



# 数字图像处理作业三

数据科学与计算机学院 17大数据与人工智能

17341015 陈鸿峥

本次作业包括PROJECT 04-01到PROJECT 04-05五个实验。

## 一、 二维快速傅里叶变换(PROJECT 04-01)

### 1. 原理

- (a) 由于 $\Im[f(x, y)(-1)^{x+y}] = F(u - M/2, v - N/2)$ ，故将原图乘上 $(-1)^{x+y}$ 后可实现频率域的中心平移
- (b) 直接乘上常数 $c$ 即可
- (c) matlab的`ifft2`即可完成逆傅里叶变换
- (d) 同(a)，恢复原图
- (e) 即复数的模， $|F(u, v)| = \sqrt{R^2(x, y) + I^2(x, y)}$

## 二、 傅里叶谱与平均值(PROJECT 04-02)

### 1. 原理

利用PROJECT 04-01中实现的函数即可求出图片的中心化傅里叶谱。

### 2. 实验结果与分析

傅里叶谱的展示通过了一个伽马变换。

Fig.4.18(a)原图

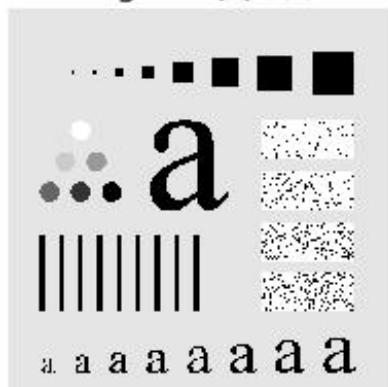
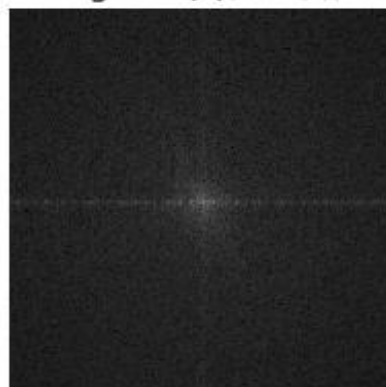
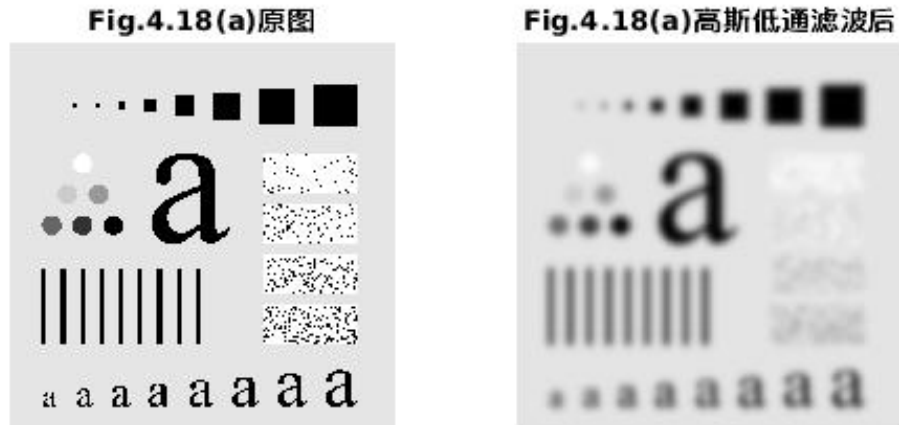


Fig.4.18(a)傅里叶谱





### 三、低通滤波器(PROJECT 04-03)

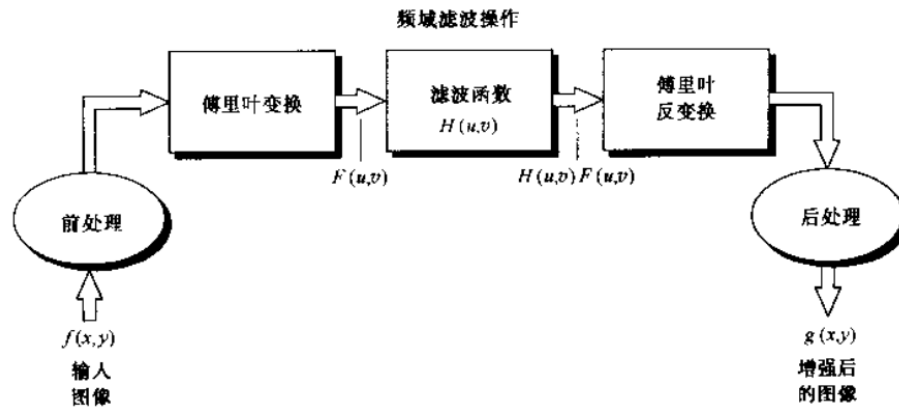
#### 1. 原理

二维高斯滤波器由下式给出

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

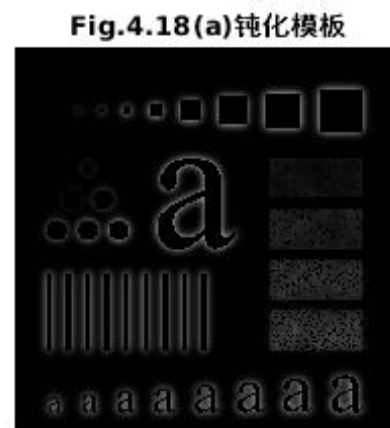
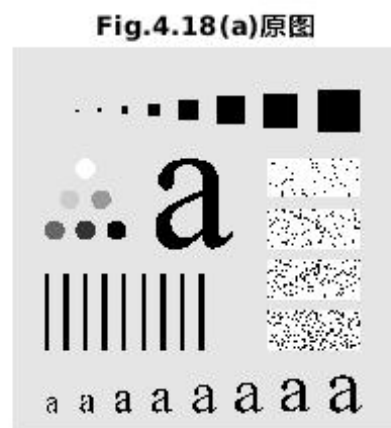
其中 $D(u, v)$ 为距傅里叶变换原点的距离， $\sigma$ 为高斯曲线扩展的程度。

频率域滤波的流程如下图所示。



1. 用 $(-1)^{x+y}$ 乘上输入图像来进行中心变换
2. 由上面得到的结果计算图像的DFT，得到 $F(u,v)$
3. 用滤波器函数 $H(u,v)$ 乘上 $F(u,v)$
4. 计算上述结果的IDFT
5. 取上式的实部，并乘上 $(-1)^{x+y}$ 恢复图像

## 2. 实验结果与分析



## 四、 高通滤波器(PROJECT 04-04)

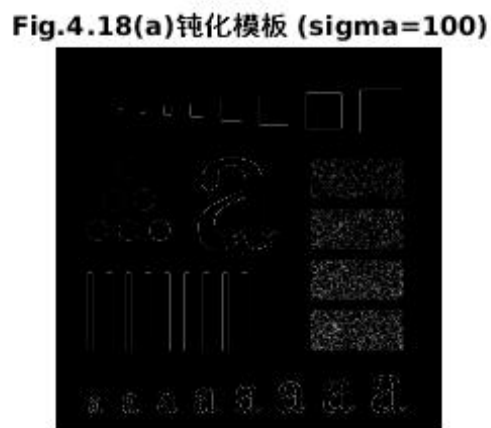
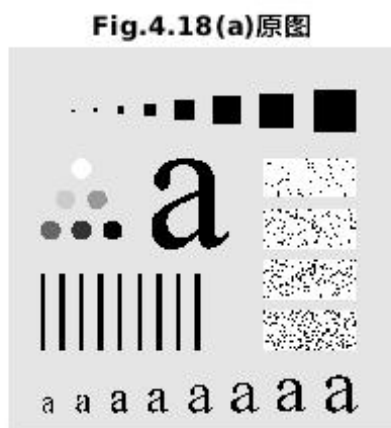
### 1. 原理

利用下式可以得到高通滤波的图像

$$f_{hp}(x, y) = f(x, y) - f_{lp}(x, y)$$

其中 $f_{lp}$ 即为上面通过高斯滤波后的图像。

### 2. 实验结果与分析



## 五、 频率域的相关(PROJECT 04-05)

### 1. 原理

利用空间域的相关与频率域的乘积构成傅里叶变换对这一性质

$$f(x, y) \circ h(x, y) \iff F^*(u, v)H(u, v)$$

## 2. 实验结果与分析



### 附录 A. 完整PROJECT代码

```
close all;clear all;clc;

% PROJECT 04-02
% (a)
I = imread('Fig0418(a).tif');
I = double(I);
[m,n] = size(I);
fftI = fft2(centerize(I));
sp = spectrum(fftI);

% (c)
s = sum(sum(abs(fftI)));
avg = s / (m * n);

% (b)
figure,
subplot(121),imshow(uint8(I));
title('Fig.4.18(a)原图')
```

```

subplot(122),imshow(sp.^0.2,[]);
title('Fig.4.18(a)傅里叶谱')

% PROJECT 04-03 (b)
gimg = gauss_lowpass(I,m/2,n/2,15);
figure,
subplot(121),imshow(uint8(I));
title('Fig.4.18(a)原图')
subplot(122),imshow(uint8(gimg));
title('Fig.4.18(a)高斯低通滤波后')

% PROJECT 04-04
% (a)
simg = I - gimg;
figure,
subplot(121),imshow(uint8(I));
title('Fig.4.18(a)原图')
subplot(122),imshow(uint8(simg));
title('Fig.4.18(a)钝化模板')

% (b)
simg2 = I - gauss_lowpass(I,m/2,n/2,100);
figure,
subplot(121),imshow(uint8(I));
title('Fig.4.18(a)原图')
subplot(122),imshow(uint8(simg2));
title('Fig.4.18(a)钝化模板 (sigma=100)')

% PROJECT 04-05
I1 = imread('Fig0441(a).jpg');
I2 = imread('Fig0441(b).jpg');
[m1,n1] = size(I1);
[m2,n2] = size(I2);
P = 298;
Q = 298;
img1 = zeros(P,Q);
img2 = zeros(P,Q);
img1(1:m1,1:n1) = I1(1:m1,1:n1);
img2(1:m2,1:n2) = I2(1:m2,1:n2);
cimg1 = centerize(img1);
cimg2 = centerize(img2);
f1 = fft2(cimg1);
f2 = fft2(cimg2);
rel = conj(f1).* f2;
% rel = f2 .* conj(f1);

```

```

newI = recover(iff2(rel));
figure,
subplot(131),imshow(uint8(I1));
title('Fig.4.41(a)原图')
subplot(132),imshow(uint8(I2));
title('Fig.4.41(b)原图')
subplot(133),imshow(uint8(newI.^0.3));
title('Fig.4.41图像相关')

% PROJECT 04-03 (a)
function g = gauss_lowpass(img,center_x,center_y,sig)
    [M,N] = size(img);
    [X,Y] = meshgrid(1:M,1:N);
    D = (X - center_x).^2 + (Y - center_y).^2;
    H = exp(-D/(2*sig^2));
    cimg = centerize(img);
    f = fft2(cimg);
    g = centerize(real(iff2(H.*f)));
end

% PROJECT 04-01
% (a)
function g = centerize(img)
    [M,N] = size(img);
    [X,Y] = meshgrid(1:M,1:N);
    ones = (-1).^(X+Y);
    g = ones.*img;
end

% (b)
function g = mul_real(A,c)
    % g = c * real(A) + c * imag(A) * i;
    g = c * A;
end

% (c)
function g = inverse_fft(A)
    g = iff2(A);
end

% (d)
function g = recover(A)
    g = centerize(real(A));
end

```



```
% (e)
function g = spectrum(A)
    g = abs(A);
end
```