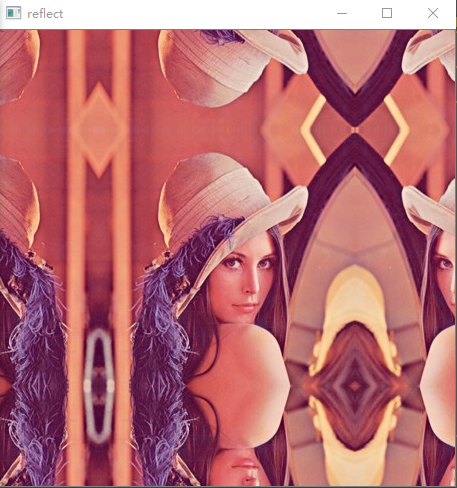
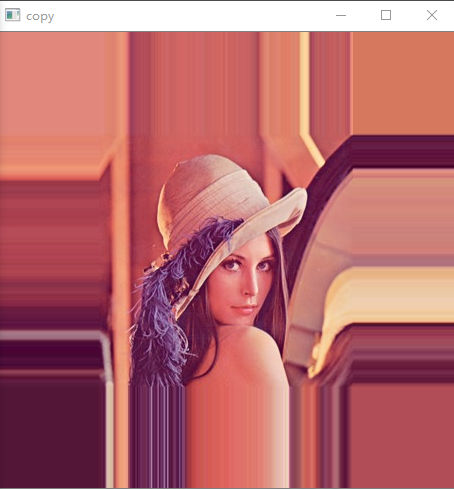
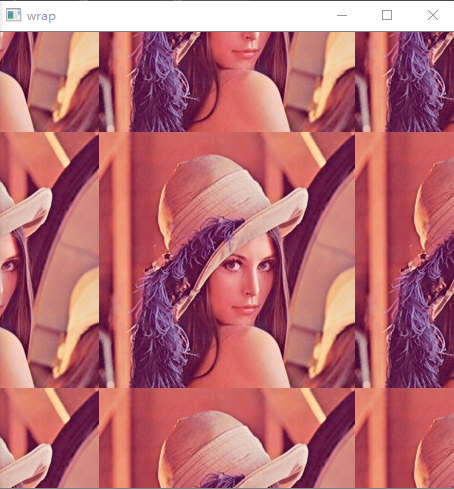
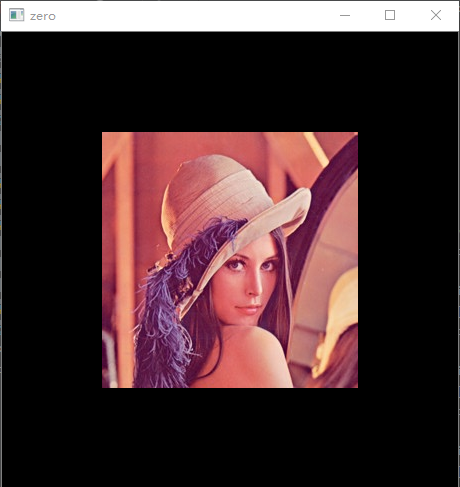


使用该核对图像进行滤波，卷积模式为same，padding为zero padding：



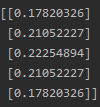
可以看到，由于使用的是zero padding，卷积核和sigma的越大，边缘越黑

Padding结果如下:

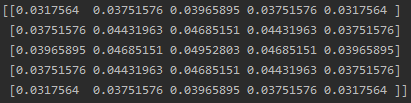


## 3.2 高斯核与高斯核卷积实验

生成size = 5， sigma = 3的一维卷积核如下所示

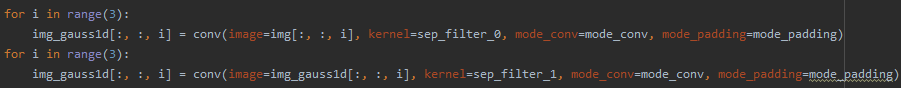


将其和它的转置进行full卷积，结果如下：

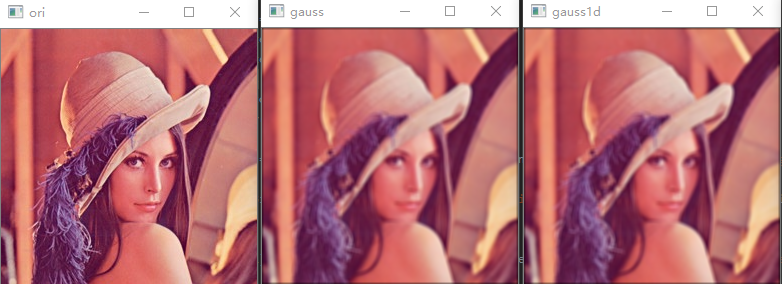


与3.1中生成的二维高斯卷积一致

将这两个卷积核分别对图像进行滤波：



与二维卷积、原图对比如下：



可见，与二维卷积核的滤波结果没有区别

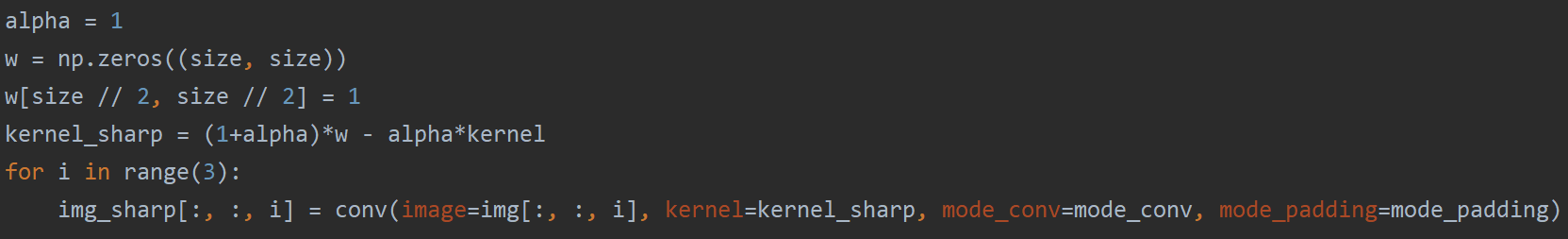
设置size=5，sigma分别为3和1的两个卷积核相减得到新卷积核，对图像进行滤波，结果如下：



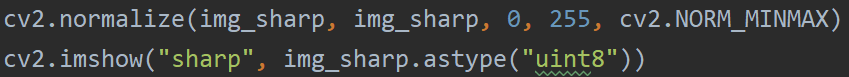
可见，使用高斯核之差进行滤波，起到的边缘提取的作用

## 3.3 图像锐化

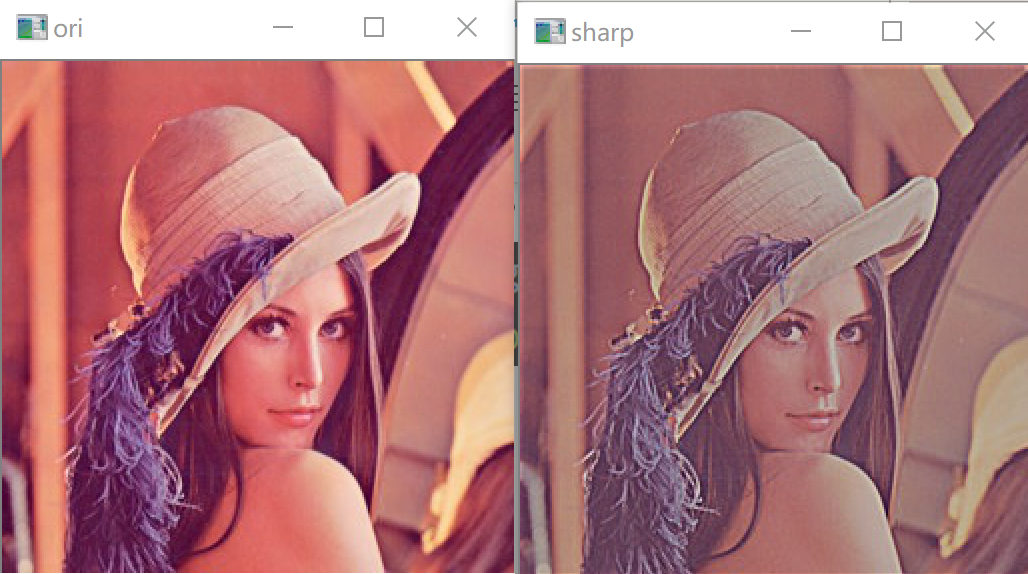
设置size = 5， sigma = 3的高斯核，设置α=1，形成新的锐化卷积核：



由于相加后会产生大于255和小于0的值，所以将图像标准化后输出：

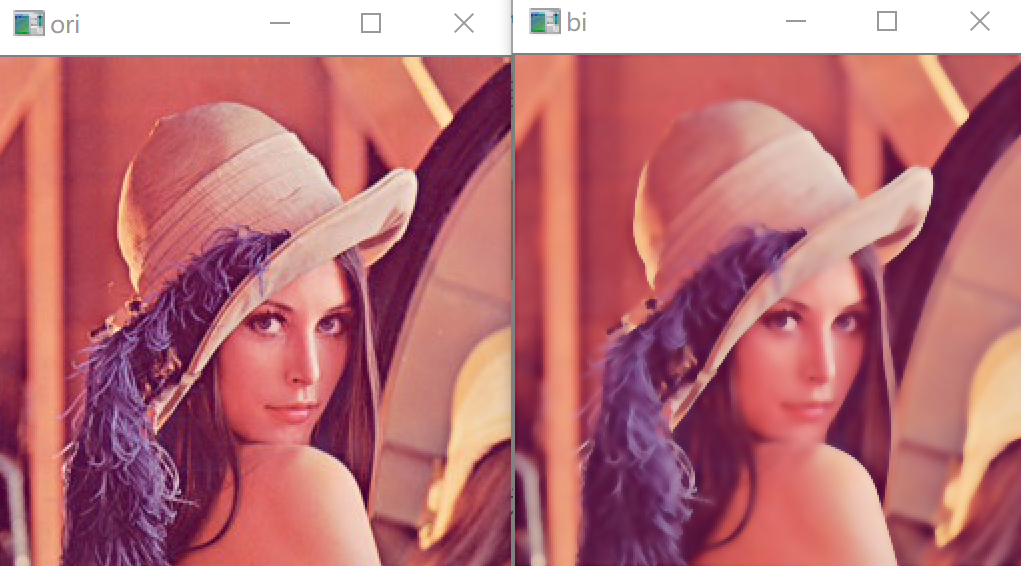


锐化结果如下图所示：



## 3.4 双边滤波

设置space权重的sigma为3，range权重的sigma为50，双边滤波结果如下：

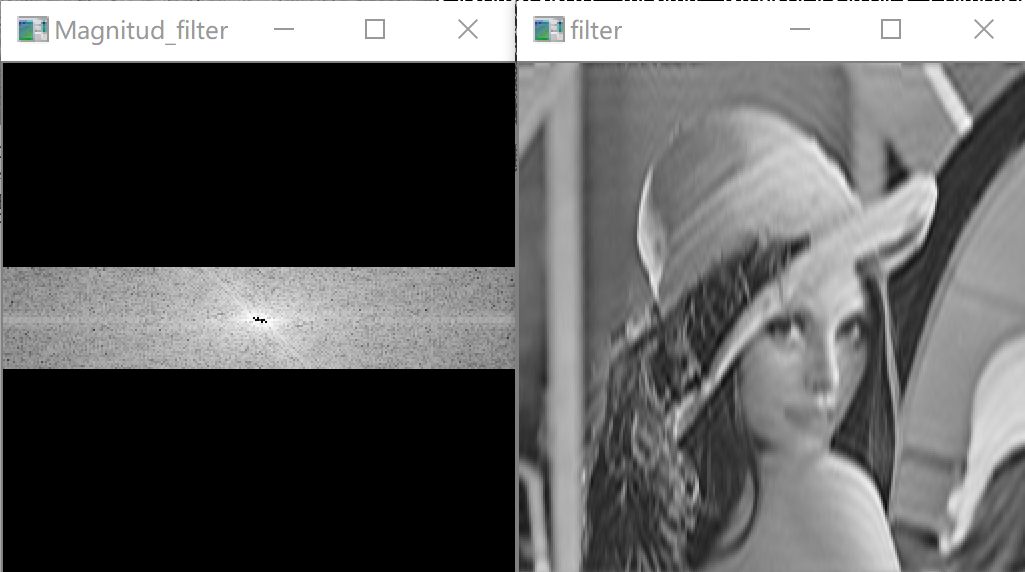


## 3.5 图像的傅里叶变换

由于自己手写傅里叶速度过慢，遂使用numpy的2维fft函数，变换后，为了方便观察，使用shift函数将低频部分移动至中心。但幅频特性输出后与实际不符，输出图像像素发现fft后的值大部分远超255，查阅资料得知，需要求幅频的分贝数进行输出，即给幅频加20log结果如下图所示：

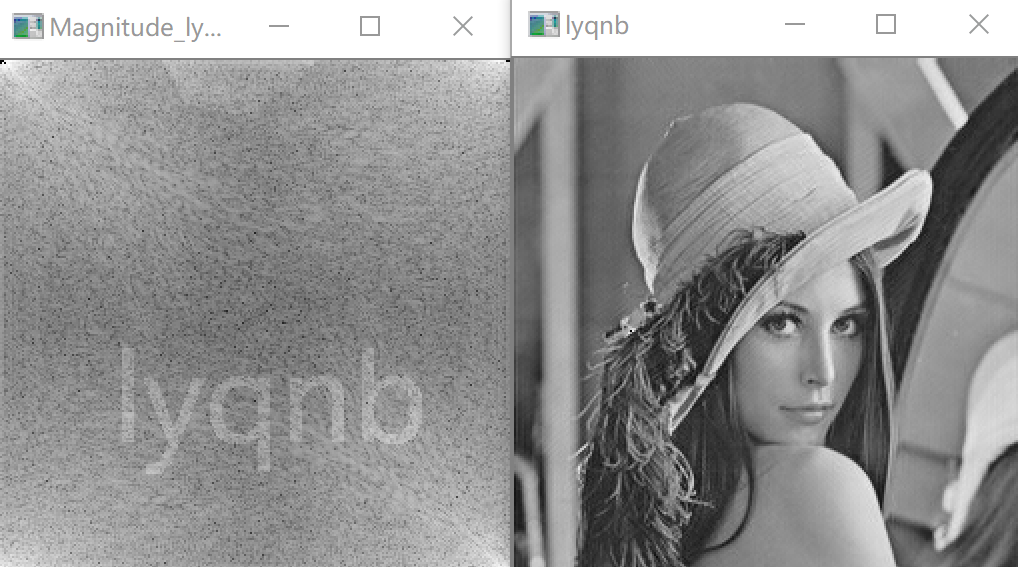


利用幅频进行滤波即给幅频添加一个Mask，使其舍弃高频部分，然后再通过傅里叶逆变换还原图像，结果如下图所示：



## 3.6 探索：频率水印

将频域加上水印后，再进行傅里叶反变换，即可得到带有频域水印的图片，下图为频域水印和逆变换后的图像



可见，在频域加入水印对原图并不会产生很大的影响，但当把该图片用傅里叶转成频域时，发生了有趣的事：



水印居然便成上下对称的了，分析得知，时信号为偶对称，由于频域转换为时域的过程中丢失了虚部，使得水印的对称分布，并且能量平分。

# 四、结论与讨论

## 4.1图像的高斯滤波与Padding

在高斯滤波中，经过选取不同的size与sigma发现，sigma越大，滤波后的图像越模糊，但sigma对图像的影响程度与size有关，size越大，影响程度越大，这是因为大的size会让图像去加权更多周围的点，使得sigma的作用大大加强。Padding与3种卷积方式的结合可以让边缘的处理更佳完善，除了zero padding外，其他三种padding都有很好的表现。

## 4.2高斯核与高斯核的卷积实验

经过实验验证，两个一维高斯核的卷积结果是具有相同sigma的二维高斯核，分步对图像进行卷积的结果也与用二维高斯核一致。在时间测试时发现，可分离卷积的计算速度并没有二维卷积快，与理论推导不符。经过多次改变参数，发现当卷积核的size很大时，可分离卷积的速度要远快与二维卷积，在较小时，例如size=3，并没有二维卷积快的原因是一维卷积需要做两次，而二维卷积只需要一次，较小的卷积核的对时间的影响没有计算次数的影响大。时间复杂度的推导会忽略掉常数倍数，所以导致在size较小时，速度与理论“不符”。在实验中，当size为3时，二维卷积的耗时为1s左右，可分离卷积的耗时为2s左右，当size为99时，二维卷积的耗时为5s左右，而可分离卷积的耗时仍为2s左右。

## 4.3 图像锐化

在实验中，alpha的选择常常会使图像像素值超过0~255的范围，此时将图像输出，便会有肉眼可见的失真，所以要将图像标准化——给RGB每层都减自己的最小值，使得下界为0，然后除最大值并乘255，这样便实现了标准化，但这样操作后，图像的色彩与原图会出现差异。

## 4.4 双边滤波

在实验中，首先使用了PPT中所给的sigma，但是滤完后发现图像和之前并没有区别，于是去掉range权重后再滤波，发现和高斯滤波结果相同，分析得可能是range权重的sigma取的过小，导致像素值稍有一点差异都会使得权重非常低，于是加大sigma到10，50，100后，发现双边滤波的效果与PPT一致，遂选择sigma = 50作为第三部分的演示图。

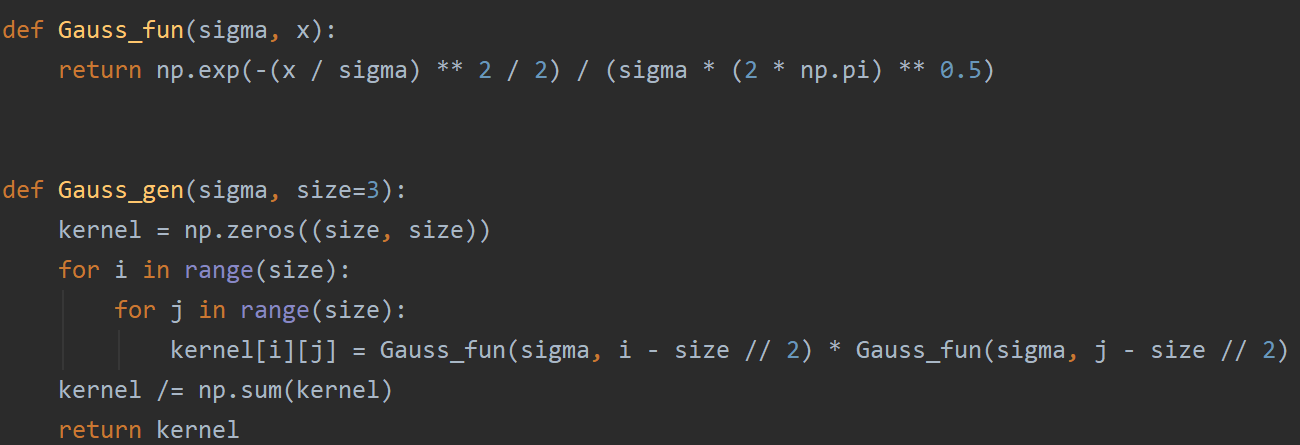
## 4.5 图像的傅里叶变换

图像可以看作是二维的离散信号，可以使用二维离散傅里叶变换，由于手写傅里叶太慢，于是使用numpy.fft.fft2实现变换。在对图像频域加Mask滤掉高频后，可以发现图像变得扭曲、模糊，这是因为图像精细程度是由高频信息决定的，舍弃后会变得粗糙、模糊。

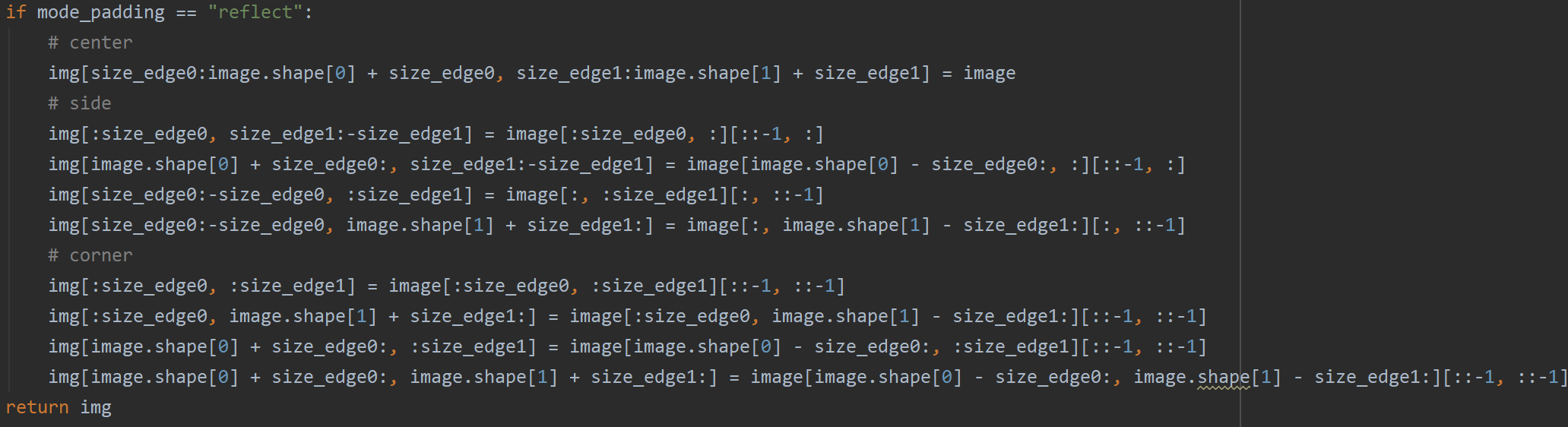
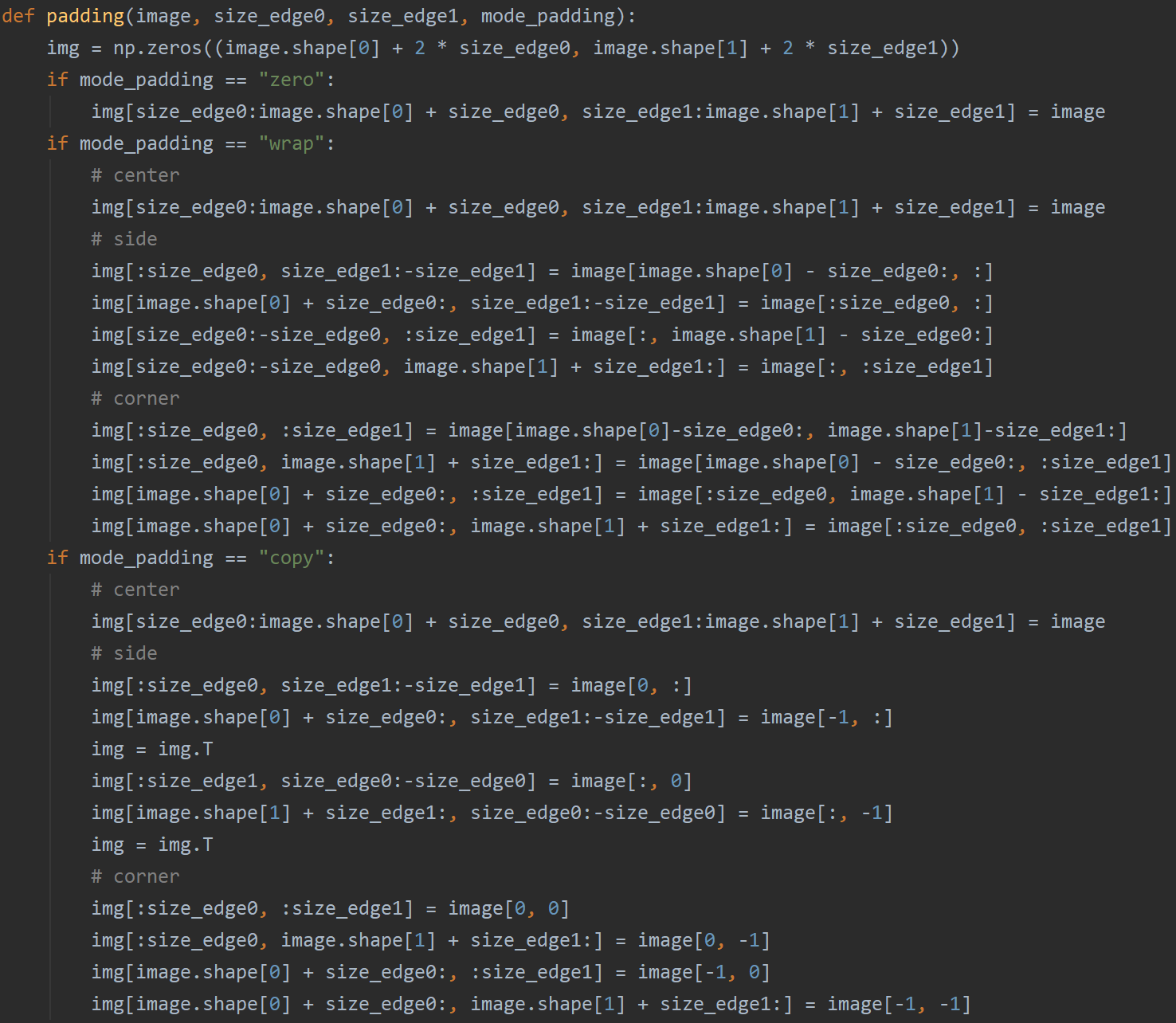
# 五、代码

## 5.1 图像的高斯滤波与Padding

高斯核生成函数：



Padding：



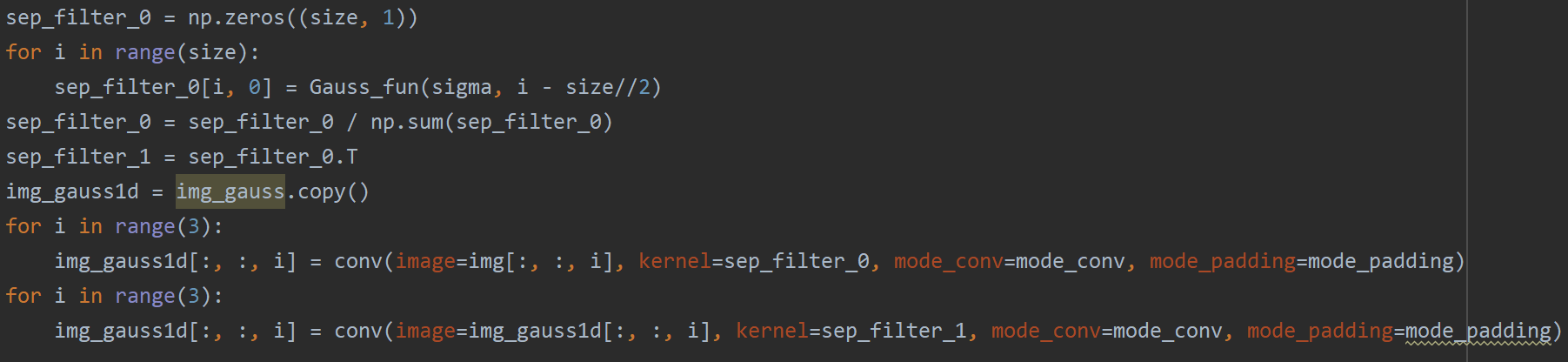
1/2D卷积：



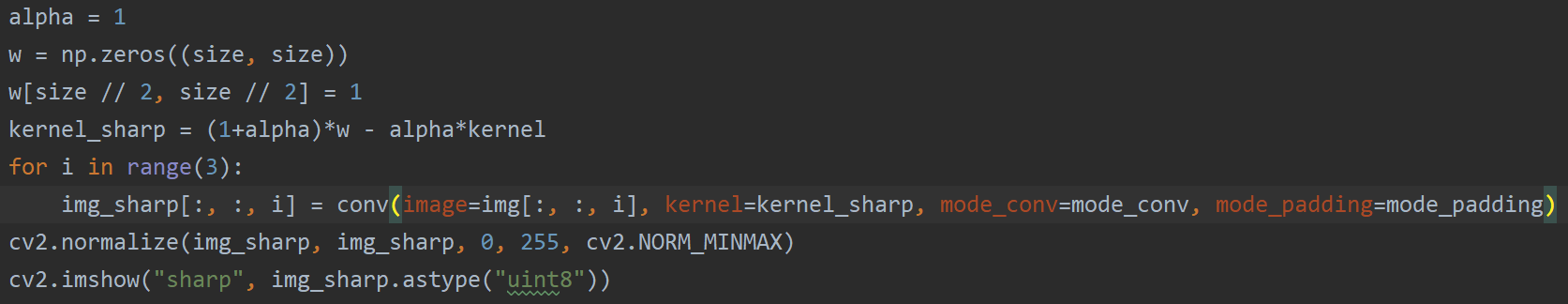
高斯滤波：



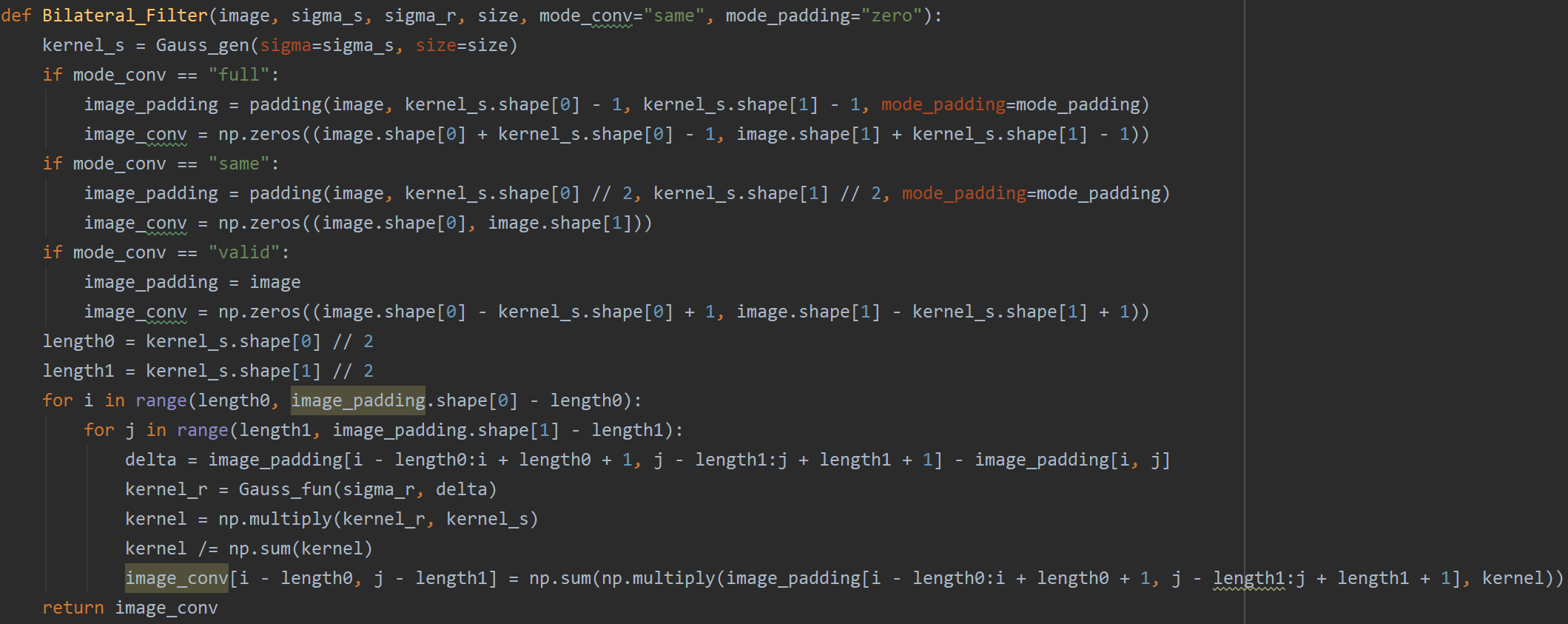
## 5.2 高斯核与高斯核的卷积实验



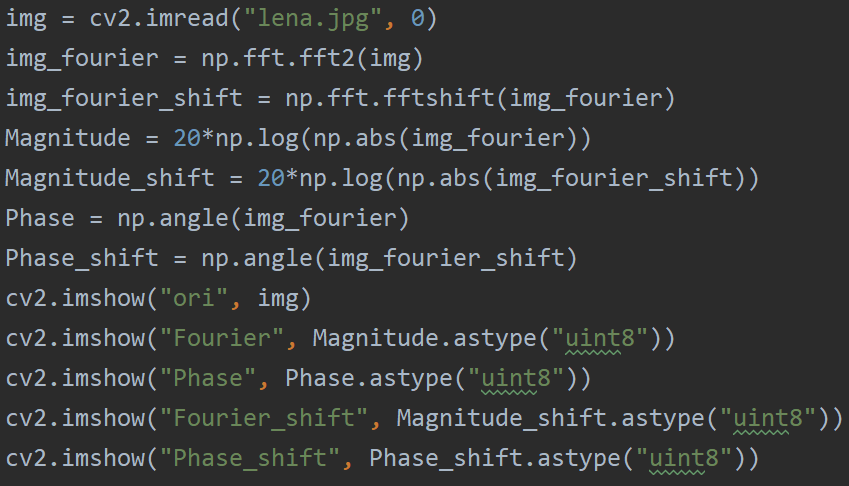
## 5.3 图像锐化



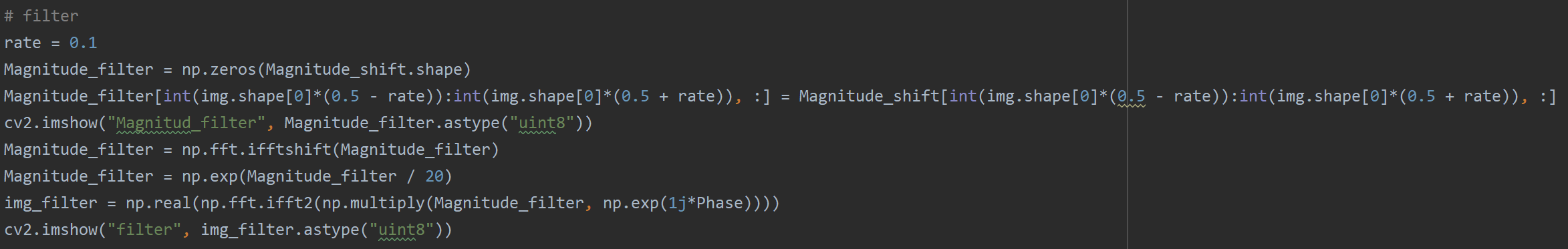
## 5.4 双边滤波



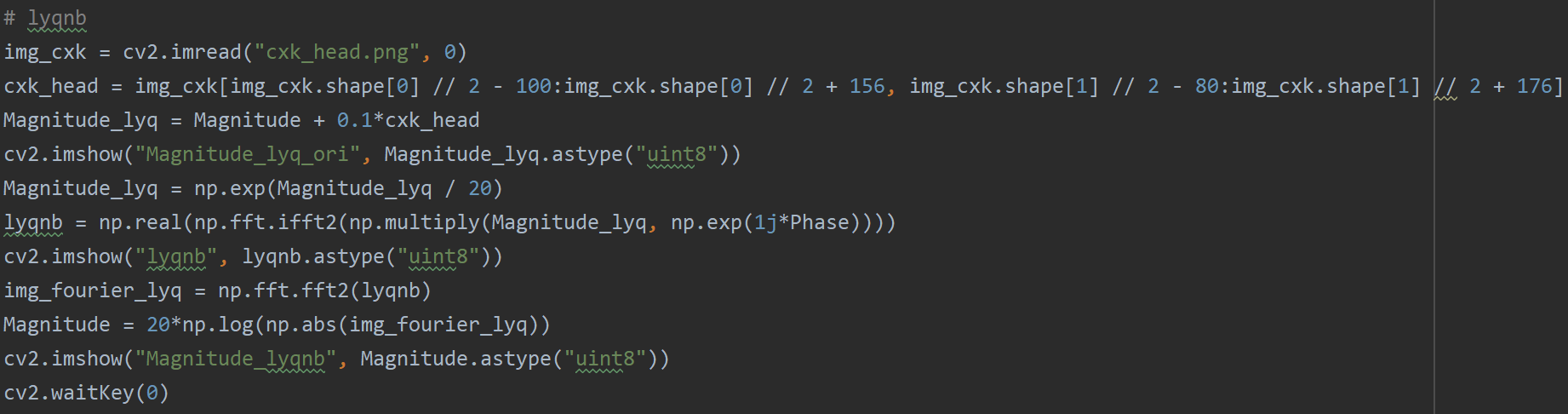
## 5.5 图像的傅里叶变换



频域滤波：



## 5.6 探索：频率水印



## 5.7 作业所有代码及图片地址

https://github.com/wyb2333/Computer-Vision-and-Pattern-Recognition