



# 数字图像处理

## 第五次作业

制作成员	班级	学号
赵国梁	自动化 63 班	2160504081

摘要：频域滤波是一类常用的数字图像处理方法，包括低通滤波和高通滤波两个部分，它与空域滤波相对应，具有全局性，更好地体现图像的整体特性。本文从频域滤波出发，利用 Matlab 编程实现两种低通滤波和四种高通滤波。在低通滤波方面，本文设计了布特沃斯低通滤波器和高斯滤波器，并分析参数的变化给滤波效果带来的影响；在高通滤波方面，本文设计了布特沃斯高通滤波器和高斯滤波器，以及拉普拉斯和 unmark 频域滤波。最后，结合上次作业研究空域滤波，分析了空域滤波和频域滤波的关系，并指出两者在滤波上的区别。

一. 频域低通滤波器: 设计低通滤波器包括 **butterworth and Gaussian** (选择合适的半径, 计算功率谱比), 平滑测试图像 **test1** 和 **2**; 分析各自优缺点;

1. 问题分析:

(1) 功率谱

功率谱定义为

$$P(u, v) = F(u, v) = R(u, v) + I(u, v)$$

$R$  和  $I$  分别是  $F(u, v)$  的实部和虚部, 并且所有的计算直接对离散变量  $u=0, 1, \dots, P-1$ ,  $v=0, 1, \dots, Q-1$ 。

(2) 频域滤波的步骤:

① : 先将图像做频域内的水平移动, 然后求原图像  $f(x, y)$  的 DFT, 得到其图像的傅里叶谱  $F(u, v)$ 。

$$F(u, v) = \mathcal{F}[(-1)^{x+y} f_p(x, y)]$$

② 与频域滤波器做乘积。  $G(u, v) = H(u, v)F(u, v)$

③ 求取  $G(u, v)$  的 IDFT, 然后再将图像做频域内的水平移动 (移动回去), 其结果可能存在寄生的虚数, 此时忽略即可。

$$g_p(x, y) = \{\text{Re}(\text{IDFT}(G(u, v)))\}(-1)^{x+y}$$

④ 这里使用 `ifft2` 函数进行 IDFT 变换, 得到的图像的尺寸为  $P \times Q$ 。切取左上角的  $M \times N$  的图像, 即可得到结果。

(3) 巴特沃斯低通滤波器:

截止频率位于距原点  $D_0$  处的  $n$  阶布特沃斯低通滤波器 (BLPF) 的传递函数定义为:

$$H(u, v) = \frac{1}{1 + (D(u, v)/D_0)^{2n}}$$

其中,  $D(u, v)$  的计算方式也就是两点间的距离, 即

$$D(u, v) = \sqrt{(u - \frac{p}{2})^2 + (v - \frac{q}{2})^2}$$

BLPF 传递函数并没有在通过频率和滤除频率之间给出明显截止的尖锐的不连续性。对于具有平滑传递函数的滤波器, 可在这样一点定义截止频率, 即使得  $H(u, v)$  下降到其最大值的某个百分比点。对于上式, 截止频率点是当  $D(u, v) = D_0$  时的点, 即  $H(u, v)$  从其最大值 1 降为 50%。

(4) 高斯低通滤波器:

$$H(u, v) = e^{\frac{-D^2(u, v)}{2D_0^2}}$$

其中,  $D_0$  是截止频率。当  $D(u, v) = D_0$  时, GLPF 下降到其最大值的 0.607 处。

## 2. 实验结果:

### (1) butterworth 低通滤波器:

test1.bmp D0=25 功率谱比  $a = 0.9741 = 97.41\%$

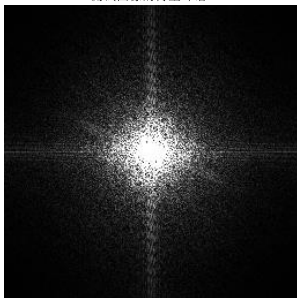
原始图像



butterworth低通滤波后



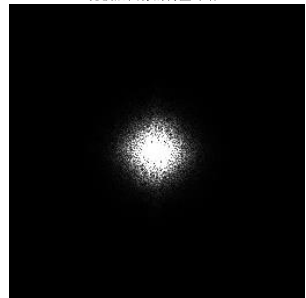
测试图像的傅里叶谱



显示为图像的滤波器 (D0=25)



滤波后图像的傅里叶谱



test1.bmp D0=50 功率谱比  $a = 0.9909 = 99.09\%$

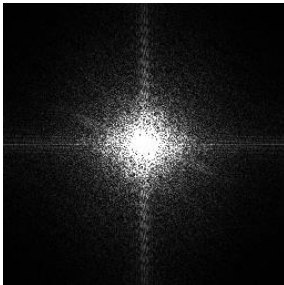
原始图像



butterworth低通滤波后



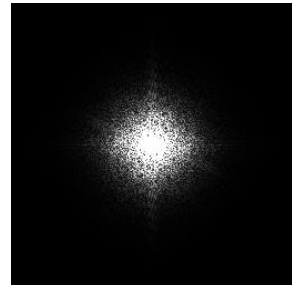
测试图像的傅里叶谱



显示为图像的滤波器 (D0=50)



滤波后图像的傅里叶谱



test1.bmp D0=75 功率谱比 a= 0.9957=99.57%

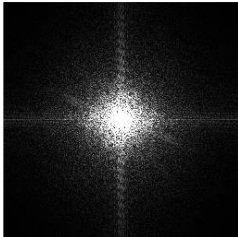
原始图像



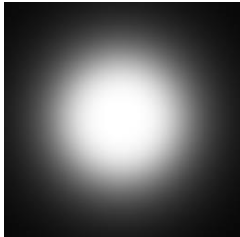
butterworth低通滤波后



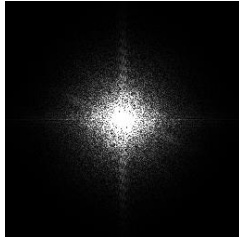
测试图像的傅里叶谱



显示为图像的滤波器 (D0=75)



滤波后图像的傅里叶谱



test2.tif D0=50 功率谱比 a= 0.9885=98.85%

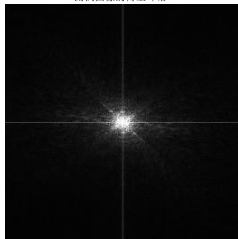
原始图像



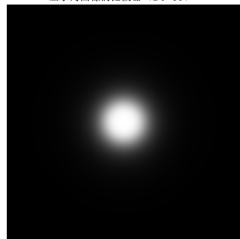
butterworth低通滤波后



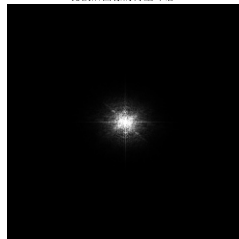
测试图像的傅里叶谱



显示为图像的滤波器 (D0=50)



滤波后图像的傅里叶谱



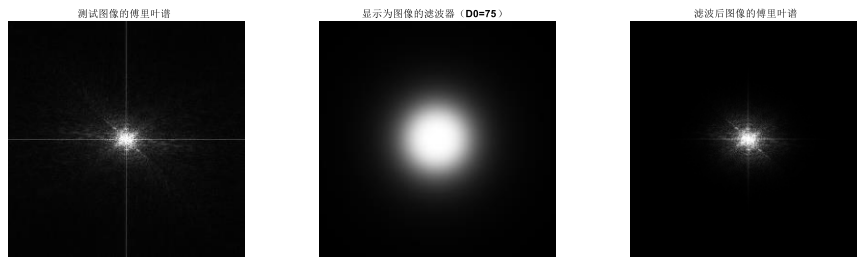
test2.tif D0=75 功率谱比 a= 0.9916=99.16%

原始图像

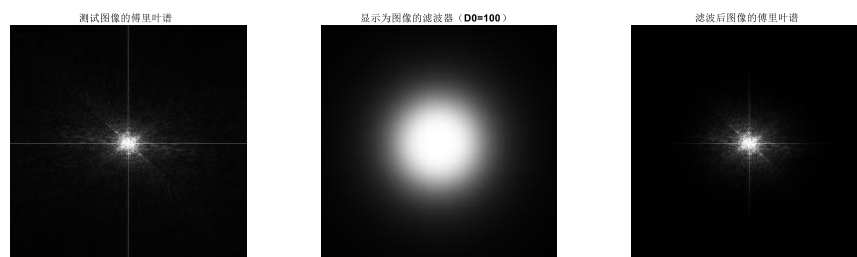
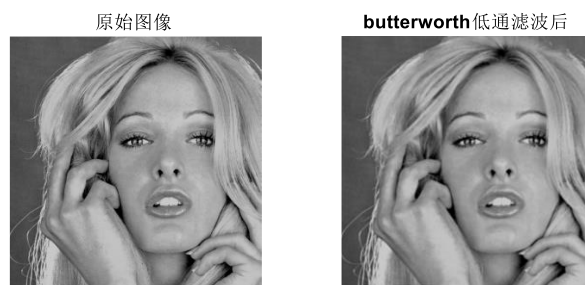


butterworth低通滤波后



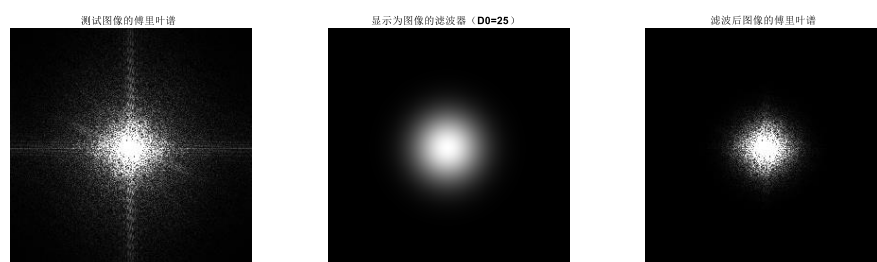
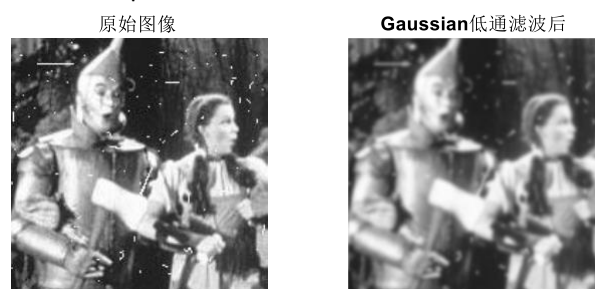


test2.tif D0=100 功率谱比  $a=0.9934=99.34\%$



(2) Gaussian 低通滤波器:

test1.bmp D0=25 功率谱比  $a=0.9657=96.57\%$



---

test1.bmp D0=50 功率谱比  $a=0.9862=98.62\%$

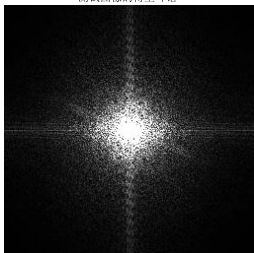
原始图像



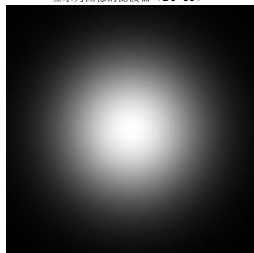
Gaussian低通滤波后



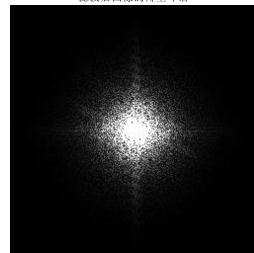
测试图像的傅里叶谱



显示为图像的滤波器 (D0=50)



滤波后图像的傅里叶谱



test1.bmp D0=75 功率谱比  $a=0.9925=99.25\%$

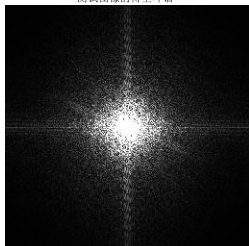
原始图像



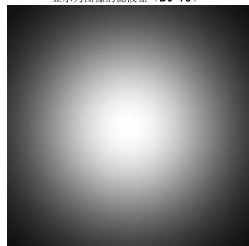
Gaussian低通滤波后



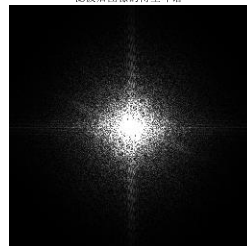
测试图像的傅里叶谱



显示为图像的滤波器 (D0=75)



滤波后图像的傅里叶谱

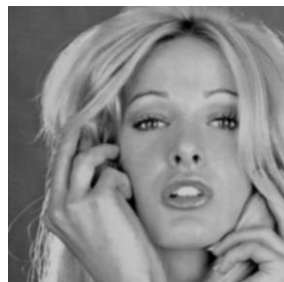


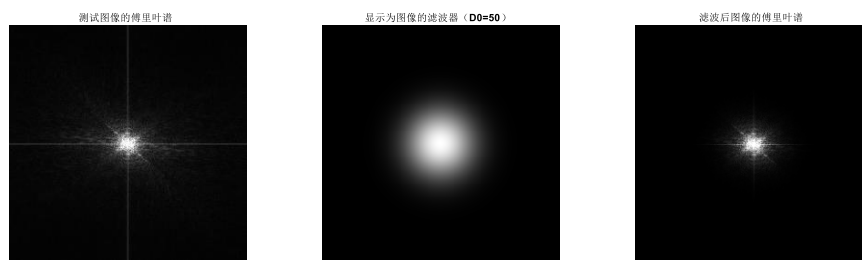
test2.tif D0=50 功率谱比  $a=0.9863=98.63\%$

原始图像

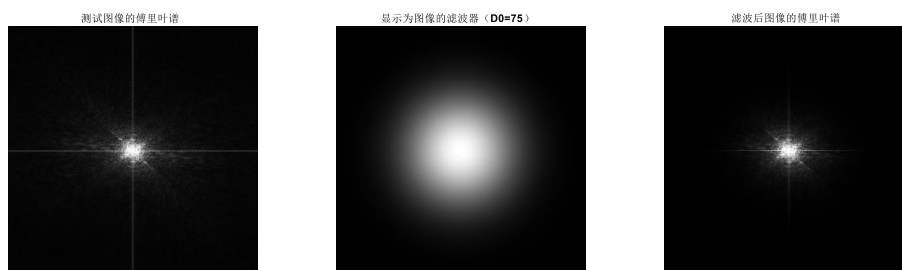
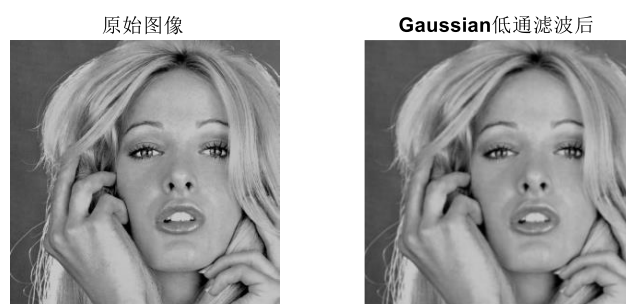


Gaussian低通滤波后

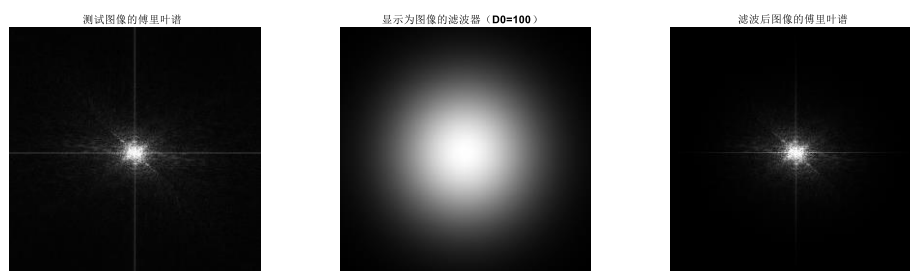
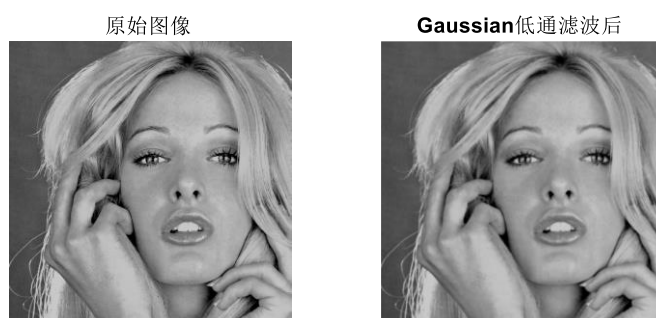




test2.tif D0=75 功率谱比  $a = 0.9902 = 99.02\%$



test2.tif D0=100 功率谱比  $a = 0.9924 = 99.24\%$



### 3. 结果分析:

对比每组图像处理结果中的原始图像和低通滤波后的图像,可以清晰看到低通滤波器的平滑效果。

以 test2 为例,分析二阶布特沃斯低通滤波器选取  $D_0$  对滤波效果的影响。分别选取  $D_0=20$ 、40、60 的二阶布特沃斯低通滤波器进行滤波。由下图结果可知,随着截止频率  $D_0$  的减小,滤波后的图像越来越模糊,功率谱比越来越小,即滤波后包含的低频分量越来越少。



以 test2 为例,分析高斯低通滤波器选取  $D_0$  对滤波效果的影响。分别选取  $D_0=20$ 、40、60 的高斯低通滤波器进行低通滤波。由下图结果可知,随着截止频率  $D_0$  的减小,滤波后的图像越来越模糊,功率谱比越来越小,即滤波后包含的低频分量越来越少。



最后,对比二阶布特沃斯低通滤波器和高斯低通滤波器的效果知,两种滤波器达到的基本效果是一致的,即平滑图像,滤除高频分量,保留低频分量。但两者在相同截止频率  $D_0$  时,得到的功率谱比却不同,主要原因是两个滤波器在过渡带处的差异。

## 二. 频域高通滤波器: 设计高通滤波器包括 **butterworth and Gaussian**, 在频域增强边缘。选择半径和计算功率谱比, 测试图像 test3,4: 分析各自优缺点;

### 1. 问题分析:

#### (1) 巴特沃斯高通滤波器:

截止频率位于距原点  $D_0$  处的  $n$  阶布特沃斯低通滤波器 (BLPF) 的传递函数定义为

$$H(u,v) = \frac{1}{1 + (D_0 / D(u,v))^{2n}}$$

巴特沃斯高通滤波器可以通过改变次数  $n$ , 对过度特性进行调整, 但是如果  $n$  值过大, 会造成振铃现象。

#### (2) 高斯高通滤波器:

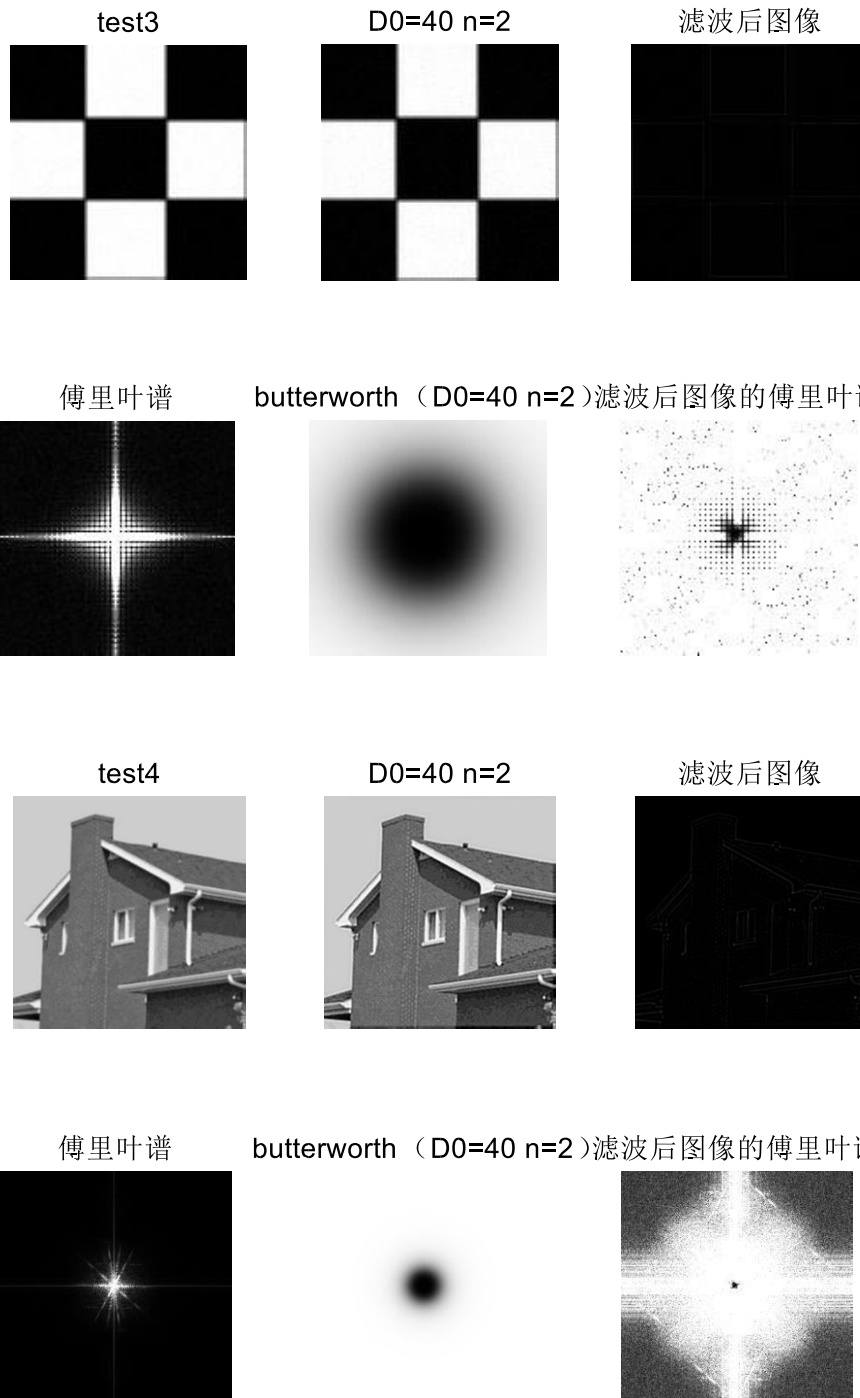


$$H(u,v) = 1 - e^{\frac{-D^2(u,v)}{2D_0^2}}$$

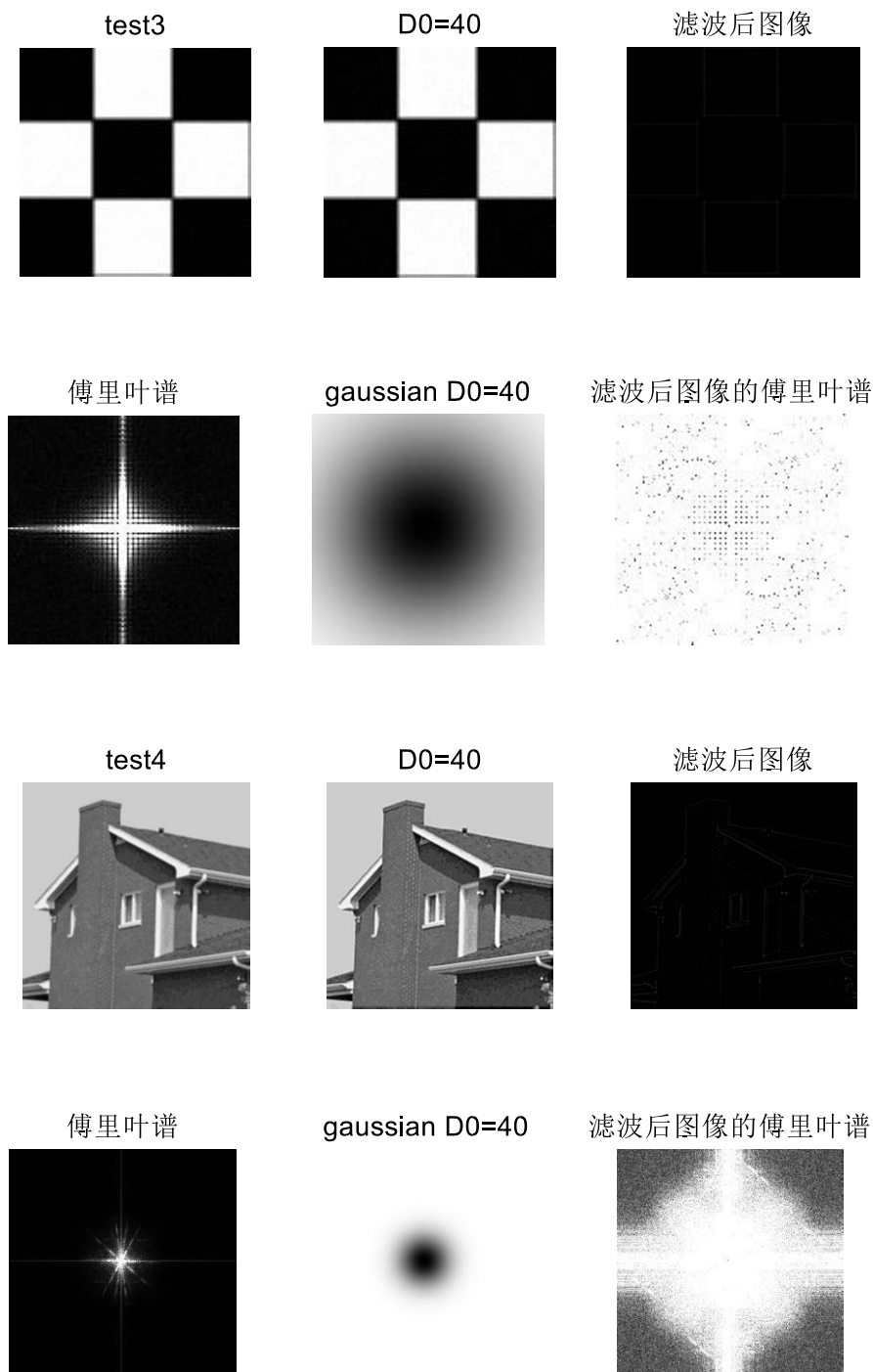
高斯高通滤波器的过度特性很好，所以不会发生振铃现象。

## 2. 实验结果：

Butterworth 高通滤波器，选择  $D_0=40$ ,  $n=2$ 。test3 的功率谱比为  $5.6939e-04$ ，test4 的功率谱比为  $0.0029$ 。



Gaussian 高通滤波器，选择  $D_0=40$ 。test3 的功率谱比为  $5.3888e-04$ ，test4 的功率谱比为 0.0024。



### 3. 结果分析;

对比每组图像处理结果中的原始图像和高通滤波后的图像，可以清晰看到高通滤波器的边缘增强效果；

以 test4 为例，分析二阶布特沃斯高通滤波器选取  $D_0$  对滤波效果的影响。分别选取  $D_0=20$ 、40、60 的二阶布特沃斯高通滤波器进行滤波。

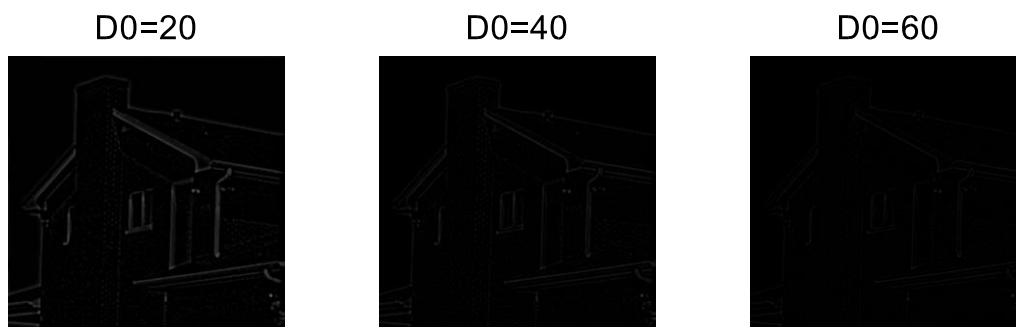
由下图结果可知，随着截止频率  $D_0$  的增加，滤波后的图像边缘应该越来越清晰，功率谱比越来越小，即滤波后包含的高频分量越来越少。但当  $D_0$  增大到一定程度时，

边缘将不见，主要是因为滤除的能量过多，导致图像仅有的可视部分全部变成了黑色。



以 test4 为例，分析高斯高通滤波器选取 D0 对滤波效果的影响。分别选取 D0=20、40、60 的高斯高通滤波器进行滤波。

由下图结果可知，随着截止频率 D0 的增加，滤波后的图像边缘应该越来越清晰，功率谱比越来越小，即滤波后包含的高频分量越来越少。但当 D0 增大到一定程度时，边缘将不见，主要是因为滤除的能量过多，图像全部变成了黑色。



对比二阶 butterworth 低通滤波器和 Gaussian 低通滤波器的效果知，两种滤波器达到的基本效果是一致的。即平滑图像，滤除高频分量，保留低频分量。

但两者在相同截止频率 D0 时，可以清晰的看出高斯高通滤波器比布特沃斯滤波器留下的分量更多，边缘更清晰明亮，功率谱比布特沃斯高通滤波器的大，主要原因是两个滤波器在过渡带处的差异。

对比高通滤波器和低通滤波器知，高通滤波器在滤波的时候会将直流分量也一同滤除，导致图像变暗。造成当 D0 增大到一定程度时，边缘将不见，整个图像变为黑色。

### 三. 其他高通滤波器：拉普拉斯和 Unmask，对测试图像 test3,4 滤波；分析各自优缺点；

#### 1. 问题分析：

##### (1) 频率域的拉普拉斯算子：

拉普拉斯算子可使用如下滤波器在频率域实现：

$$H(u, v) = -4\pi^2[(u - P/2)^2 + (v - Q/2)^2] = -4\pi^2 D^2(u, v)$$

拉普拉斯图像由下式得到：

$$\nabla^2 f(x, y) = \mathfrak{F}^{-1}\{H(u, v)F(u, v)\}$$

增强可使用下式实现：

$$g(x, y) = f(x, y) + c \nabla^2 f(x, y)$$

在频率域：

$$g(x, y) = \mathfrak{F}^{-1}\{F(u, v) - H(u, v)F(u, v)\} = \mathfrak{F}^{-1}\{[1 - H(u, v)]F(u, v)\} = \mathfrak{F}^{-1}\{[1 - 4\pi^2 D^2(u, v)]F(u, v)\}$$

(2) 钝化模板 (unmask) :

钝化模板由下式给出：

$$g_{mask}(x, y) = f(x, y) - f_{LP}(x, y)$$

$$f_{LP}(x, y) = \mathfrak{F}^{-1}[H_{LP}(u, v)F(u, v)]$$

最后的图像由下式给出：

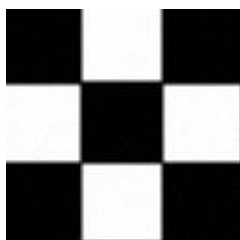
$$g(x, y) = f(x, y) + k * g_{mask}(x, y)$$

最后的频域滤波公式：

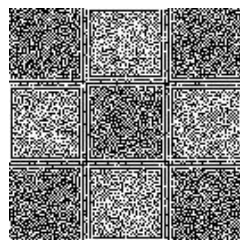
2. 实验结果：

laplace 高通滤波器，test3 的功率谱比为 1.6861e+07，test4 的功率谱比为 8.4623e+08。

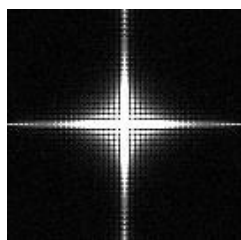
原图



结果图



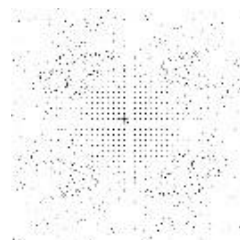
傅里叶谱



laplace



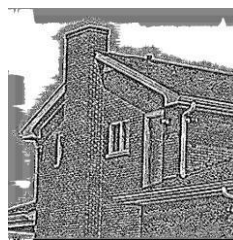
滤波后图像的傅里叶谱



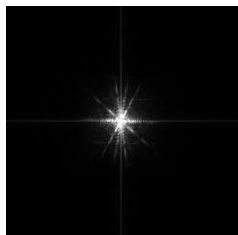
原图



结果图



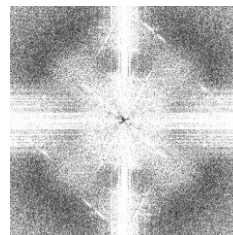
傅里叶谱



laplace

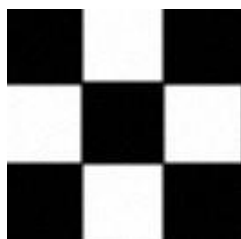


滤波后图像的傅里叶谱

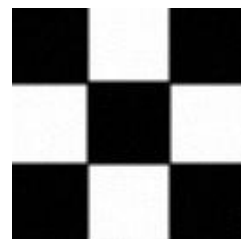


unmask 高通滤波器, test3 的功率谱比为 1.0042, test4 的功率谱比为 1.0083。

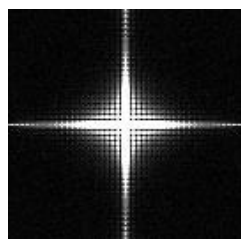
原图



结果图



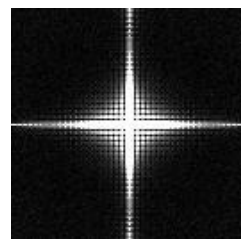
傅里叶谱



unmask



滤波后图像的傅里叶谱



原图



结果图





### 3. 结果分析:

对比每组图像处理结果中的原始图像和滤波后的图像,大致看到滤波器的边缘增强效果, test4 的效果较为明显。拉普拉斯滤波之后两个矩阵相加会超过 0-255 的范围,故在编程实现时需要进行归一化处理后。对比拉普拉斯算子和 unmask 滤波,两者的滤波后图像没有太大的区别。

## 四. 比较并讨论空域低通高通滤波 (Project3) 与频域低通和高通的关系。

空间域和频域滤波间的纽带是卷积定理。空间域中的滤波定义为滤波函数  $h(x, y)$  与输入图像  $f(x, y)$  进行卷积; 频率域中的滤波定义为滤波函数  $H(u, v)$  与输入图像的傅里叶变换  $F(u, v)$  进行相乘。空间域的滤波器和频率域的滤波器互为傅里叶变换。

从空域的角度看,平滑滤波是要滤除不规则的噪声或干扰的影响;从频域的角度看,不规则的噪声具有较高的频率,所以可用具有低通能力的频域滤波器来滤除。故空域的平滑滤波对应频域的低通滤波,频域里低通滤波器的转移函数应该对应空域里平滑滤波器的模板函数的傅里叶变换。从空域的角度看,锐化滤波是要增强边缘和轮廓处的强度;从频域的角度看,边缘和轮廓处都具有较高的频率,所以可用具有高通能力的频域滤波器来增强。故空域的锐化滤波对应频域的高通滤波,频域里高通滤波器的转移函数应该对应空域里锐化滤波器的模板函数的傅里叶变换。即空域和频域的滤波器组成傅里叶变换对。空域的锐化滤波或频域的高通滤波可用两个空域的平滑滤波器或两个频域的低通滤波器实现。

在频域中分析图像的频率成分与图像的视觉效果间的对应关系比较直观。空域滤波在具体实现上和硬件设计上有一定优点。空域中每次都只是一部分像素的性质,而频域技术每次都利用图像中所有像素的数据,具有全局性,有可能更好地体现图像的整体特性,如整体对比度和平均灰度值等。

频域增强技术与空域增强技术也有密切的联系。一方面,许多空域增强技术可借助频域概念来分析和帮助设计;另一方面,许多空域增强技术可转化到频域实现,而许多频域增强技术可转化到空域实现。

---

## 附录:

### 参考文献

- [1] 冈萨雷斯.数字图像处理（第三版）北京：电子工业出版社，2011
- [2] 周品.MATLAB 数字图像处理 北京：清华大学出版社，2012
- [3] 杨杰.数字图像处理及 MATLAB 实现 北京：电子工业出版社，2010

### 代码

1. 设计低通滤波器包括 butterworth and Gaussian (选择合适的半径, 计算功率谱比), 平滑测试图像 test1 和 2; 分析各自优缺点;

#### Butterworth 低通:

```
close all
clear
clc
I=imread('test2.tif');
figure(1);
subplot(1,2,1);
imshow(I);
title('原始图像');
f=double(I);
F=fft2(f);
F=fftshift(F);
[P,Q]=size(F);
n_butterworth=2;
D0=100;
for u=1:P
    for v=1:Q
        D(u,v)=sqrt((u-fix(P/2))^2+(v-fix(Q/2))^2);
        H(u,v)=1/(1+(D(u,v)/D0)^(2*n_butterworth));
        G(u,v)=H(u,v)*F(u,v);
    end
end
g=ifftshift(G);
g=ifft2(g);
g=uint8(real(g));
subplot(1,2,2);
imshow(g);
title('butterworth 低通滤波后');
imwrite(g,'test1 (D0=75) .bmp');
S=0;
S1=0;
for u=1:P
```

---

```

        for v=1:Q
            L1=(abs (G(u,v)))^2;
            S1=S1+L1;
            L=(abs(F(u,v)))^2;
            S=S+L;
        end
    end
    L=S1/S
    %绘制频谱图（显示为图像）
    figure(2)
    subplot(1,3,1);
    imshow(abs(255.*F./max(max(F))));
    title('测试图像的傅里叶谱');
    subplot(1,3,2);
    imshow(abs(H));
    title('显示为图像的滤波器（D0=100）');
    subplot(1,3,3);
    imshow(abs(255.*G./max(max(G))));
    title('滤波后图像的傅里叶谱');
Gaussian 低通：
    close all
    clear
    clc
    I=imread('test2.tif');
    figure(1);
    subplot(1,2,1);
    imshow(I);
    title('原始图像');
    f=double(I);
    F=fft2(f);
    F=fftshift(F);
    [P,Q]=size(F);
    D0=75;
    for u=1:P
        for v=1:Q
            D(u,v)=sqrt((u-fix(P/2))^2+(v-fix(Q/2))^2);
            H(u,v)=exp(-D(u,v)^2/(2*D0^2));
            G(u,v)=F(u,v)*H(u,v);
        end
    end
    g=ifftshift(G);
    g=ifft2(g);
    g=uint8(real(g));
    subplot(1,2,2);

```



---

```

imshow(g);
title('Gaussian 低通滤波后');
S=0;
S1=0;
for i=1:P
    for j=1:Q
        L=(abs(F(i,j)))^2;%计算结果图像的功率谱
        S=S+L;
        L1=(abs(G(i,j)))^2;
        %计算源图像的功率谱
        S1=S1+L1;
    end
end
L=S1/S;%计算功率谱比?%绘制频谱图（显示为图像）
figure(2)
subplot(1,3,1);
imshow(abs(255.*F./max(max(F))));
title('测试图像的傅里叶谱');
subplot(1,3,2);
imshow(abs(H));
title('显示为图像的滤波器（D0=75）');
subplot(1,3,3);
imshow(abs(255.*G./max(max(G))));
title('滤波后图像的傅里叶谱');

```

2. 设计高通滤波器包括 butterworth and Gaussian，在频域增强边缘。选择半径和计算功率谱比，测试图像 test3, 4: 分析各自优缺点；

butterworth 高通：

```

A=imread('test4 copy.bmp');
% A=imread('test3_corrupt.pgm'); f=double(A);
F=fft2(f);
F=fftshift(F);
[x,y]=size(F);
n=2;
D0=40;
for u=1:x
    for v=1:y
        D(u,v)=sqrt((u-fix(x/2))^2+(v-fix(y/2))^2);
        H(u,v)=1/(1+(D0/D(u,v))^(2*n));
        G(u,v)=H(u,v)*F(u,v);
    end
end
g=ifftshift(G); g=ifft2(g);
B=real(g)+f; B=uint8(B);
s=0;

```

---

```

s1=0;
for u=1:x
    for v=1:y
        L=(abs(G(u,v)))^2;
        s=s+L;
        L1=(abs(F(u,v)))^2;
        s1=s1+L1;
    end
end
a=s/s1
subplot(2,3,1);imshow(A);title('test4');
subplot(2,3,2);imshow(B);title('D0=40 n=2');
subplot(2,3,3);imshow(uint8(real(g)));title('滤波后图像');
subplot(2,3,4);imshow(abs(255.*F./max(max(F))));title('傅里叶谱');
subplot(2,3,5);imshow(abs(H));title('butterworth 滤波 D0=40 n=2');
subplot(2,3,6);imshow(abs(255.*G./max(max(G))));title('滤波后图像的傅里叶谱');
Gaussian 高通
% A=imread('test4 copy.bmp');
A=imread('test3_corrupt.pgm');
f=double(A);
F=fft2(f);
F=fftshift(F); [x,y]=size(F); D0=40;
for u=1:x
    for v=1:y
        D(u,v)=sqrt((u-fix(x/2))^2+(v-fix(y/2))^2);
        H(u,v)=1-exp(-D(u,v)^2/(2*D0*D0));
        G(u,v)=H(u,v)*F(u,v);
    end
end
g=ifftshift(G); g=ifft2(g);
B=real(g)+f; B=uint8(B);
s=0;
s1=0;
for u=1:x
    for v=1:y
        L=(abs(G(u,v)))^2;
        s=s+L;
        L1=(abs(F(u,v)))^2;
        s1=s1+L1;
    end
end
a=s/s1
subplot(2,3,1);imshow(A);title('test3');
subplot(2,3,2);imshow(B);title('D0=40');

```

---

```
subplot(2,3,3);imshow(uint8(real(g)));title('滤波后图像');
subplot(2,3,4);imshow(abs(255.*F./max(max(F))));title('傅里叶谱');
subplot(2,3,5);imshow(abs(H));title('gaussian D0=40');
subplot(2,3,6);imshow(abs(255.*G./max(max(G))));title('滤波后图像的傅里谱');
```

**3. 拉普拉斯和 Unmask，对测试图像 test3, 4 滤波；分析各自优缺点；**

laplace

```
A=imread('test4 copy.bmp');
% A=imread('test3_corrupt.pgm');
f=double(A);
F=fft2(f);
F=fftshift(F);
[x,y]=size(F);
for u=1:x
    for v=1:y
        D(u,v)=sqrt((u-fix(x/2))^2+(v-fix(y/2))^2);
        H(u,v)=1+4*pi*pi*D(u,v)^2;
        G(u,v)=H(u,v)*F(u,v);
    end
end
g=ifftshift(G); g=ifft2(g);
B=uint8(real(g));
s=0;
s1=0;
for u=1:x
    for v=1:y
        L=(abs(G(u,v)))^2;
        s=s+L;
        L1=(abs(F(u,v)))^2;
        s1=s1+L1;
    end
end
a=s/s1
subplot(2,3,1);imshow(A);title('原图');
subplot(2,3,3);imshow(B);title('结果图');
subplot(2,3,4);imshow(abs(255.*F./max(max(F))));title('傅里叶谱');
subplot(2,3,5);imshow(abs(255.*H./max(max(H))));title('laplace');
subplot(2,3,6);imshow(abs(255.*G./max(max(G))));title('滤波后图像的傅里谱');
```

unmask

```
A=imread('test4 copy.bmp');
% A=imread('test3_corrupt.pgm');
f=double(A);
F=fft2(f);
F=fftshift(F);
[x,y]=size(F);
```

---

```
D0=60;
for u=1:x
    for v=1:y
        D(u,v)=sqrt((u-fix(x/2))^2+(v-fix(y/2))^2);
        H(u,v)=1-exp(-D(u,v)^2/(2*D0^2));
        G(u,v)=F(u,v)+H(u,v)*F(u,v);
    end
end
g=ifftshift(G); g=ifft2(g);
B=uint8(real(g));
s=0;
s1=0;
for u=1:x
    for v=1:y
        L=(abs(G(u,v)))^2;
        s=s+L;
        L1=(abs(F(u,v)))^2;
        s1=s1+L1;
    end
end
a=s/s1
subplot(2,3,1);imshow(A);title('原图');
subplot(2,3,3);imshow(B);title('结果图');
subplot(2,3,4);imshow(abs(255.*F./max(max(F))));title('傅里叶谱');
subplot(2,3,5);imshow(abs(255.*H./max(max(H))));title('unmask');
subplot(2,3,6);imshow(abs(255.*G./max(max(G))));title('滤波后图像的傅里叶谱');
```