

数字图像处理第五次作业

姓名：边策

班级：自动化 63

学号：2160504062

摘要：本次实验进行频域上的高通低通滤波。MATLAB 频域上的滤波由以下步骤实现

1. 给定一幅大小为 $M \times N$ 的输入图像 $f(x, y)$ ，将图像化为浮点型（最好是 single 型，可以使 FFT 速度变快）。典型地，我们取 $P=2M$ ， $Q=2N$ 。

2. 在 f 图像右下方向填充必要的 0，使其大小变为 $P \times Q$ ，得到图像 $f_p(x, y)$ 。通常用函数 $F = \text{fft2}(f, P, Q)$ ；

3. $F = \text{ifftshift}(F)$ 使图像移到变换中心。

4. 生成一个大小为 $P \times Q$ 的滤波函数 $H(u, v)$ ，中心在 $(P/2, Q/2)$ 处。

5. 形成乘积 $G(u, v) = H(u, v) .* F(u, v)$

6. 获得 G 的 IFFT：

$g = \text{ifft2}(g)$

7. 从 $g_p(x, y)$ 的左上象限提取 $M \times N$ 区域

8. 转化为 uint8 型得到最终结果 $g(x, y)$

报告正文

问题讨论

1. 第一问

我们知道巴特沃斯滤波器和高斯滤波器在频域的表达形式为
高斯低通滤波器：

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

巴特沃斯低通滤波器：

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

D_0 表示截止频率。

其中

$$D(u, v) = \left[\left(u - \frac{P}{2} \right)^2 + \left(v - \frac{Q}{2} \right)^2 \right]^{1/2}$$

我们根据这个原理首先在matlab中涉及一个矩阵D，其意义为各个点到中心距离($P/2, Q/2$)的欧式距离。代码如下：

```
function [U,V]=dftuv(M,N)
u=single(0:(M-1));
v=single(0:(N-1));
idx=find(u>M/2);
u(idx)=u(idx)-M;
idy=find(v>N/2);
v(idy)=v(idy)-N;
[V,U]=meshgrid(v,u);
end
function D=hypot1(M,N)
[U,V]=dftuv(M,N);
DSQ=sqrt(U.^2+V.^2);
D=fftshift(DSQ);
end
```

通过这个函数得到矩阵D后根据表达式可以设计出相应的巴特沃斯低通滤波器和高斯低通滤波器。通过选择合适的 D_0 达到滤波效果

2. 第二问

同上一问我们从书中得到高斯高通滤波器和巴特沃斯高通滤波器的表达式如下：

高斯滤波器：

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

巴特沃斯低通滤波器：

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

D_0 表示截止频率。通过调用函数 $D=\text{hypot1}(M, N)$ 就可生成 $M \times N$ 的滤波模型

通过选择合适的 D_0 达到滤波效果

3. 第三问

其他的高通滤波器

Unmask

$$g(x, y) = F^{-1}\{[1 + k * [1 - H_{LP}(u, v)]]F(u, v)\}$$

其中 k 取 1，对应着非锐化掩蔽算法。

上式计算最终得到的 $g(x, y)$ 是锐化的结果，即高通边缘部分与原图叠加的结果。

Laplace 算法

$$H(u, v) = -4\pi^2((u - P/2)^2 + (v - \frac{Q}{2})^2)$$

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

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

因为 $H(u, v)$ 是负的，所以这里 c 取 -1。

在空域计算时， $f(x, y)$ 与 $\nabla^2 f(x, y)$ 具有可比的数值，可以直接相加；而频域计算 $\nabla^2 f(x, y)$ 时会引入已标定系数的 DFT，这些系数的幅度与 f 的最大值相比可能要大几个数量级。

得到的 $g(x, y)$ 是锐化后的图像， $-\nabla^2 f(x, y)$ 可以理解为细节信息，将细节信息加到原图像上，可以复原背景特性并保持拉普拉斯锐化处理后的效果。

最后通过标定，进行可视化。

结果及分析

1. 第一问

Test1 经低通滤波效果

原图



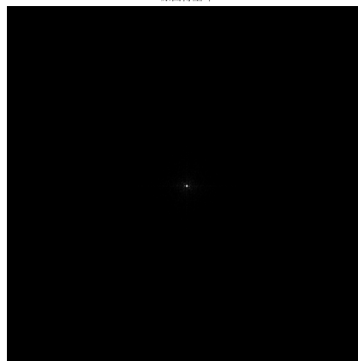
低通高斯结果



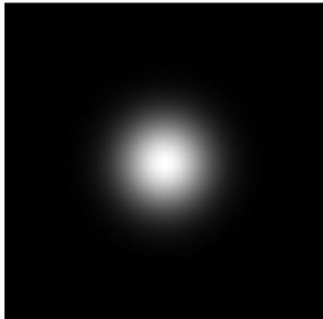
低通巴特沃斯结果



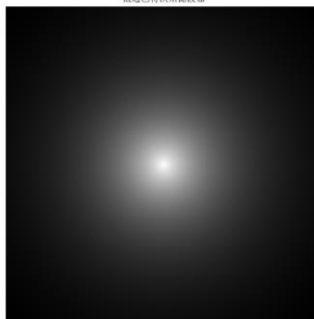
原图傅里叶



低通高斯滤波器



低通巴特沃斯滤波器



高斯与巴特沃斯的D0分别为256和128

高斯与巴特沃斯的功率谱比分别为 0.9623和 0.9119

Test2

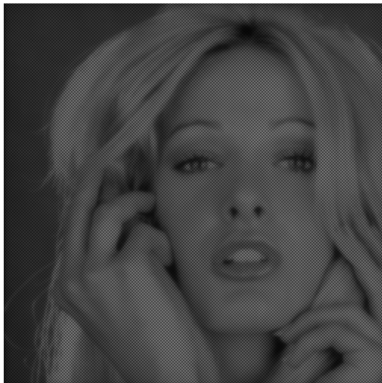
原图



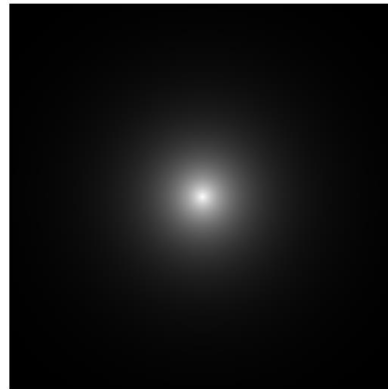
低通高斯结果



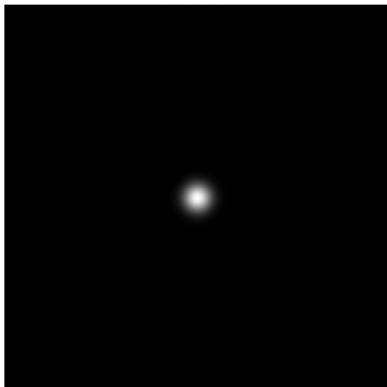
低通巴特沃斯结果



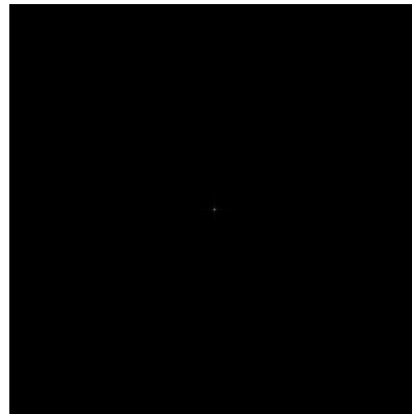
低通巴特沃斯滤波器



低通高斯滤波器



原图傅里叶



高斯与巴特沃斯的D0分别为256和64

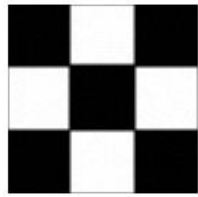
高斯与巴特沃斯的功率谱比分别为 0.8488和 0.9053

可以看出经低通滤波后，图像变得模糊高频噪声得以消除，但是总体图像变得模糊，且亮度变暗，高频部分的削弱导致部分功率损失所以得出上面的图像。**Test1**因为低频部分较多所以效果不明显，而**test2**因为高频部分多所以效果很明显。

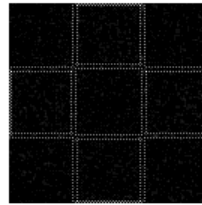
2. 第二问

Test3

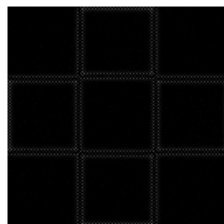
原图



高通巴特沃斯结果



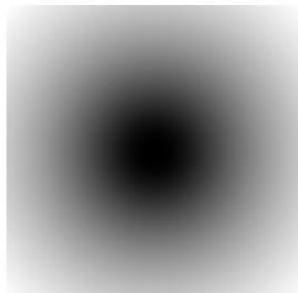
高通高斯结果



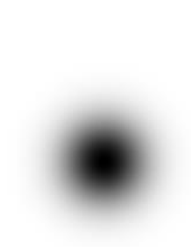
原图傅里叶



高通巴特沃斯滤波器



高通高斯滤波器



高斯与巴特沃斯的D0均为25

高斯与巴特沃斯的功率谱比分别为

2.8346e-04和 0.0136

Test4

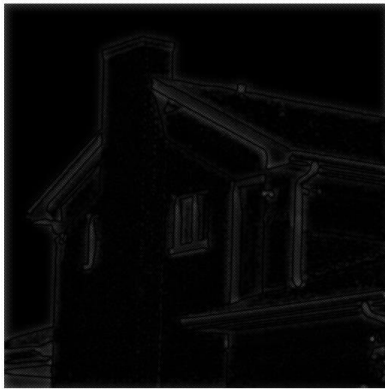
原图



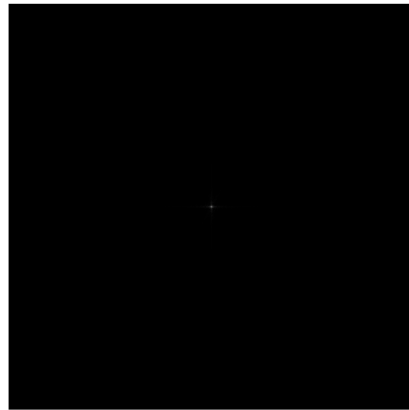
高通巴特沃斯结果



高通高斯结果



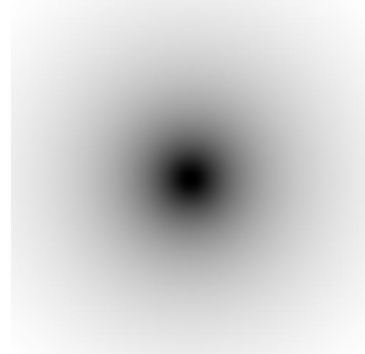
原图傅里叶



高通高斯滤波器



高通巴特沃斯滤波器



高斯与巴特沃斯的 D_0 均为25

高斯与巴特沃斯的功率谱比分别为 0.0022和 0.0285

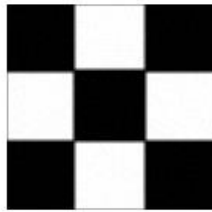
Test4

高通滤波器如果不凸显图像的话只有边缘被显示出来，可以达到边缘检测的目的，提高截止频率会增大功率谱比，从效果上看二阶的巴特沃斯效果比高斯的视觉效果强许多，但高斯滤的成分过多，所以在实际应用中，高斯的 D_0 应该减小

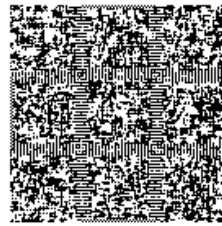
3. 第三问

Test3

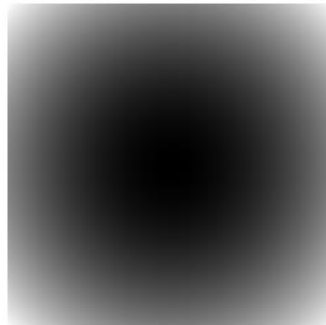
原图



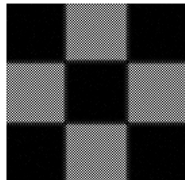
拉普拉斯结果



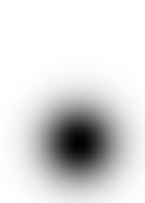
拉普拉斯滤波器



UNSHAP结果



UNSHAP



Test4

原图

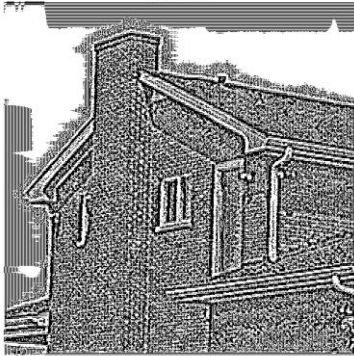


UNSHAP结果

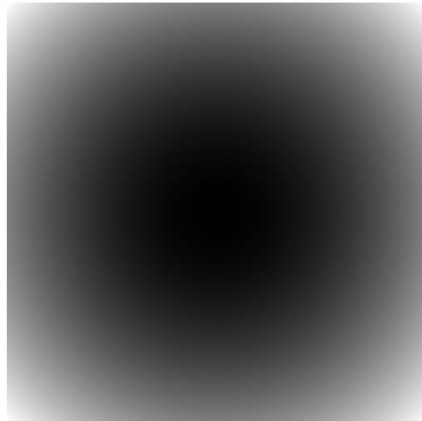


UNSHAP

拉普拉斯结果



拉普拉斯滤波器



拉普拉斯算子使用来计算二阶微分的，对细节信息要比梯度敏感得多，从图中我们也可以看到，在方块中分布的噪声信息很明显。而 test4 中房屋边缘不平滑的位置也有较多的毛刺噪声。

最后心得

频域滤波的主要大体思想是将图片矩阵转到频域，并将空域的卷积转换为频率域的成绩，再将频域结果转化为空域。在处理时为了处理方便我们将图片频谱以及滤波器频谱全部中心化，所以简单地理解来看，低通频域滤波对应的是中心通过，边缘不通过的频谱，频域高通滤波器则相反。且频域滤波处理与 Project3 中的一些线性空域滤波存在着对应关系，同时频域滤波器无法实现非线性滤波而空域滤波器则可以。

空间域和频率域滤波间的纽带是卷积定理。即：

$$h(x, y) \longleftrightarrow H(u, v)$$

空间域的卷积模板根据处理图像尺寸的不同，得到的频率域不同；我们知道图像空域与频域大小相同，在进行频率域滤波时，我们需要对源图像和模板进行同时补零扩充，所以频域结果是面向问题的。空域补零，频域采样间隔减小。

在实践中，我们宁愿使用小的卷积模板来实现卷积滤波，在硬件和固件中实现时速度快而容易；不过滤波的概念在频率域更加直观。取两个域的特性的优点的一个方法就是在频率域规定一个滤波器，得到它的 IDFT 全尺寸模板；或者可以给定小尺寸空间滤波器，得到其全尺寸频率域表示。

频率域滤波器仅限于**线性滤波**；而空域滤波器可以实现**线性滤波与非线性滤波**。

空域技术中的卷积操作都是基于**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，具有全局的性质，有可能更好地体现图像的整体特性，如平均亮度和对比度。

附录

参考文献:

1. 数字图像处理: MATLAB 版: 本科教学版/ (美) Gonzalez,R.C, woods, R.E, Eddins, S.L. 著; 阮秋琦译.—2 版.—北京: 电子工业出版社, 2014.1
2. 数字图像处理: 第三版/ (美) 拉斐尔·冈萨雷斯, (美) 理查德·伍兹著; 阮秋琦等译.—北京: 电子工业出版社, 2017.5

代码:

Fifth1.m

```
clear;
[f,map_f]=imread('C:\Users\15pr\Desktop\ÐÃ¼"îÄ¼p¼Ð
(2)\µÛ5'î×÷Òµ\test2.tif');
figure
imshow(f)
title('Ô-î¼');
f = single(f);
PQ =2*size(f);
F=fft2(f,PQ(1),PQ(2));
F=fftshift(F);

pinyutuxiang(F);
title('Ô-î¼,µÄiÒ¶')
H1=batewosi(PQ(1),PQ(2),3,512,1);
pinyutuxiang(H1);
title('µíí"°îîØîÖÈ¹ÄÈ²"Æ÷')
G1=H1.*F;
g1=ifft2(G1);
g1=g1(1:size(f,1),1:size(f,2));
g1=uint8(g1);
figure;
imshow(g1)
title('µíí"°îîØîÖÈ¹¼á¹û')
Pf=abs(F).^2;
Pg1=abs(G1).^2;
Pf=sum(Pf(:));
Pg1=sum(Pg1(:));
P=Pg1/Pf
H2=gaosi(PQ(1),PQ(2),25,1);
pinyutuxiang(H2);
title('µíí",ßÈ¹ÄÈ²"Æ÷')
G2=H2.*F;
g2=ifft2(G2);
g2=g2(1:size(f,1),1:size(f,2));
g2=uint8(g2);
```

```

figure;
imshow(g2)
title('píí',ßË¹¼á¹û')
Pg2=abs(G2).^2;
Pg2=sum(Pg2(:));
P=Pg2/Pf
function [U,V]=dftuv(M,N)
u=single(0:(M-1));
v=single(0:(N-1));
idx=find(u>M/2);
u(idx)=u(idx)-M;
idy=find(v>N/2);
v(idy)=v(idy)-N;
[V,U]=meshgrid(v,u);
end
function D=hypot1(M,N)
[U,V]=dftuv(M,N);
DSQ=sqrt(U.^2+V.^2);
D=fftshift(DSQ);
end
function H=batewosi(M,N,n,D0,type)
if(type==1)
D=hypot1(M,N);
H=(1+D./D0).^(-2*n);
elseif(type==2)
D=hypot1(M,N);
H=(1+D0./D).^(-2*n);
else
error('Wrong')
end
end
function H=gaosi(M,N,D0,type)
if(type==1)
D=hypot1(M,N);
H=exp(-(D.^2)/(2*(D0^2)));
elseif(type==2)
D=hypot1(M,N);
H=1-exp(-(D.^2)/(2*(D0^2)));
else
error('Wrong')
end
end
function pinyutuxiang(F)

```

```

F=abs(F);
figure
imshow(F,[]);
end
fifth3.m
clear;
[f,map_f]=imread('C:\Users\15pr\Desktop\ÐÂ½"îÄ¼p¼Ð
(2)\µÚ5'î×÷Òµ\test4 copy.bmp');
figure
imshow(f)
title('Ô-î¼');
f = single(f);
PQ =2*size(f);
F=fft2(f,PQ(1),PQ(2));
F=fftshift(F);

pinyutuxiang(F);
title('Ô-î¼,µÄiÒ¶')
H1=batewosi(PQ(1),PQ(2),2,35,2);
pinyutuxiang(H1);
title('„ßí"°îÒîÖÈ¹ÄÈ²"Æ÷')
G1=H1.*F;
g1=ifft2(G1);
g1=g1(1:size(f,1),1:size(f,2));
g1=uint8(g1);
g1=g1*16;
figure;
imshow(g1)
title('„ßí"°îÒîÖÈ¹¼á¹û')
Pf=abs(F).^2;
Pg1=abs(G1).^2;
Pf=sum(Pf(:));
Pg1=sum(Pg1(:));
P=Pg1/Pf
H2=gaosi(PQ(1),PQ(2),25,2);
pinyutuxiang(H2);
title('„ßí"„ßÈ¹ÄÈ²"Æ÷')
G2=H2.*F;
g2=ifft2(G2);
g2=g2(1:size(f,1),1:size(f,2));
g2=uint8(g2);
figure;
imshow(g2)

```

```

title('βÍ´βË¹¼á¹û')
Pg2=abs(G2).^2;
Pg2=sum(Pg2(:));
P=Pg2/Pf
Fifth3.m
clear;
[f,map_f]=imread('C:\Users\15pr\Desktop\ÐÂ½´îÄ¼p¼Ð
(2)\µÚ5´î×÷Òµ\test3_corrupt.pgm');
figure
imshow(f)
title('Ô-î¼');
f = single(f);
PQ =2*size(f);
F=fft2(f,PQ(1),PQ(2));
F=fftshift(F);
pinyutuxiang(F);

title('Ô-î¼,µÀiÒ¶')
D=hypot1(PQ(1),PQ(2));
H1=1-4*pi*pi*(D.^2);
pinyutuxiang(H1);
title('À-ÆÕÀ-Ë¹ÂË²´Æ÷')
G1=H1.*F;
g1=ifft2(G1);
g1=g1(1:size(f,1),1:size(f,2));
g1=uint8(g1);
figure;
imshow(g1)
title('À-ÆÕÀ-Ë¹¼á¹û')
k1=1;k2=1;
D=hypot1(PQ(1),PQ(2));
H1=gaosi(PQ(1),PQ(2),25,2);
H1=k1+k2.*H1;
pinyutuxiang(H1);
title('UNSHAP')
G1=H1.*F;
g1=ifft2(G1);
g1=g1(1:size(f,1),1:size(f,2));
g1=uint8(g1);
figure;
imshow(g1)
title('UNSHAP¼á¹û')

```