经过课程的学习，相信你已经深刻了解了数字图像中空间域与频域的含义。本次作业旨在让你掌握在频域对图像进行处理的基本方法：

### assignment1

实现教材图3.2中的频率域低通滤波，比较零填充的效果。其中b是不进行零填充，直接在频率域低通滤波的结果；c是预先进行零填充，然后在频率域低通滤波的结果。

图片包含 图示

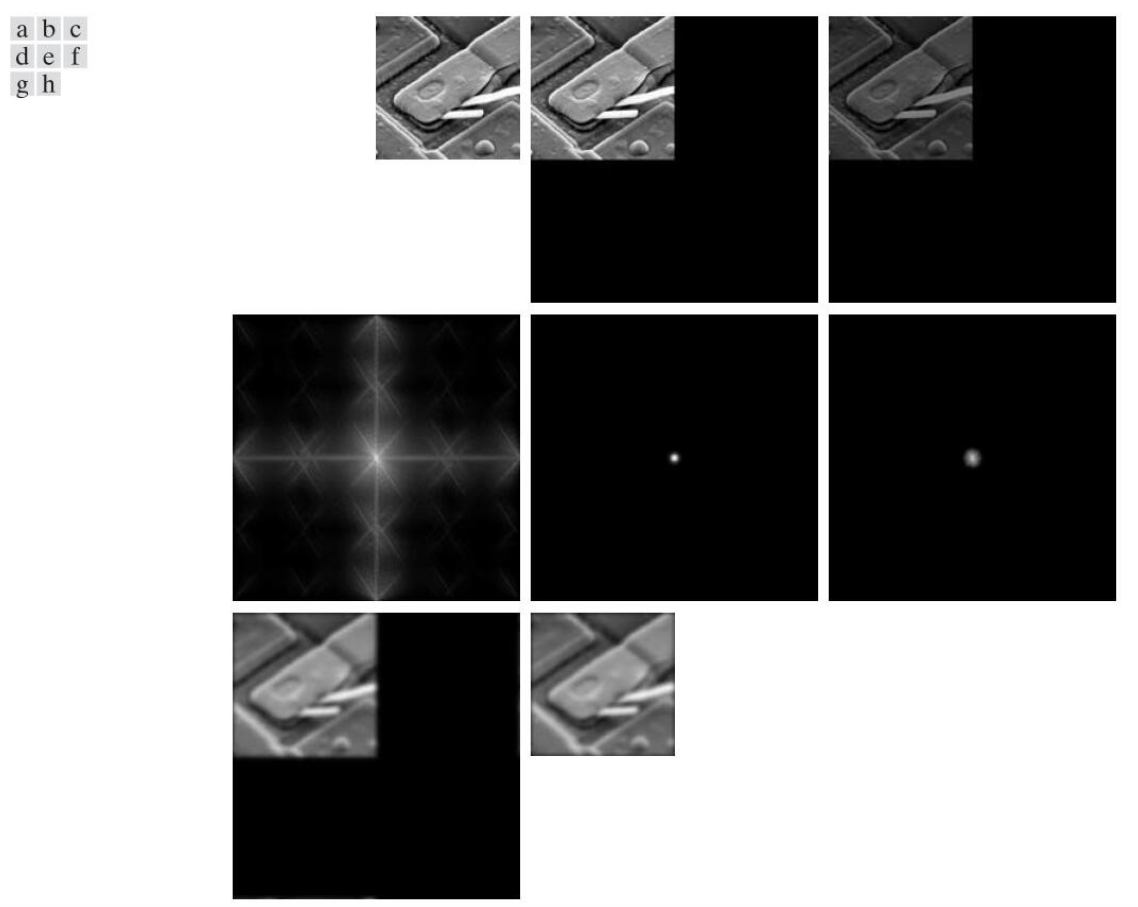
描述已自动生成

#### 作业要求

1. 你需要在文档里说明实现步骤，并对实验过程中遇到的问题进行阐述。
2. 可以使用fft，fft2，ifft2，fspecial函数，也可以尝试自己实现傅里叶变换。

### assignment2

实现教材图4.36中的滤波过程，生成从b到h的图像。



#### 作业要求

1. 你需要在文档里说明实现步骤，并对实验过程中遇到的问题进行阐述。
2. 可以使用fft，fft2，ifft2，fspecial函数，也可以尝试自己实现傅里叶变换。

### assignment3

仿照教材中图4.39，验证空间域滤波和频率域滤波的等价性。对于给定的空间域滤波器（如[sobel](https://en.wikipedia.org/wiki/Sobel_operator)算子），我们希望你能生成空间域滤波器对应的频域滤波器，然后基于你得到的频域滤波器在频域对图像进行对应的滤波操作，并且与在空间域直接滤波得到的结果进行对比。



图2为频域中滤波得到的结果，图3为直接在空间域滤波得到的结果。

#### 作业要求

1. 实现空间域滤波，生成图2；实现频率域滤波，生成图3。
2. 你需要在文档里说明实现步骤，并对实验过程中遇到的问题进行阐述。
3. 需要提交原图和算子在频域中的图片。
4. 实现频域滤波时可以使用fft，fft2，ifft2函数，但不允许使用conv2，imfilter，fftfilt等库函数（但你可以使用它们来观察效果）。
5. 我们鼓励你使用更多的图片与其他空间域算子来验证空间域滤波和频率域滤波的等价性，并会根据实验效果酌情加分。

需要注意的是，最后得到的结果与你的显示与存储方式有关，这无须在意。

**注意**

1. 本次作业不提供框架代码，你可以自由地实现你的方法，但你需要保证代码的可复现性。
2. 如果你使用python, c++等其他语言完成作业，允许使用numpy, opencv等工具包进行基本的数据与图像处理，相应库函数的使用权限同上，请仔细斟酌。
3. 作业 2 的截止时间是 12 月 1 日 23:59:59。
4. 如果发现作业之间高度相似将被判定为互相抄袭行为，抄袭和被抄袭双方的成绩都将被严肃处理。因此请主动防止自己的作业被他人抄袭。

提交邮箱地址：lelegan@163.com

### 提交格式

一个具有如下结构的 zip 文件: （--x表示名字为 x 的目录）

xxx.zip

--doc: 一个描述你的实验的思路,方法和实现的 pdf，doc或md 文件。

--assignment1

--code:

--asset:

--image: 你用来评价算法所有图像

--assignment2

--code:

--asset:

--image: 你用来评价算法所有图像

--assignment3

--code:

--asset:

--image: 你用来评价算法所有图像

请按照 **学号\_作业序号\_v版本号.zip** 命名提交文件，本次作业序号为2。版本号从1开始，例如学号为111220001的同学提交第一次作业的第一个版本，命名为 **111220001\_1\_v1.zip**（你的提交文件需要与这个文件的组织结构相同）。

批改作业将以作业提交期限前版本号最高的文件为准，如果最高版本文件毁损，将使用次高版本文件以此类推。如对本次作业存在问题，欢迎邮件联系助教。