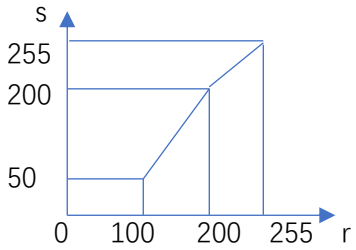
第一次书面作业

1. 若一幅 5*5 图像的图像数据如图(a)所示，灰度变换函数如图(b)所示。

0	10	20	30	40
50	60	70	80	90
100	110	120	130	140
150	160	170	180	190
200	210	220	230	240

(a)图像数据



（b）灰度变换函数

(1) 试简述灰度变换的基本原理。

(2) 试指出图(b)所示的灰度变换函数的数学表达形式及其功能。

(3) 试求图(a)所示的图像经图(b)所示的灰度变换函数进行灰度变换处理后的结果。

解：(1) 灰度变换的基本原理：灰度变换属于空域变换增强技术，是一种点操作，根据原始图像中每个像素的灰度值，按照某种映射规则将其转化为另一灰度值，即将原始图像 f(x,y)中的每个像素的灰度值 按灰度变换函数进行变换，得到目标图像 g(x,y)对应像素的灰度值，其关键是灰度变换函数的设计。

(2) 图(b)所示的灰度变换函数的数学表达形式为：

$$g(x,y)=\begin{cases}50 & 0\le f(x,y)\le 100\\ 1.5\times f(x,y)-100 & 100< f(x,y)<200\\ f(x,y) & 200\le f(x,y)\le 255\end{cases}$$

功能：将原始图像中灰度值小于 100 的像素点的灰度值保持为恒定值，即 50；将灰度值介于 100 至 200 间的像素点的灰度值进行拉伸，即按照函数 g(x,y)=1.5×f(x,y)-100 所指规律进行变换；将灰度值 大于 200 的像素点的灰度值保持不变。

(3) 图(a)所示的图像经图(b)所示的灰度变换函数进行灰度变换处理后的结果为：

50	50	50	50	50
50	50	50	50	50
50	65	80	95	110
125	140	155	170	185
200	210	220	230	240

2. 【非课内上机实习】（采用 Matlab/C++/Python，opencv 均可）

(1) 打开一幅真彩色图像，利用式 Gray(i,j)=0.299*R(i,j)+0.587*G(i,j)+0.144*B(i,j)对其进行灰度化，并显示变换前后图像。

解：

```
Image1=im2double(imread('peppers.jpg'));
R=Image1(:,:,1);
G=Image1(:,:,2);
B=Image1(:,:,3);
Y=0.299*R+0.587*G+0.114*B;
imshow(Y);
```

(2)打开一幅真彩色图像，将绿色和蓝色通道进行互换，显示通道互换后的图像，并对结果进行说明。

解：

```
Image1=imread('peppers.jpg');
%通道互换
Image2=Image1;
Image2(:,:,2)=Image1(:,:,3);
Image2(:,:,3)=Image1(:,:,2);
imshow(Image2);
```

(3)将灰度为256级的图像降低为8级(将图像重新量化)并编程运行显示结果。果。

解：以将灰度为 256 降低为 8（图 a）为例编程并运行程序

```
level1 = 256;
level2 = 8;
ratio = level1/level2;
I1 = imread('cameraman.tif');
subplot(121);imshow(I1);
S = size(I1);
for m = 1: S(1)
     for n = 1: S(2)
         I2(m,n) = uint8(round(double(I1(m,n))/ratio));
         I2(m,n) = uint8(ratio*double(I2(m,n)));
     end
end
subplot(122);imshow(I2);
imwrite(I2,'cameraman_d.tif')
```

程序运行结果如下：

（a） 原始图像（灰度级为 256） （b）灰度级为 8

3. 设有一幅 64×64 的离散图像，其灰度分成 8 层，灰度 n_k 的值和分布情况如表 1 所示。试绘制该图像的直方图，并求经过直方图均衡后的图像的直方图。说明为什么对数字图像进行直方图均衡化后，通常并不能产生完全平坦的直方图。

表 1 一幅图像的灰度分布

K	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
r_k	0	1/7	2/7	3/7	4/7	5/7	6/7	1
n_k	560	920	1046	705	356	267	170	72

解：

K	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
r_k	0	1/7	2/7	3/7	4/7	5/7	6/7	1
n_k	560	920	1046	705	356	267	170	72
$P_r(r_k)$	0.14	0.22	0.26	0.17	0.09	0.07	0.04	0.02
S_k'	0.14	0.36	0.62	0.79	0.88	0.95	0.99	1
S_k''	1/7	3/7	4/7	6/7	6/7	1	1	1
S_k	1/7	3/7	4/7		6/7			1
n_{ks}	560	920	1046		1061			509
$p_r(s_k)$	0.14	0.22	0.26		0.26			0.13

4. 如图 1 所示，设原图像为 10×10 的点阵，求边界点保持不变、经过 3×3 窗口中值滤波的图像。

1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	5	5	5	5	5	5	1	1
1	1	5	5	5	5	5	5	1	1
1	1	5	5	8	8	5	5	1	1
1	1	5	5	8	8	5	5	1	1
1	1	5	5	5	5	5	5	1	1
1	1	5	5	5	5	5	5	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1

图 1　受干扰的二维图像

解：　（1）采用 3×3 窗口在图像上进行扫描，窗口中心值为窗口灰度值排序的中值。

（2）采用中值滤波的程序验证结果：

```
I = [1 1 1 1 1 1 1 1 1 1;
     1 1 1 1 1 1 1 1 1 1;
     1 1 5 5 5 5 5 5 1 1;
     1 1 5 5 5 5 5 5 1 1;
     1 1 5 5 8 8 5 5 1 1;
     1 1 5 5 8 8 5 5 1 1;
     1 1 5 5 5 5 5 5 1 1;
     1 1 5 5 5 5 5 5 1 1;
     1 1 1 1 1 1 1 1 1 1;
     1 1 1 1 1 1 1 1 1 1];
imshow(I);
J = medfilt2(I)
figure, imshow(J);
```

```
J =
     0     1     1     1     1     1     1     1     1     0
     1     1     1     1     1     1     1     1     1     1
     1     1     1     5     5     5     5     1     1     1
     1     1     5     5     5     5     5     5     1     1
     1     1     5     5     5     5     5     5     1     1
     1     1     5     5     5     5     5     5     1     1
     1     1     5     5     5     5     5     5     1     1
     1     1     1     5     5     5     5     1     1     1
     1     1     1     1     1     1     1     1     1     1
     0     1     1     1     1     1     1     1     1     0
```

5. 设图像如图 2 所示，分别求经过邻域平滑和高通算子锐化的结果。其中边缘点保持不变，邻域平滑掩码取 3×3 矩阵，即 $H=\frac{1}{8}\begin{bmatrix}1&1&1\\1&0&1\\1&1&1\end{bmatrix}$；邻域高通算子取 3×3 矩阵，即 $H=\begin{bmatrix}-1&-1&-1\\-1&8&-1\\-1&-1&-1\end{bmatrix}$。

1	1	3	4	5
2	1	4	5	5
2	3	5	4	5
3	2	3	3	2
4	5	4	1	1

图 2　一幅 5×5 的灰度图像矩阵

解：邻域平滑的结果：

1	1	3	4	5
2	21/8	26/8	35/8	5
2	22/8	25/8	32/8	5
3	29/8	27/8	25/8	2
4	5	4	1	1

邻域高通算子滤波的结果：

1	1	3	4	5
2	-13	6	5	5
2	2	15	0	5
3	-13	-4	-1	2
4	5	4	1	1

对于出现的负值区间，可以通过图像增强的方法，即适当的变换映射为可观测的图像灰度范围。

6.已知一幅如图 3 所示的图像。可见原图中左边暗，右边亮，中间存在着一条明显的边界。

0	0	1	255	254	254	254
1	1	1	254	253	254	254
0	0	0	255	255	253	253
1	1	0	254	254	254	254

图 3　有垂直边界的一幅图像

试用 Sobel 算子对给定的图像进行模板操作并分析得到的结果图像。

解：程序如下：

```
I = [0   0   1   255  254  254  254;
     1   1   1   254  253  254  254;
     0   0   0   255  255  253  253;
     1   1   0   254  254  254  254]
J = edge(I,'sobel',0.1);
subplot(121); imshow(I,[0 255]);
subplot(122); imshow(J);
```

运行结果如下：

```
J =
     0     0     0     0     0     0     0
     0     0     1     0     0     0     0
     0     0     1     0     0     0     0
     0     0     0     0     0     0     0
```

可见，有一条一个像素的边缘线。在图像的边界处，由于算子只能确定模板的中心值，未能检测边缘。对于实际图像来说，目标一般在图像的内部，所以无碍于实际应用。