

电子科技大学

实验报告

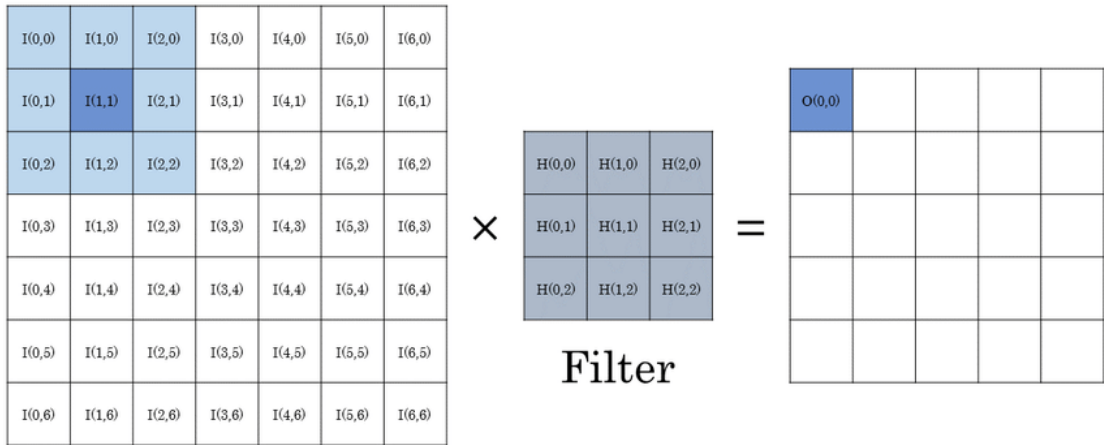
学生姓名：蔡与望 学号：2020010801024 指导教师：徐行

一、实验项目名称：图像过滤与融合

二、实验原理：

2.1 图像过滤

从空域上来看，图像过滤是用卷积核对图像进行一次卷积，如下图所示。



Input image

Output image

从频域上来看，图像过滤是一种过滤频率成分的方法，最典型的有低通和高通滤波。

分类	定义	用途	典型
低通滤波	只允许低频成分通过	平滑、降噪	均值、高斯
高通滤波	只允许高频成分通过	锐化、边缘检测	索伯、拉普拉斯

2.2 图像融合

图像融合，就是将一幅图像的低频成分和另一幅的高频成分叠加。在对两幅图像分别进行滤波时，截止频率需要相同。

同时，想要让图像融合的效果更好，就要让图像的结构类似，例如两幅都是脸部的照片。

三、实验目的：

- 1. 掌握图像过滤的原理和方法。
- 2. 掌握图像融合的原理和方法。

四、实验内容：

- 1. 基于 `scikit-image` 中所包含的常见滤波器，对图像进行滤波，直观地对比不同滤波器的效果。
- 2. 编写 `my_filter` 函数，实现高通滤波和低通滤波两种滤波形式。
- 3. 通过 `my_filter` 函数对图像进行高低通滤波，并且对图像分别在高低频进行融合。

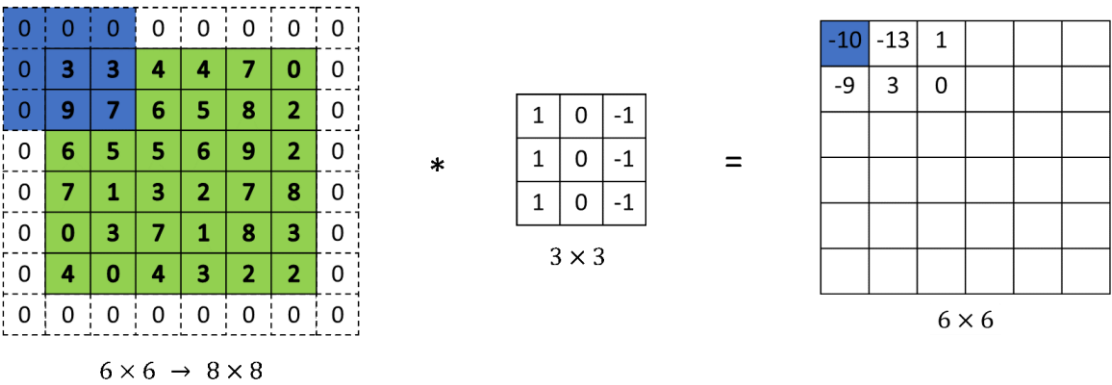
五、实验步骤：

5.1 图像过滤

代码见 `package/filter.py`。

首先，将卷积核翻转 180° 。虽然大多数（包括该实验中所有）的卷积核都是上下左右对称的，但如果不翻转，这个核做的就不是卷积运算，而是互相关运算。

接着，为了卷积核能够更方便的处理边缘像素，我们在原图像的周围用 0 扩充，如下图所示。



然后，对于每个颜色通道，遍历每行每列，计算每个像素的卷积结果。

5.2 图像融合

代码见 `package/hybrid.py`。

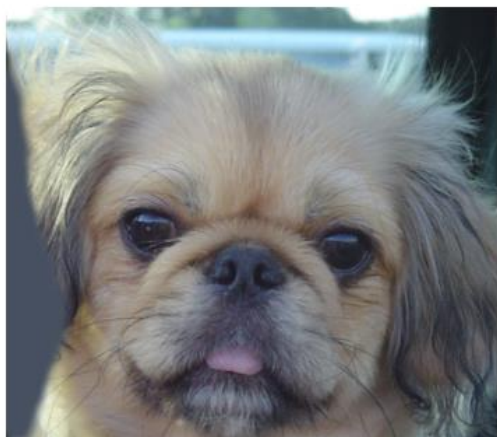
首先，创建一个高斯核，其标准差和半径均为截止频率。

接着，用高斯核过滤图像 1 和图像 2，得到它们的低频成分。从图像 2 中减去它自己的低频成分，就得到了图像 2 的高频成分。

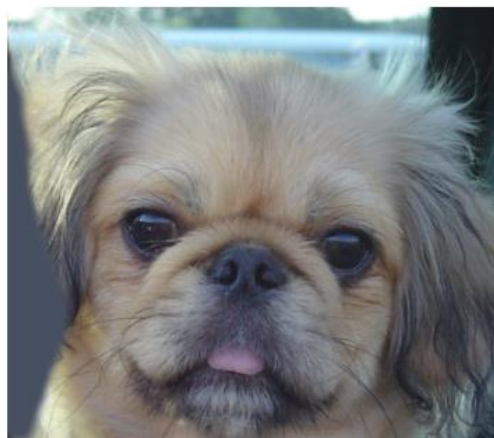
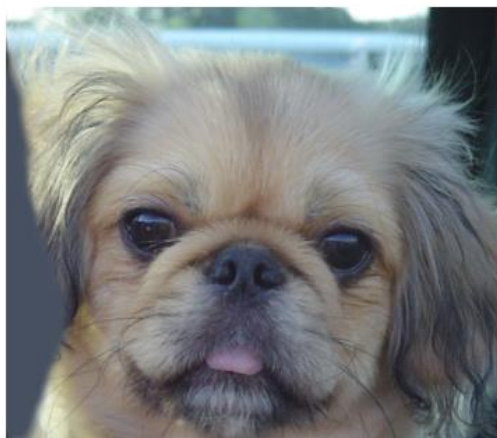
最后，将图像 1 的低频成分和图像 2 的高频成分相加，得到融合图像。

六、实验数据及结果分析：

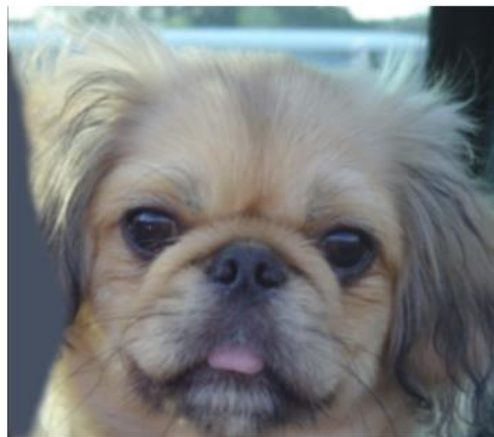
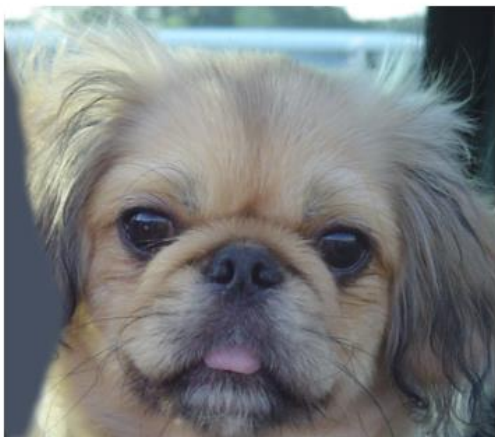
实验使用的两幅图像分别是狗和猫的脸部照片。



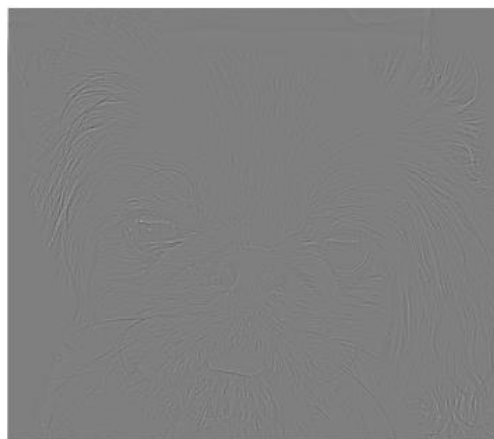
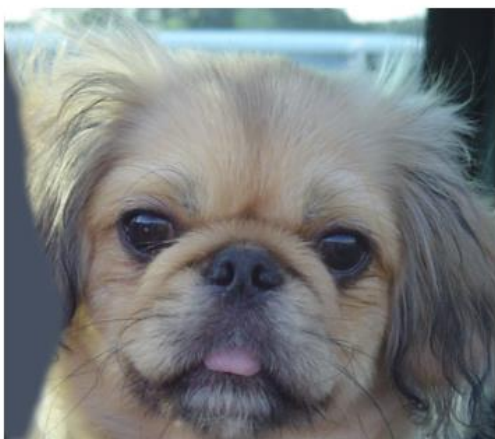
使用一个只有中心为 1、其它均为 0 的核滤波，结果与原图一致。这是因为，原图中的每个像素都变成了自己乘 1，相当于没有变。



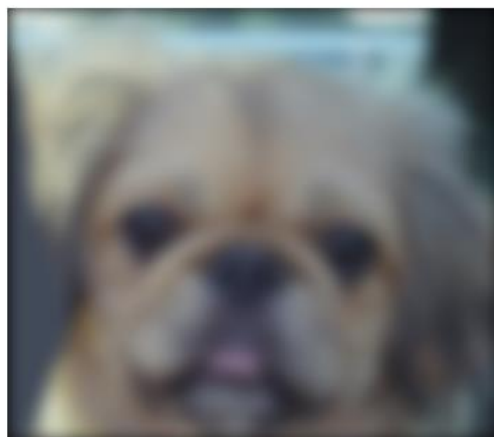
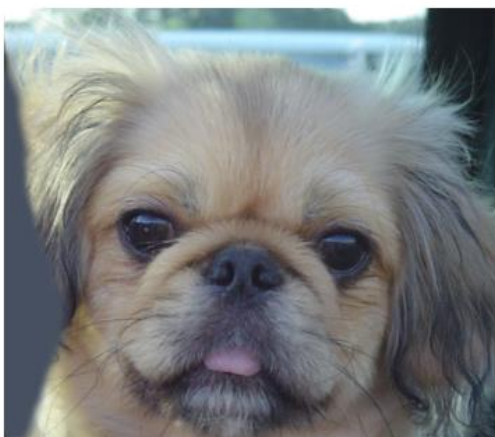
均值滤波将原图变得更模糊。这是因为，原图中的每个像素都变成了周围像素的均值，相当于在图像的每个地方都做了平滑处理。



将原图减去均值滤波的结果，得到了类似于高通滤波的结果。这是因为，原图减去其低频成分，自然就等于其高频成分。

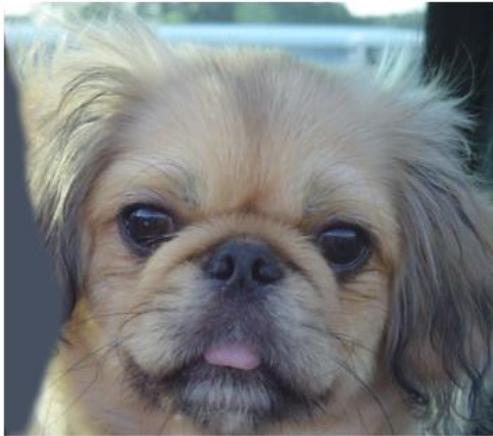


高斯滤波也将原图变得更模糊。这是因为，高斯核在两个维度上都是正态分布的向量，相当于一种考虑了邻近程度影响的均值滤波。

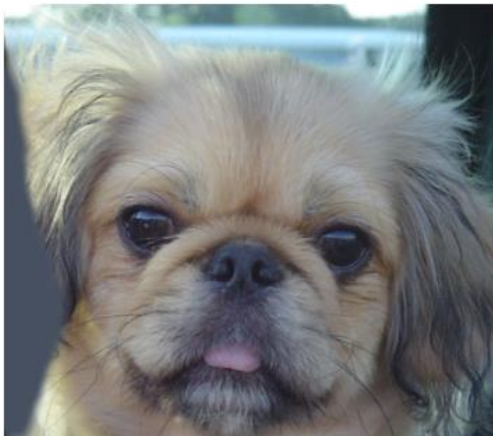


索伯滤波检测出了图像的垂直（水平）边界。这是因为，索伯算子相当于是把像素一侧的值减去了另一侧的值。如果像素不在边界上，那么其两侧的颜色不会差很多，得到的卷积结果也接近于 0。而对于边界上的像素而言，两侧颜色有明显差异，所以索伯算子把边界的

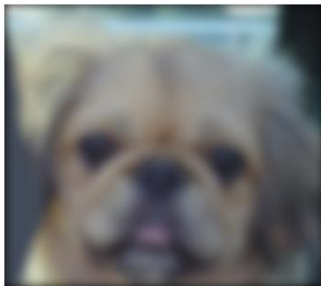
像素点都检测了出来。



拉普拉斯滤波也检测出了图像的边界。这是因为，拉普拉斯算子相当于从像素周围的值中减去了该像素的值。与索伯滤波同理，它只在边界像素上的卷积结果不接近于 0。



令截止频率为 7，融合狗图的低频成分和猫图的高频成分，如下图所示。



七、实验结论：

从结果中可以看出，图像过滤和图像融合的效果都比较令人满意。因此可以说，本实验中编写的代码能够很好地完成实验要求。

八、总结及心得体会：

通过本实验，我对图像过滤和融合的原理有了清晰的认知，对如何使用代码完成这些操作有了亲身的体验。同时，我通过分析卷积核的特点，探究了各种过滤效果的产生原因。

九、对本实验过程及方法的改进建议：

无。

报告评分：

指导教师签字：