1 习题

请完成下列问题，并将你的答案写到报告中。

1.1 直方图均衡化 （15 分）

假设你对一张图已经进行了一次直方图均衡化的操作。如果对这张图进行第二次直方图均衡化，得到的结果跟第一次均衡化的结果一样吗？请给予证明。

Ans:一样

证明：若图像灰度级 L，$P_r\left(r_j\right)$ 为原图灰度值 r_j 的概率密度，$p_s\left(r_j\right)$ 表示均衡化变换后灰度值 r_j 的概率密度。

$$S_k = T\left(r_k\right) = \left(L-1\right)\sum_{j=0}^{k} P_r\left(r_j\right), k = 0,1,2,\cdots,L-1$$

对输入图像中每个具有 r 值的像素值产生一个输出灰度值 s，变换 T 保证函数在区间[0,L-1]上为一个单调递增函数。灰度变换不改变灰度的相对大小，也不改变其概率分布。

剔除概率密度为 0 的灰度值，得到长为 x 的灰度序列 A。

可知灰度均衡化前后得到的灰度序列 A 相同。

$$\sum_{j=0}^{k} P_r\left(r_j\right) = \sum_{j=0}^{k} P_s\left(r_j\right), k = 0,1,2,\cdots,x-1$$

可知第二次直方图均衡化后得到的结果和第一次均衡化后结果一样。

1.2 空间滤波（20 分）

给定一张 4 x 4 的灰度图和一个 3 x 3 的滤波器：

$$\textit{图像：}\begin{bmatrix} 85 & 13 & 20 & 80 \\ 169 & 8 & 243 & 20 \\ 18 & 155 & 163 & 44 \\ 12 & 34 & 50 & 80 \end{bmatrix} \quad \textit{滤波器：}\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

1. （7 分）用给定的滤波器对这张灰度图（边界补零）进行卷积，写出卷积后的结果（大小应为 4 x 4）。

   Ans:

   图像：$\begin{bmatrix} 85 & 13 & 20 & 80 \\ 169 & 8 & 243 & 20 \\ 18 & 155 & 163 & 44 \\ 12 & 34 & 50 & 80 \end{bmatrix}$ 滤波器：$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$

补零：$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 85 & 13 & 20 & 80 & 0 & 0 \\ 0 & 0 & 169 & 8 & 243 & 20 & 0 & 0 \\ 0 & 0 & 18 & 155 & 163 & 44 & 0 & 0 \\ 0 & 0 & 12 & 34 & 50 & 80 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$ 卷积结果：$\begin{bmatrix} -177 & -420 & -271 & -263 \\ -75 & -218 & -249 & -107 \\ 131 & 324 & 107 & 133 \\ 173 & 336 & 362 & 207 \end{bmatrix}$

2. （8 分）请说出你得到的卷积结果中正数和负数分别表示什么含义。

Ans: 用此掩模对原矩阵进行卷积，通过计算 y（竖直）方向的梯度，检测 x（水平）方向的边缘。

正数表示此点处梯度大于 0，负数表示此点处梯度小于 0。

可以明确地反映出图像灰度值在 y 方向上的变化情况。

3. （5 分）根据你所学到的知识，谈一谈题目中给出的 3 x 3 滤波器可以有哪些应用。

Ans: 可以对图像进行边缘检测，找出边缘后可以进行对应的边缘增强，图像锐化。

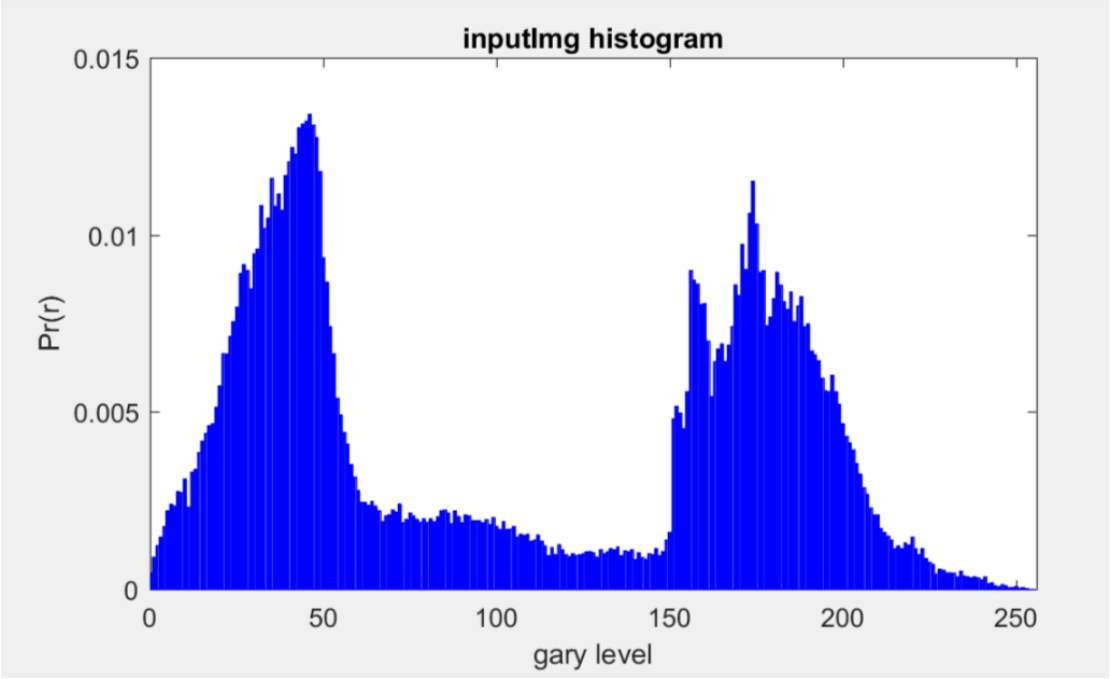
2 编程题

2.2 直方图均衡化（35 分） 实现一个对灰度图做直方图均衡化的函数（不允许直接调用现成的直方图均衡化接口，例如 Matlab 的 “histeq” 接口）。函数的格式为 “equalize_hist(input_img) output_img”，该函数返回一张灰度 级分布均匀的灰度图。如有必要，你可以修改该函数的格式。

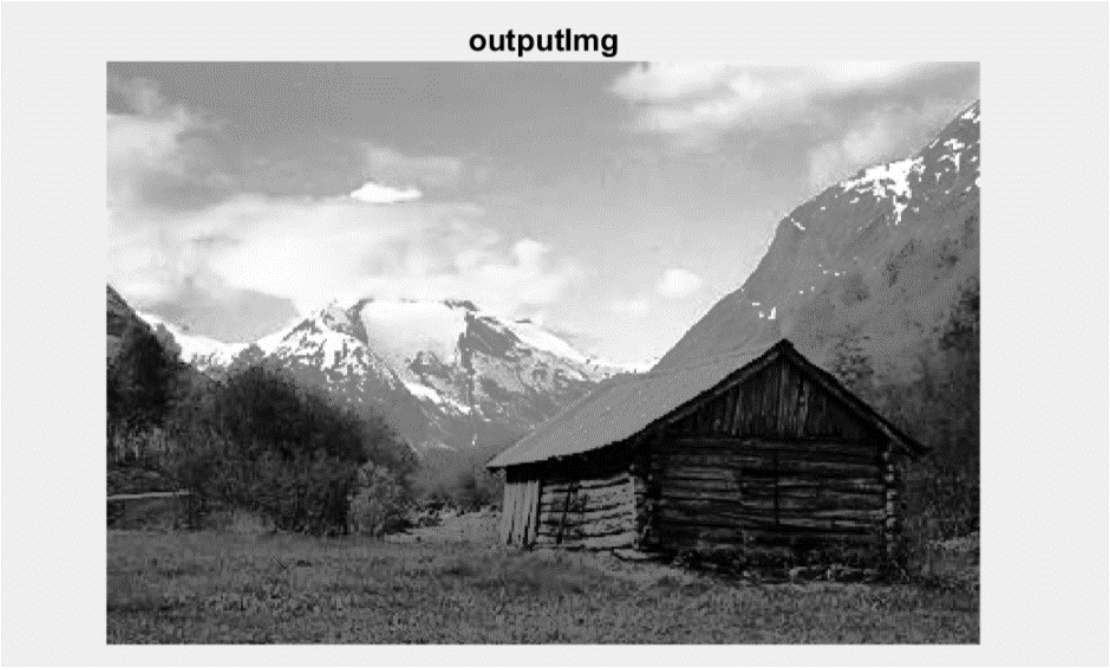
请载入对应你学号的输入图像，并用你实现的程序来完成以下任务：

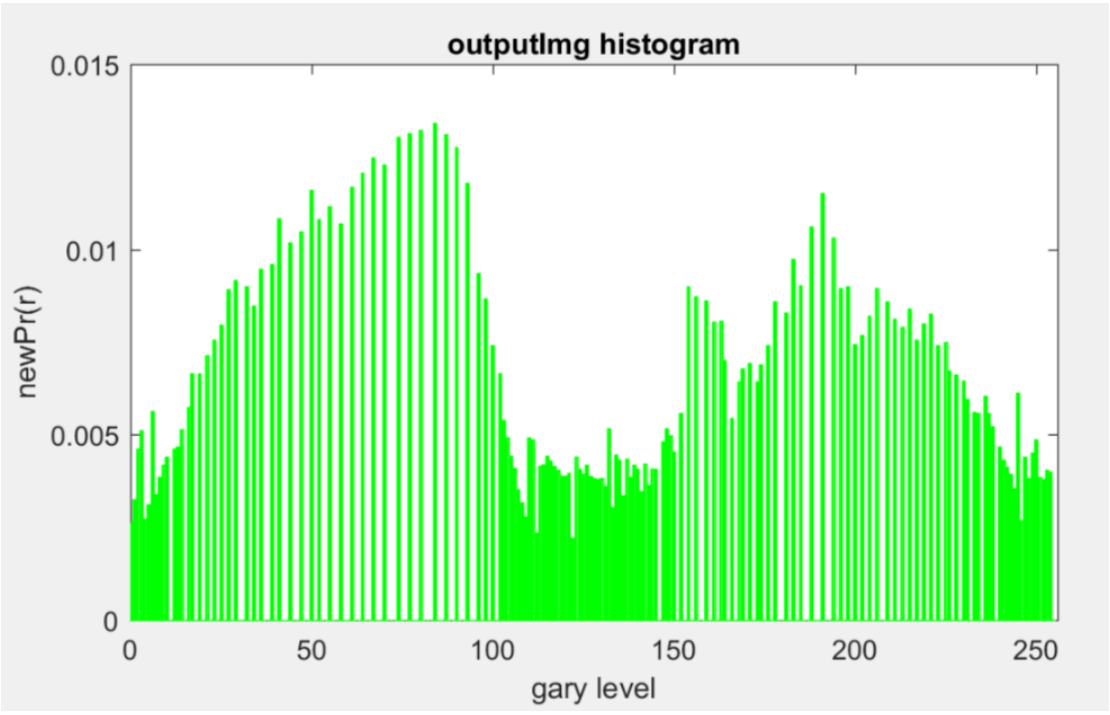
Run equalize.m

1. （5 分）计算并显示图像的直方图，并把结果粘贴到报告里。注意：你必须用你自己实现的函数 来计算直方图，但是允许调用现成的 API 来显示直方图。(例如，你不能调用 Matlab 的“imhist” 来计算直方图，但是可以调用“subplot”, “hist”来显示直方图。)

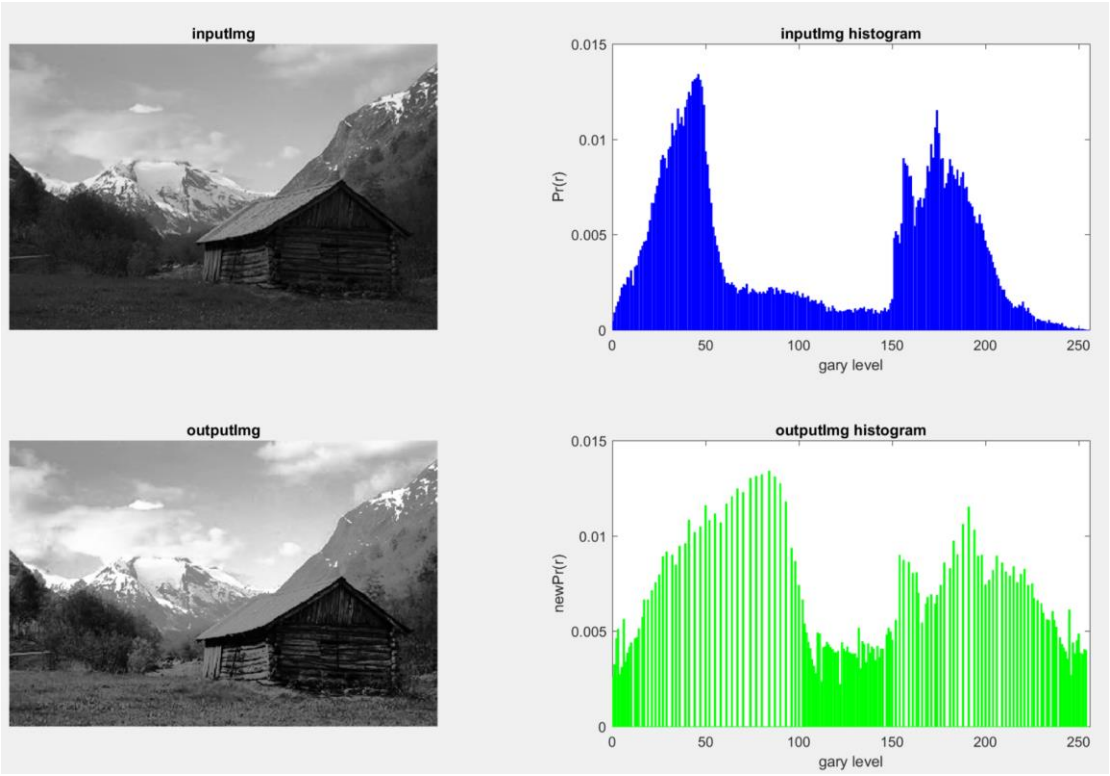


2. （10 分）进行直方图均衡化，将均衡化后的结果和相应的直方图粘贴到报告里。



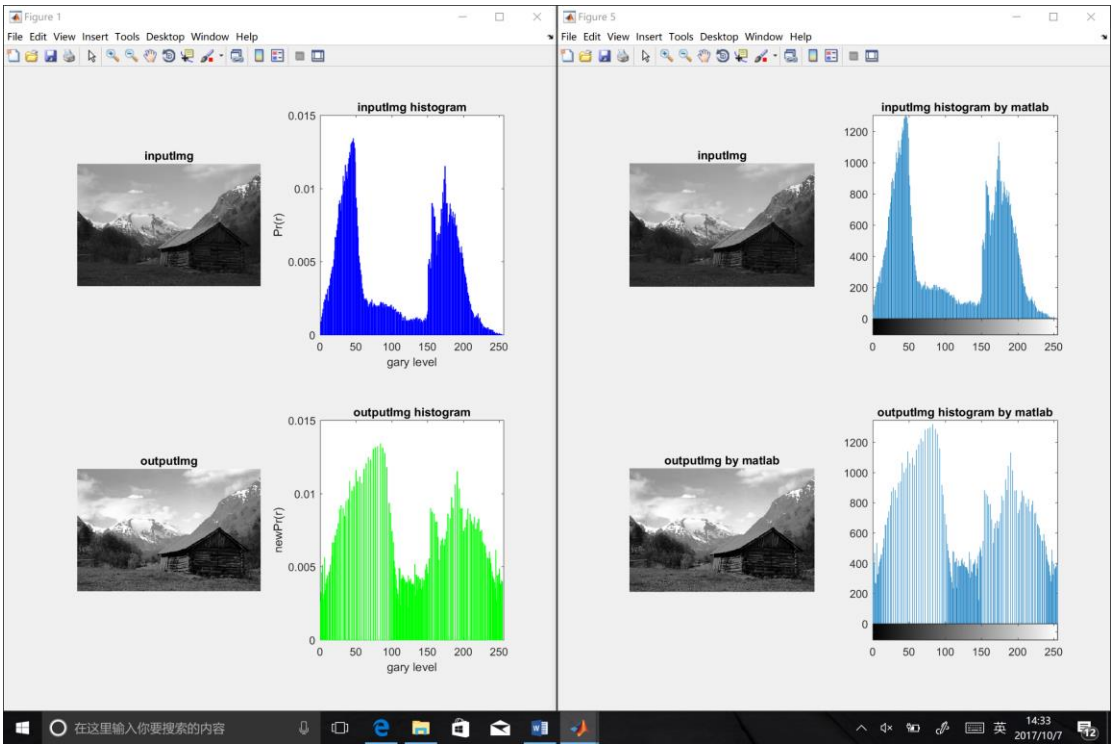


3. （8 分）分析直方图均衡化后的结果，字数不能超过一页。



从直方图来看：原图中直方图的分量集中在灰度级的较低端和较高端，变换后直方图倾向于占据整个可能的灰度级而且分布较为均匀。

从图像处理结果看：原图暗区和亮区对比强烈而且内部细节不明显，变换后图像有高对比度的外观，并可以展示灰色调的较大变化，显示出丰富的灰度变化和详细的细节信息，最终效果是一幅灰度细节丰富且动态范围较大的图像。



自定义函数运算结果和 matlab 中 imhist，histeq 函数运算结果对比

4. （12 分）详细描述你是如何实现直方图均衡化操作的，也就是说，针对“equalize_hist”函数进行算法说明，字数不能超过两页。请集中在算法描述方面，不要过多地复制/粘贴代码到报告上。

直方图均衡化算法说明：

对于离散的灰度值，我们处理其概率（直方图值）与求和来代替处理概率密度函数与积分。

一幅数字图像中灰度级 r_k 出现的概率近似为 $p_r(r_k)=\frac{n_k}{MN}, k=0,1,2,\cdots,L-1$

M,N 分别为图像矩阵的行数和列数，n_k 是灰度为 r_k 的像素个数，L 是图像中可能的灰度级的数量。

变换 $S_k=T(r_k)=(L-1)\sum_{j=0}^{k}P_r(r_j)=\frac{L-1}{MN}\sum_{j=0}^{k}n_j, k=0,1,2,\cdots,L-1$

可以由原图像灰度值的概率密度分布直方图计算得到变化后对应的灰度值和对应每个变换后灰度值的新的概率密度分布，以此得到均衡化后的直方图和输出图像。

1) 直方图显示

新建两个 1 行 256 列的零矩阵 Nk 和 Pr。

Nk 存储原图 0 到 255 个灰度级（rk）分别对应的像素个数，Pr 存储每个灰度级的概率密度。

由 Pr 可直接得到原图的概率密度直方图。

2) 直方图均衡化

利用变换公式计算出每个灰度值对应的变换后灰度值。

由概率密度 Pr 计算累计分布，并乘以系数 L-1=255，可以得到近似的灰度值 Sk。

因为概率密度求和结果基本为小数，需要对得到变换后的灰度值四舍五入近似为最接近的整数。

新建一个 1 行 256 列的零矩阵 newPr，存储均衡化后的灰度值对应概率密度。

遍历求和得到每个新的灰度值对应的概率密度：遍历过程中对每一个灰度值 i，只要变化后灰度值为 i，由其对应的原灰度值的概率密度求和得到灰度值 i 的概率密度 newPr(i)。

3) 输出原图像

遍历图中每一个像素点，把原图中像素点灰度值由转换函数转为均衡后灰度值，得到输出图像。

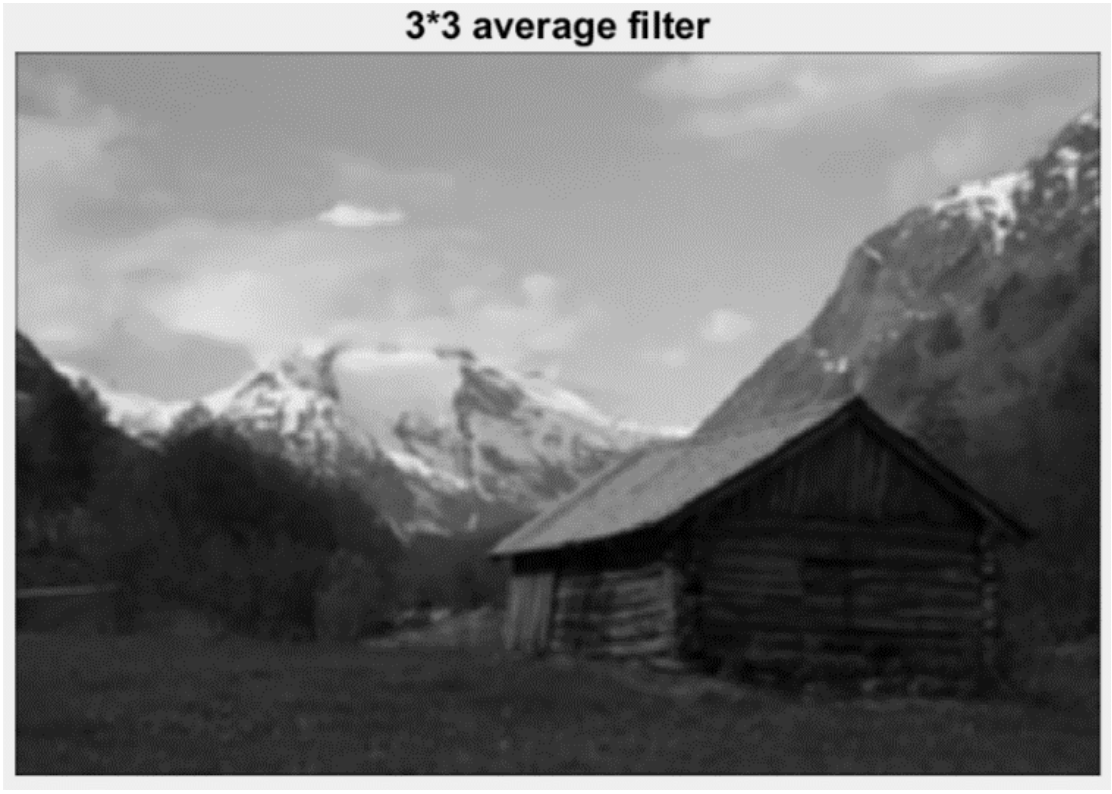
2.3 空间滤波（30 分）

实现一个对灰度图进行空间滤波的函数。函数的格式是“filter2d(input_img, filter) output_img”， 这里的“filter”是给定的滤波器。如有必要，可以修改函数的格式。 请载入对应你学号的输入图像，并用你实现的“filter2d”函数来完成以下任务：

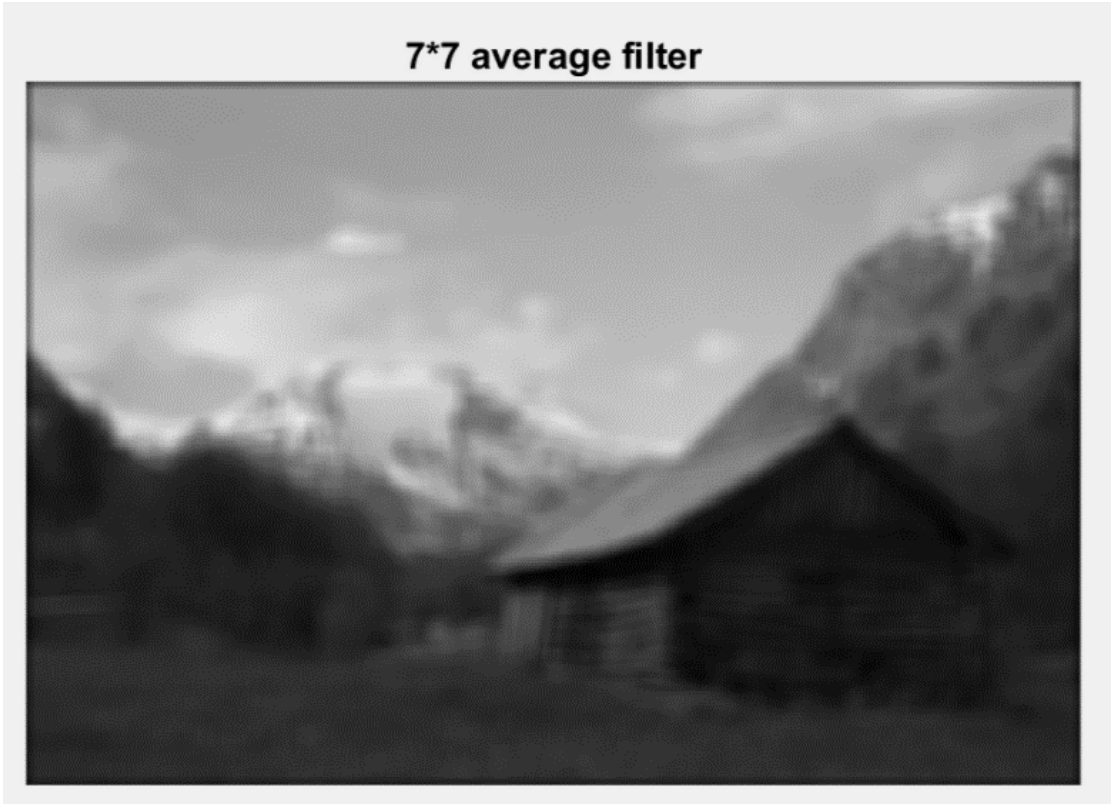
1. （9 分）分别用 3 x 3，7 x 7 和 11 x 11 的均值滤波器来平滑你输入的图像，将相应的三个输出结 果粘贴到报告里。

Run average_filter.m

3 x 3 均值滤波：

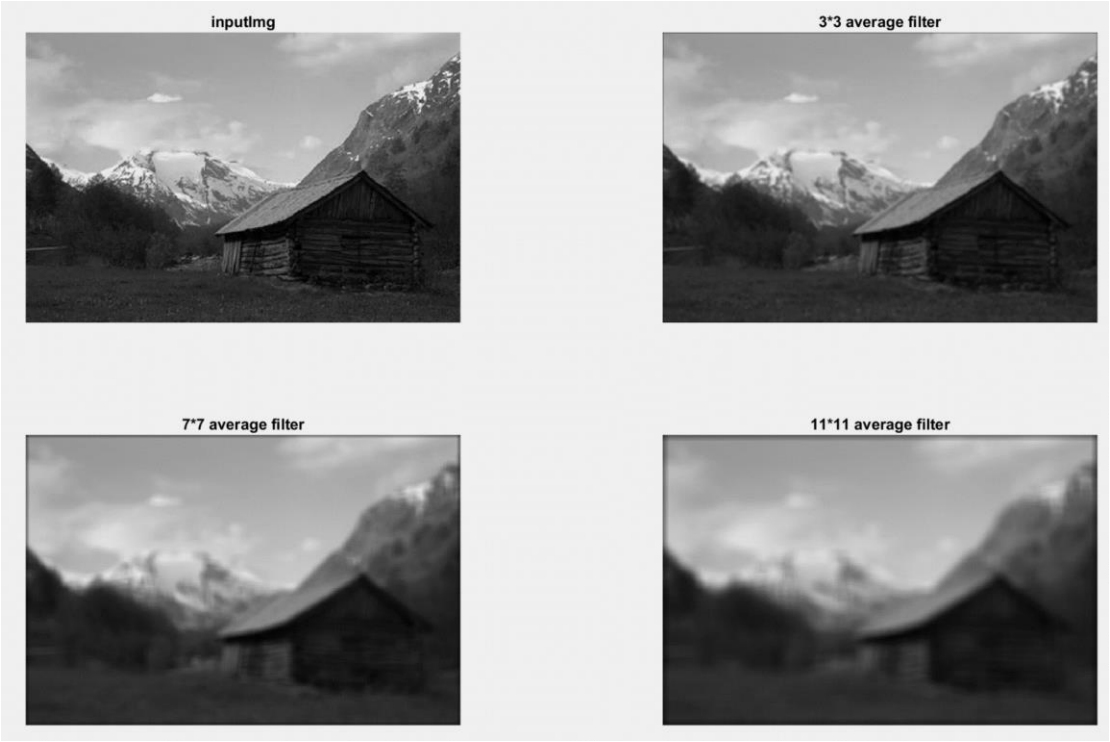


7 x 7 均值滤波:



11 x 11 均值滤波:

和输入图像的对比：



2. （6 分）用 3 x 3 的拉普拉斯滤波器来锐化你输入的图像（课本上有 4 种拉普拉斯滤波器，参见图 3.37，你可以使用其中任意一种），并将输出结果放在报告中。除此之外，

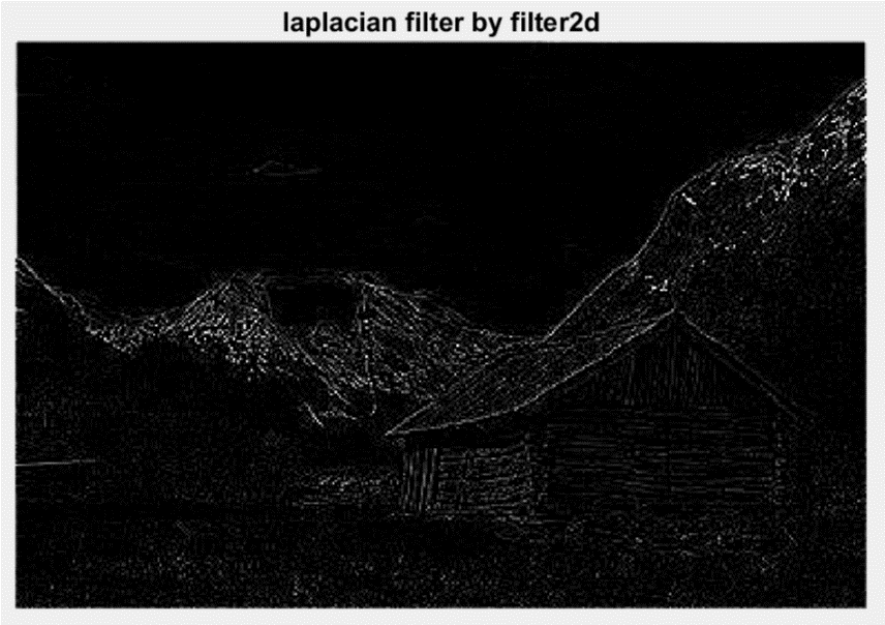
请简单介绍一下为什么拉普拉斯滤波器可以用于图像的锐化。

Run Laplacian_filter.m

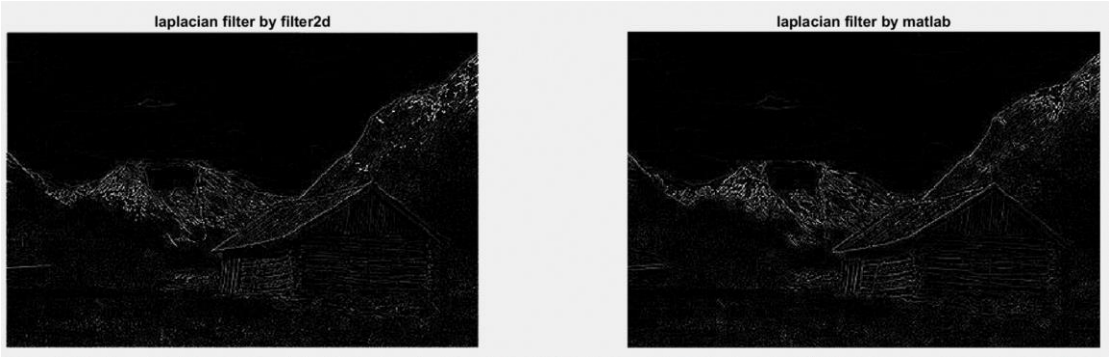
选用拉普拉斯滤波器：

0	−1	0
−1	4	−1
0	−1	0

输出结果：



和 matlab 的 imfilter 函数结果对比：



为什么拉普拉斯滤波器可以用于图像的锐化:

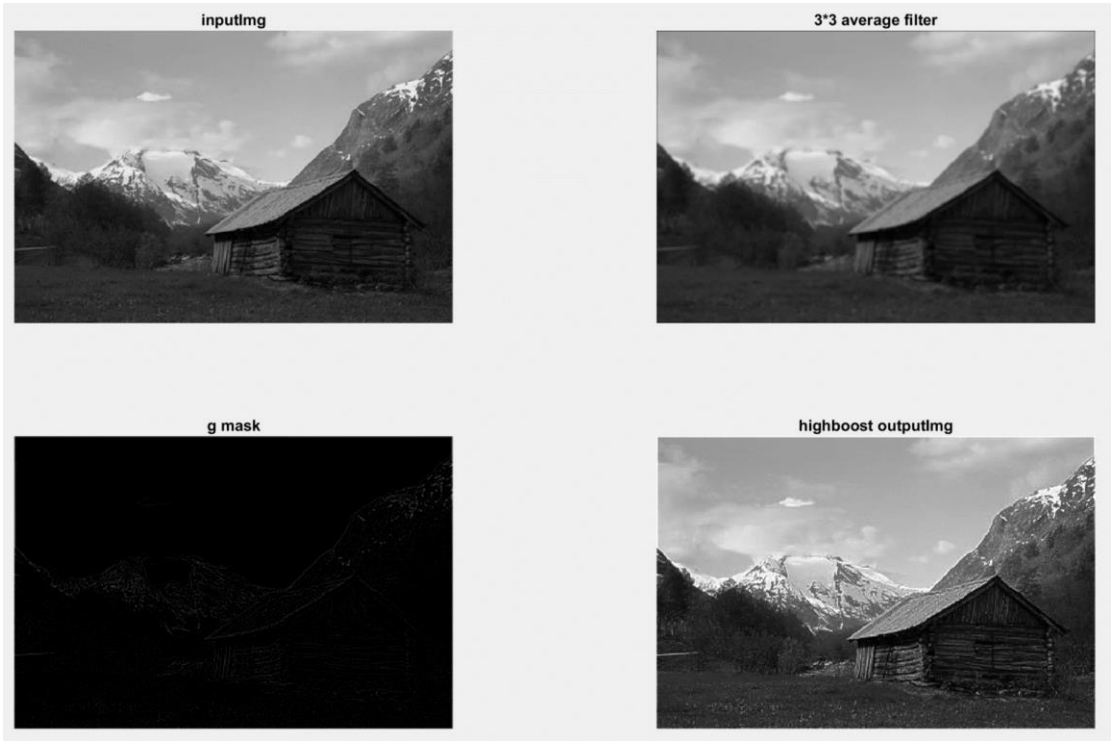
数字图像中的边缘在灰度上常常类似于斜坡过度，这样就导致一阶微分产生较粗的边缘，因为沿着斜坡的微分为零，另一方面，二阶微分产生由零分开的一个像素宽的双边缘，在增强细节方面比一阶微分要好，适合锐化图像。

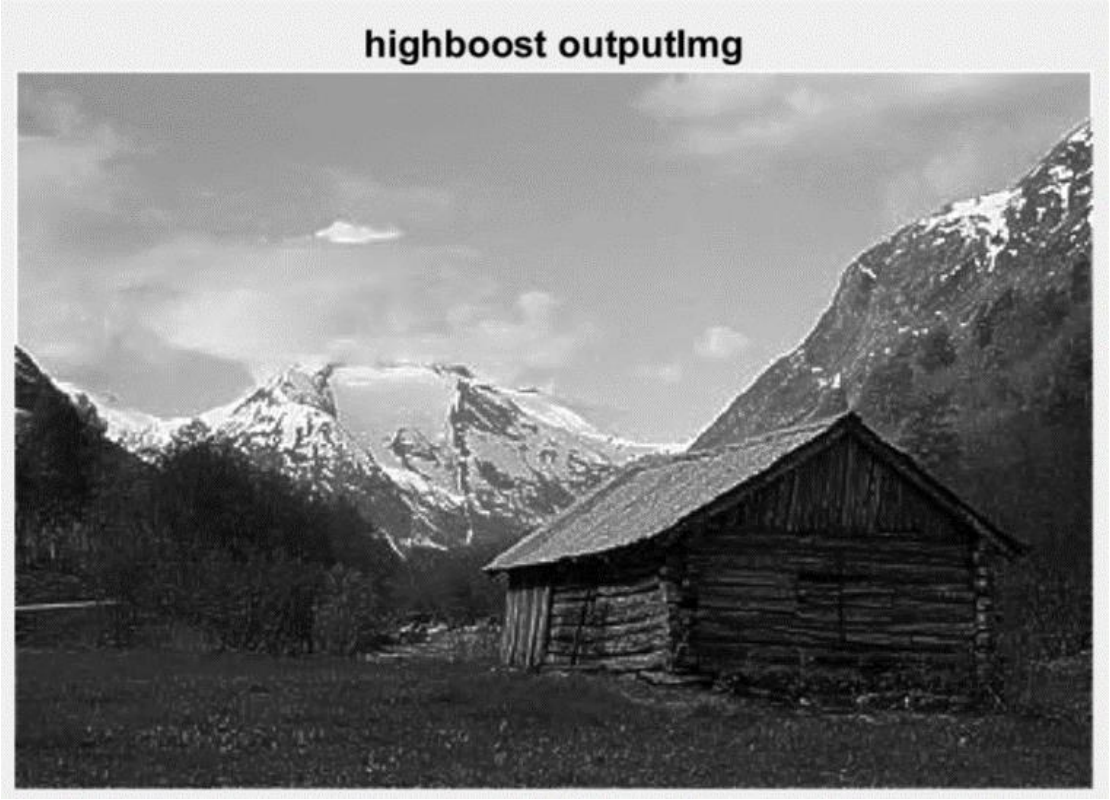
拉普拉斯算子是最简单的各项同性微分算子，其应用强调图像中灰度的突变，并不强调灰度级缓慢变化的区域，这将产生把浅灰色边线和突变点叠加到暗色背景中的图像。将原图像和拉普拉斯图像叠加的一起，可以复原背景特性并保持拉普拉斯锐化处理的效果（如果所使用的滤波器具有负的中心系数，则必须将原图减去经拉普拉斯变换后的图像得到锐化结果）。

3. （5 分）将高提升滤波（high-boost filter）用在输入的图像中（也就是说，$g(x,y) = f(x,y) + k * gmax(x,y)$，其他细节参见课本式(3.6-9)）。过程中涉及的平滑部分应用用课本图 3.32(a)所示的 滤波器来完成。请自行选择合适的 k（式(3.6-9)中的权值）。在报告中，你需要说明你选择的 k 的值，并贴上对应的输出结果。

Run highboost_filter.m

k=2





4. （10 分）详细描述你是如何实现空间滤波操作的，也就是说，针对“filter2d”函数进行算法说明，字数不能超过两页。

filter2d 是一个可用于均值滤波和拉普拉斯变换的通用滤波器函数。
filtersum 记录输入滤波器的矩阵和，若输入均值滤波器，filtersum != 0，滤波器运算中间值要除以 filtersum，若输入拉普拉斯滤波器，filtersum == 0，设 filtersum 为 1，相当于省略除以系数的步骤。
matlab 中，常使用 imshow()函数来显示图像，而此时的图像矩阵可能经过了某种运算。在 matlab 中，为了保证精度，经过了运算的图像矩阵 I 其数据类型会从 unit8 型变成 double 型。如果直接运行 imshow(I)，我们会发现显示的是一个白色的图像。这是因为 imshow()显示图像时对 double 型是认为在 0~1 范围内，即大于 1 时都是显示为白色，而 imshow 显示 uint8 型时是 0~255 范围。而经过运算的范围在 0-255 之间的 double 型数据就被不正常得显示为白色图像了。
所以在运算过程中将图像矩阵转化为 double 类型，最后输出图像时再转化为 unit8 形式。

波器，其最小尺寸为3×3。一般来说，使用大小为 $m \times n$ 的滤波器对大小为 $M \times N$ 的图像进行线性空间滤波，可由下式表示：

$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$

其中，x 和 y 是可变的，以便 w 中的每个像素可访问 f 中的每个像素。

当然，也可以使用偶数尺寸的滤波器，或使用混合有偶数尺寸和奇数尺寸的滤波器。但是，使用奇数尺寸的滤波器可简化索引，并更为直观，因为滤波器的中心落在整数值上。

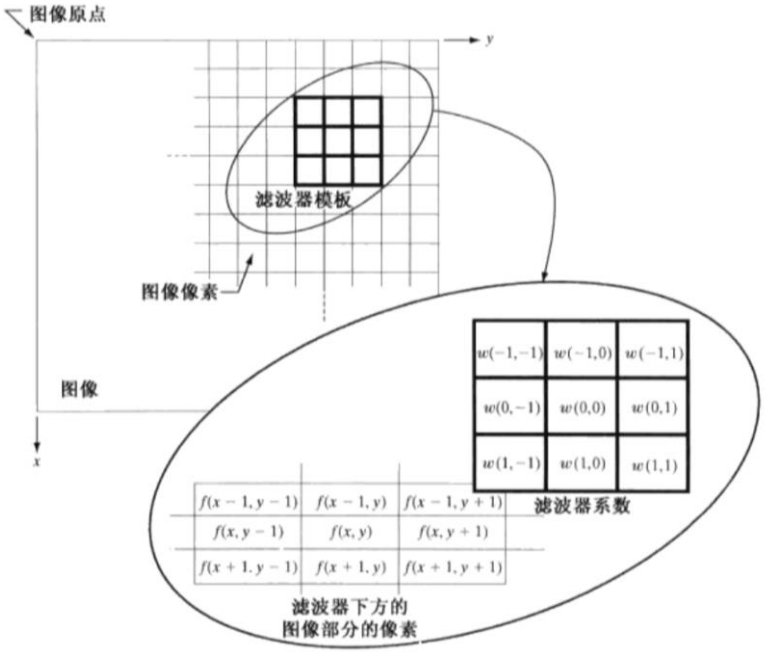


图 3.28　使用大小为 3×3 的滤波器模板的线性空间滤波的机理。表示滤波器模板系数的坐标所选择的形式简化了线性滤波的表达式

考虑：

```
[m,n] = size(input_img);
[a,b] = size(filter);
```

在图像的顶部和底部填充 a-1 行 0，在左侧和右侧填充 b-1 行 0，填充后遍历图中每一个点，按上述线性滤波的公式，把矩阵相乘的结果除以 filternum 的值赋值给中间值。

最后截取掉填充的部分，得到经过滤波器滤波的图像。

检验正确性：

在以上操作中，肉眼可见 filter2d 函数滤波后图像和 matlab 自带 filter2 函数和 imfilter 函数结果一致。

在 filter2d.m 中注释掉此语句：output_img = uint8(output_img);

考虑练习 1.2 中的问题，运行 exercise.m

filter2d.m | exercise.m | average_filter.m | laplacian_filter.m

```
1 —    clc;
2 —    clear;
3
4 —    A = [85 13 20 80; 169 8 243 20; 18 155 163 44; 12 34 50 80];
5 —    filter = [1 1 1; 0 0 0; -1 -1 -1];
6 —    B = filter2(filter, A);
7 —    C = filter2d(A,filter);
8 —    B
9 —    C
```

```
B =

  -177  -420  -271  -263
   -75  -218  -249  -107
   131   324   107   133
   173   336   362   207
```

```
C =

  -177  -420  -271  -263
   -75  -218  -249  -107
   131   324   107   133
   173   336   362   207
```

可以得到运算结果 B，C 值相同。且符合按教材方法笔算得到的结果。