

可视化作业3

18300290007 加兴华

说明：将图片库文件夹放之于路径文件夹下可以保证程序正确运行，smooth_fliter.m, sharpening_fliter.m分别为第一题两种操作的代码。

HW3-1

编程实现图像域基于空间滤波器的（1）平滑操作、（2）锐化算法算法；并把算法应用与图片上，显示与原图的对比差别。

备注：实现的代码不能调用某个算法库里面的函数实现平滑或锐化。

(1)

代码： (matlab)

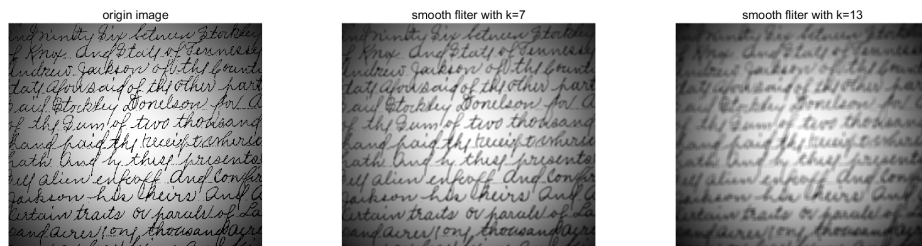
```
1  clc;clear all;close all
2  img=imread('图片库//writting.tif');%读取图像信息
3  subplot(131)
4  imshow(img)
5  title('origin image')
6  b=fill(img,7);
7  b=smoothfliter(b,7);
8  subplot(132)
9  imshow(b)
10 title('smooth fliter with k=7')
11 b=fill(img,13);
12 b=smoothfliter(b,13);
13 subplot(133)
14 imshow(b)
15 title('smooth fliter with k=13')
16
17 % 扩展图像来处理边界
18 function B=fill(A,k) % 采用复制图像边界来扩展
19 p=(k-1)/2; % 滑窗半径
20 [m,n,o]=size(A);
21 % 因为下面生成的扩展矩阵为double类型，为使矩阵乘法合法必须转化
22 A=im2double(A);
23 % 扩展矩阵：temp2为行扩展，temp1为列扩展
24 temp1=blkdiag(ones(1,p+1),eye(n-2),ones(1,p+1));
25 temp2=blkdiag(ones(p+1,1),eye(m-2),ones(p+1,1));
26 % 对所有通道操作赋值
27 for i=1:o
28     B(:, :, i)=temp2*A(:, :, i)*temp1;
29 end
30 end
31
32 % 平滑操作函数
33 function b=smoothfliter(img,k)
34 p=(k-1)/2; % 滑窗半径
35 [m,n,o]=size(img);
36 % 因为下面生成的滤波器矩阵为double类型，为使矩阵乘法合法必须转化
37 img=im2double(img);
```

```

38 % 原尺寸
39 M=m-2*p;
40 N=n-2*p;
41 b=zeros(M,N,o);
42 %生成平滑滤波器
43 temp=ones(k)/9;
44 % 进行卷积核操作
45 for i=1:M
46     for j=1:N
47         for t=1:o
48             b(i,j,t)=sum(sum(temp.*img(i:i+2*p,j:j+2*p,t)));
49         end
50     end
51 end
52 % 归一化
53 for i=1:o
54     maxb=max(max(b(:, :, i)));
55     minb=min(min(b(:, :, i)));
56     b(:, :, i)=(b(:, :, i)-minb)./(maxb-minb);
57 end
58 end

```

运行结果：



分析：可以看出，平滑操作在图片上的体现为变得模糊，并且随着平滑滤波器滑窗的增大，对图片的模糊效果也会增强。

(2)

代码： (matlab)

```

1 clc;clear all;close all
2 img=imread('图片库//moon.tif');%读取图像信息
3 subplot(231)
4 imshow(img)
5 title('origin image')

```

```

6  b=fill(img);
7  b=laplacefliter(b);
8  subplot(232)
9  imshow(b)
10 title('laplace fliter')
11 img=im2double(img);
12 subplot(233)
13 imshow(img-0.1*b)
14 title('sharpened with w=0.1')
15 subplot(234)
16 imshow(img-0.5*b)
17 title('sharpened with w=0.5')
18 subplot(235)
19 imshow(img-0.8*b)
20 title('sharpened with w=0.8')
21 subplot(236)
22 imshow(img-1*b)
23 title('sharpened with w=1')
24
25 % 扩展图像来处理边界
26 function B=fill(A) % 采用复制图像边界来扩展
27 k=3; % 锐化滑窗为3
28 p=(k-1)/2; % 滑窗半径
29 [m,n,o]=size(A);
30 % 因为下面生成的扩展矩阵为double类型，为使矩阵乘法合法必须转化
31 A=im2double(A);
32 % 扩展矩阵: temp2为行扩展, temp1为列扩展
33 temp1=blkdiag(ones(1,p+1),eye(n-2),ones(1,p+1));
34 temp2=blkdiag(ones(p+1,1),eye(m-2),ones(p+1,1));
35 % 对所有通道操作赋值
36 for i=1:o
37     B(:, :, i)=temp2*A(:, :, i)*temp1;
38 end
39 end
40
41 % 获得拉普拉斯滤波
42 function b=laplacefliter(img)
43 k=3; % 锐化滑窗为3
44 p=(k-1)/2; % 滑窗半径
45 [m,n,o]=size(img);
46 % 因为下面生成的滤波器矩阵为double类型，为使矩阵乘法合法必须转化
47 img=im2double(img);
48 % 原尺寸
49 M=m-2*p;
50 N=n-2*p;
51 b=zeros(M,N,o);
52 % 生成图像二阶导数滤波器
53 temp=zeros(k);
54 temp(p+1,1:k)=ones(1,k);
55 temp(1:k,p+1)=ones(k,1);
56 temp(p+1,p+1)=2-2*k;
57 % 进行卷积核操作
58 for i=1:M
59     for j=1:N
60         for t=1:o
61             b(i,j,t)=sum(sum(temp.*img(i:i+2*p,j:j+2*p,t)));
62         end
63     end

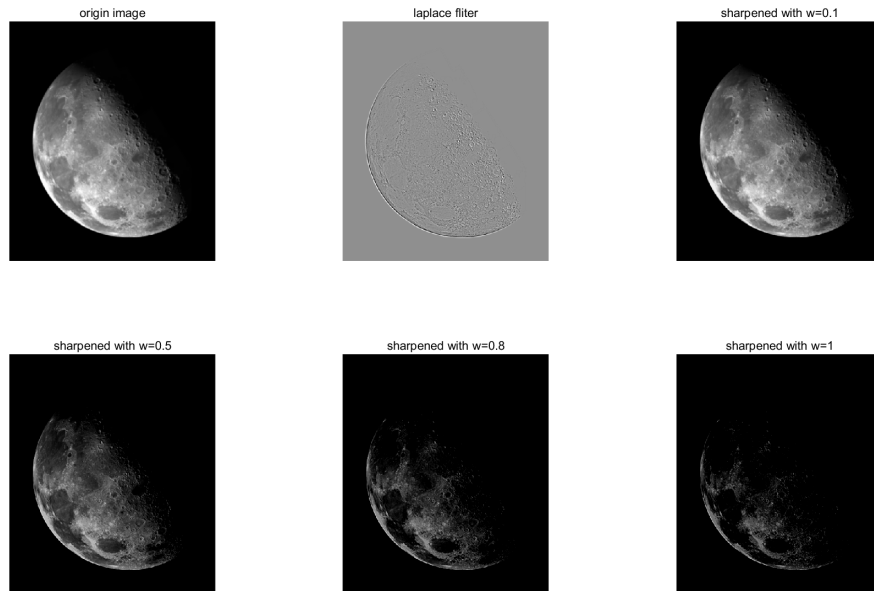
```

```

64 end
65 % 归一化
66 for i=1:o
67     maxb=max(max(b(:,:,i)));
68     minb=min(min(b(:,:,i)));
69     b(:,:,i)=(b(:,:,i)-minb)./(maxb-minb);
70 end
71 end

```

运行结果：



分析：可以看出，原图通过减去合适的拉普拉斯滤波会使得轮廓更加鲜明，但是随着拉普拉斯滤波权重的增大，原图却也只剩下轮廓的细节，所以权重并非越大越好，在我使用的图中，我认为 $w=0.1$ 到 0.5 之间是最合适的。

HW3-2

证明：

- (1) 证明冲击串 (impulse train) 的傅里叶变换后的频域表达式也是一个冲击串。
- (2) 证明实信号 $f(x)$ 的离散频域变换结果是共轭对称的。
- (3) 证明二维变量的离散傅里叶变换的卷积定理。

(1)

1.

$$S_{\Delta T}(t) = \sum_{-\infty}^{+\infty} c_n e^{\frac{j2\pi}{\Delta T} nt} \quad (\because S_{\Delta T}(t) \text{ 是周期为 } \Delta T \text{ 的函数, } \therefore \text{ 可以展开成 } \omega = \frac{2\pi}{\Delta T} \text{ 的傅里叶级数})$$

$$\text{式中 } c_n = \frac{1}{\Delta T} \int_{-\frac{\Delta T}{2}}^{\frac{\Delta T}{2}} S_{\Delta T}(t) e^{-\frac{j2\pi}{\Delta T} nt} dt = \frac{1}{\Delta T}$$

$$\therefore F(S_{\Delta T}(t)) = \frac{1}{\Delta T} F\left(\sum_{-\infty}^{+\infty} e^{\frac{j2\pi}{\Delta T} nt}\right) = \frac{1}{\Delta T} \sum_{-\infty}^{+\infty} F\left(e^{\frac{j2\pi}{\Delta T} nt}\right) \quad (\because \text{ 傅里叶变换的线性性})$$

$$= \frac{1}{\Delta T} \sum_{-\infty}^{+\infty} \delta\left(u - \frac{n}{\Delta T}\right) \quad (\because F\left(e^{\frac{j2\pi}{\Delta T} nt}\right) = \int_{-\infty}^{+\infty} e^{-j2\pi\left(u - \frac{n}{\Delta T}\right)t} dt = \infty \cdot 1_{\left(u = \frac{n}{\Delta T}\right)} = \delta\left(u - \frac{n}{\Delta T}\right))$$

(2)

2.

$$F_m = \frac{1}{M} \sum_{n=0}^{M-1} f_n e^{-\frac{j2\pi mn}{M}}, \quad m = 0: M-1$$

$$F_{-m}^* = \frac{1}{M} \sum_{n=0}^{M-1} f_n e^{\frac{j2\pi(-m)n}{M}}, \quad m = 0: M-1$$

$$\therefore F_{-m}^* = F_m, \text{ 得证}$$

(3)

3.

$$\text{二维离散傅里叶变换: } F(u, v) = F(f) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) e^{-\frac{j2\pi mu}{M}} e^{-\frac{j2\pi nv}{N}}$$

一方面:

$$\begin{aligned} F(f * h) &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) * h(m, n) e^{-\frac{j2\pi mu}{M}} e^{-\frac{j2\pi nv}{N}} \\ &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \left\{ \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \cdot h(m-x, n-y) \right\} e^{-\frac{j2\pi mu}{M}} e^{-\frac{j2\pi nv}{N}} \\ &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \cdot h(m-x, n-y) e^{-\frac{j2\pi mu}{M}} e^{-\frac{j2\pi nv}{N}} \\ &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left[f(x, y) e^{-\frac{j2\pi xu}{M}} e^{-\frac{j2\pi yv}{N}} \right] \cdot \left[h(m-x, n-y) e^{-\frac{j2\pi(m-x)u}{M}} e^{-\frac{j2\pi(n-y)v}{N}} \right] \\ &= \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left[f(x, y) e^{-\frac{j2\pi xu}{M}} e^{-\frac{j2\pi yv}{N}} \right] \cdot \left\{ \sum_{m=-x}^{M-1-x} \sum_{n=-y}^{N-1-y} h(m, n) e^{-\frac{j2\pi mu}{M}} e^{-\frac{j2\pi nv}{N}} \right\} \\ &= F(h) \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left[f(x, y) e^{-\frac{j2\pi xu}{M}} e^{-\frac{j2\pi yv}{N}} \right] \quad (\because \text{ 时移不变性}) = F(h) \cdot F(f) \end{aligned}$$

另一方面:

$$\begin{aligned} F^{-1}(F(h) * F(f)) &= \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) * H(u, v) e^{\frac{j2\pi mu}{M}} e^{\frac{j2\pi nv}{N}} \\ &= \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} \left\{ \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} F(x, y) \cdot H(u-x, v-y) \right\} e^{\frac{j2\pi mu}{M}} e^{\frac{j2\pi nv}{N}} \\ &= \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left[F(x, y) e^{\frac{j2\pi mx}{M}} e^{\frac{j2\pi ny}{N}} \right] \cdot \left\{ \sum_{u=-x}^{M-1-x} \sum_{v=-y}^{N-1-y} H(m, n) e^{\frac{j2\pi mu}{M}} e^{\frac{j2\pi nv}{N}} \right\} \quad (\text{与上述同理可得}) \\ &= MN \cdot f(m, n) \cdot h(m, n) \end{aligned}$$

