Github 账号：[117503445](https://github.com/117503445)

实验摘要： 使用 MATLAB，利用频域完成图片的合成分解，音频重采样，图片盲水印。

实验题目

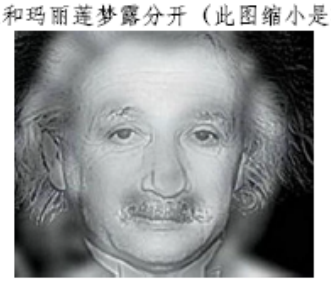
信号与系统实验（四）

1. 图片的高频信息与低频信息

   合成图片。找两张轮廓比较像的图片A和B，有一张是你本人。提取一张照片的低频信息，另一张图片的高频信息，结合这两个照片。设置不同的频率门限，组合照片，组合的效果是，放大看是A，缩小看是B。例如以下两张图片

   合成后为

   分解图片。把下图爱因斯坦和玛丽莲梦露分开（此图缩小是玛丽莲梦露，截取不同的频段）



   部分代码见“图片的合成.m”链接：http://pan.baidu.com/s/1dFtd0qP**密码：**1oa4

2. 采集一段人说话时的声音(一般最高频率在4kHz左右)，并进一步经过若干次取样，得到对同一段连续信号在不同取样频率下的离散信号，例如最初的取样率是44kHz，经过下取样后可以得到22kHz、11kHz、5.5kHz、2.75kHz等频率下的取样结果。试针对该信号及其取样信号，分析取样率对信号重构的影响。

3. **频域制作数字盲水印和去除数字盲水印** https://www.zhihu.com/question/50735753，看懂，想想，有想法写出来，做一个好玩的东西。

实验内容

1.

Input

1-1-in1.png

1-1-in2.png

Output

1-1-out.png

Code

```
I1 = imread('1-1-in1.png');
g1 = rgb2gray(I1);
s = fftshift(fft2(g1));
[M, N] = size(s);
n1 = fix(M / 2);
n2 = fix(N / 2);

%理想低通滤波器取 d0=10  （15,30）可变
d0 = 10;

for i = 1:M

    for j = 1:N
        d = sqrt((i - n1)^2 + (j - n2)^2);

        if d < d0
            h = 1;
        else
            h = 0;
        end
```

```
        s(i, j) = h * s(i, j);
    end

end

s = ifftshift(s);
s = uint8(real(ifft2(s)));
figure(1);
imshow(s);

I2 = imread('1-1-in2.png');
g2 = rgb2gray(I2);
s2 = fftshift(fft2(g2));
[M2, N2] = size(s2);
n12 = fix(M2 / 2);
n22 = fix(N2 / 2);

%理想高通滤波器取 d02=5  （15,30）可变
d02 = 10;

for i = 1:M2

    for j = 1:N2
        d = sqrt((i - n12)^2 + (j - n22)^2);

        if d < d02
            h = 0;
        else
            h = 1;
        end

        s2(i, j) = h * s2(i, j);
    end

end
```

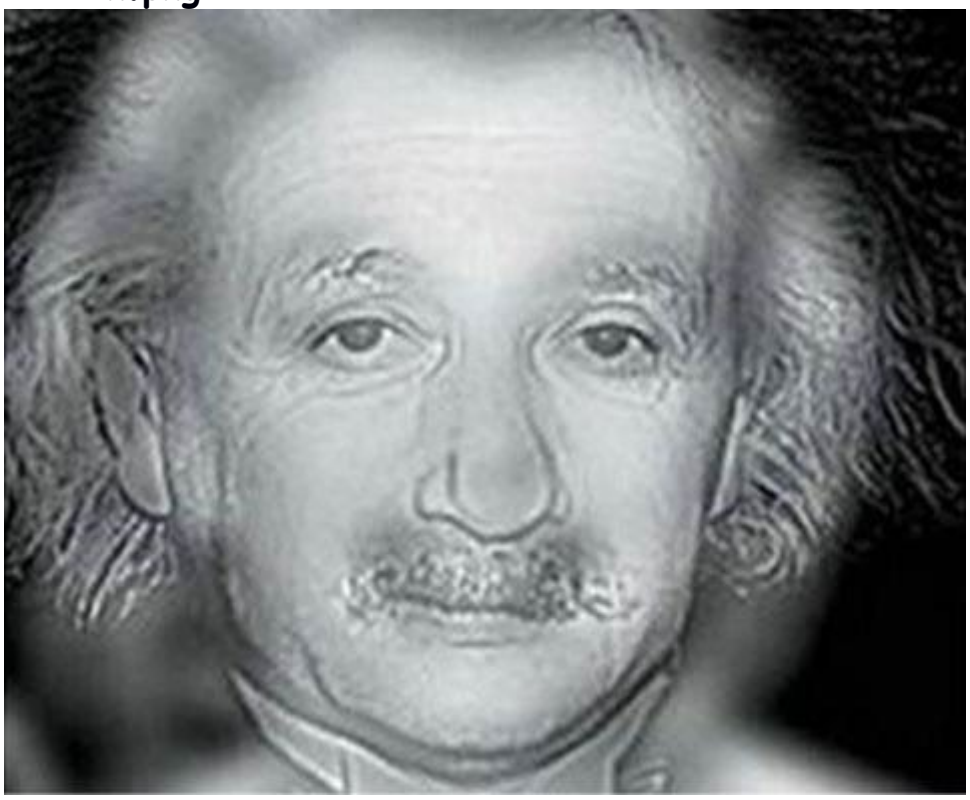
```
s2 = ifftshift(s2);
s2 = uint8(real(ifft2(s2)));
figure(2);
imshow(s2);

%图片合并
s3 = imadd(s, s2);
figure(3);
imshow(s3);
imwrite(s3, '1-1-out.png', 'PNG')
```

2.

Input

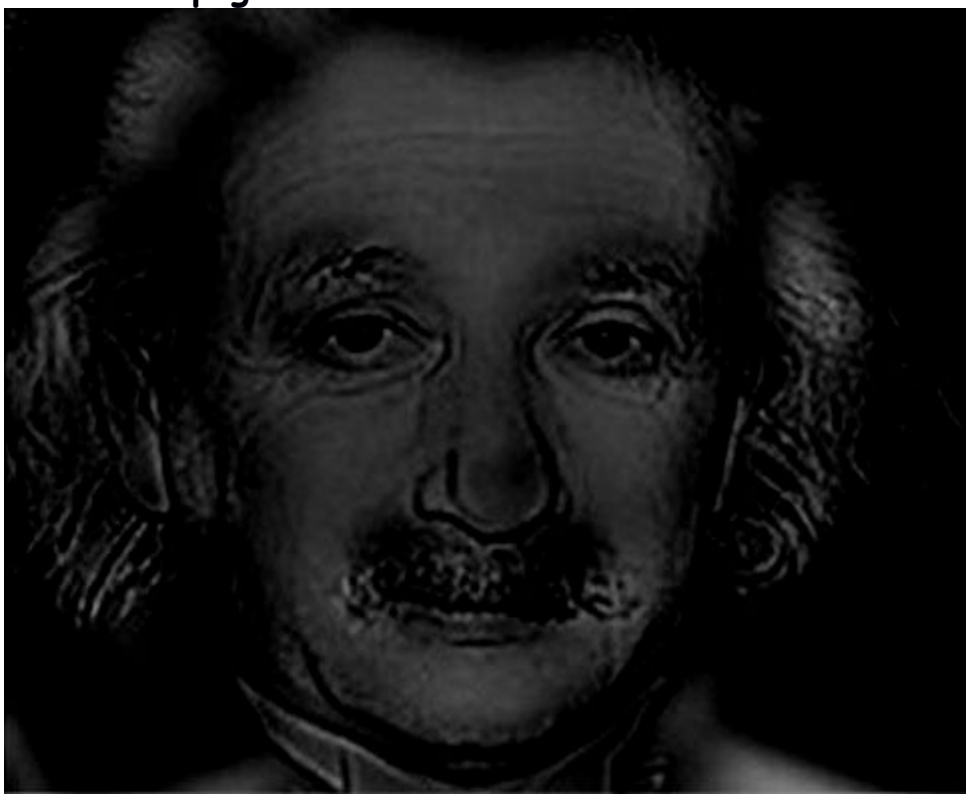
1-2-in.png



Output

1-2-out1.png

1-2-out2.png



Code

```
I1 = imread('1-2-in.png');
g1 = rgb2gray(I1);
s = fftshift(fft2(g1));
[M, N] = size(s);
```

```
n1 = fix(M / 2);
n2 = fix(N / 2);

%理想低通滤波器取 d0=10  (15,30) 可变
d0 = 10;

for i = 1:M

    for j = 1:N
        d = sqrt((i - n1)^2 + (j - n2)^2);

        if d < d0
            h = 1;
        else
            h = 0;
        end

        s(i, j) = h * s(i, j);
    end

end

s = ifftshift(s);
s = uint8(real(ifft2(s)));
figure(1);
imshow(s);
imwrite(s, '1-2-out1.png', 'PNG')

I2 = imread('1-2-in.png');
g2 = rgb2gray(I2);
s2 = fftshift(fft2(g2));
[M2, N2] = size(s2);
n12 = fix(M2 / 2);
n22 = fix(N2 / 2);

%理想高通滤波器取 d02=5  (15,30) 可变
```

```
d02 = 2;

for i = 1:M2

    for j = 1:N2
        d = sqrt((i - n12)^2 + (j - n22)^2);

        if d < d02
            h = 0;
        else
            h = 1;
        end

        s2(i, j) = h * s2(i, j);
    end

end

s2 = ifftshift(s2);
s2 = uint8(real(ifft2(s2)));
figure(2);
imshow(s2);
imwrite(s2, '1-2-out2.png', 'PNG')
```

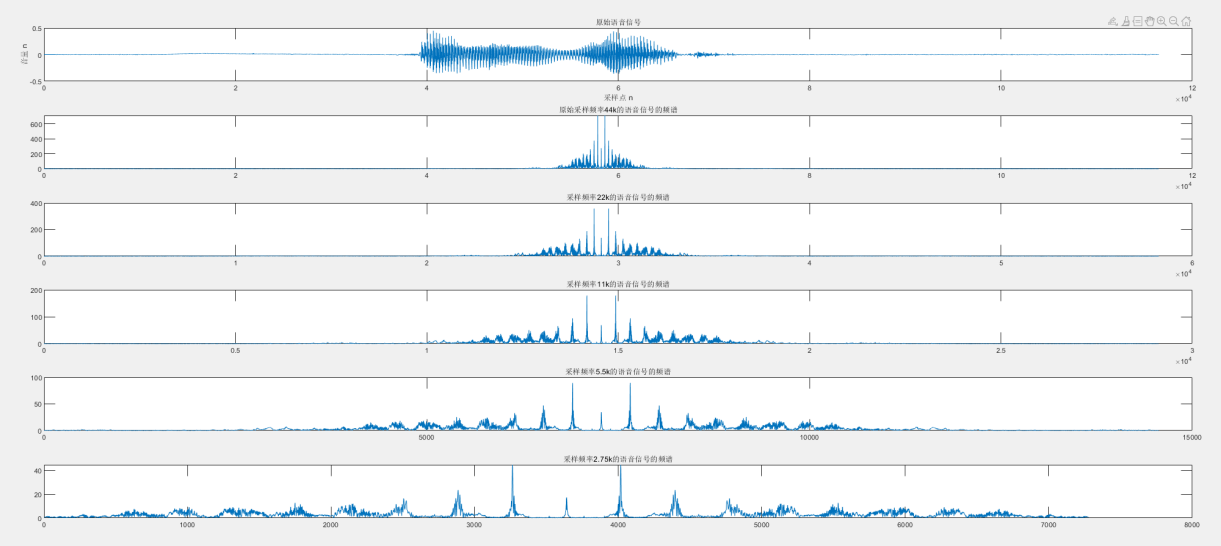
2.

Input

2-in-hello-world.wav

齐浩天自己录制的音频，人声，“Hello World”。

Output



通过 sonud 播放声音，可以明显观察到随着采样率降低，声音失真，变得低沉。

```
Fs = 44100;
file = '2-in-hello-world.wav';

[x1,Fs] = audioread(file);%x1 为所读取的音频数据,Fs 为采样频率
sound(x1,Fs); %播放音乐
figure(1);
subplot(611);
plot(x1);%做原始语音信号的时域图形
title('原始语音信号')
xlabel('采样点 n');
ylabel('音量 n');

y1=fft(x1); %做 length(x1)点的 FFT
y1=fftshift(y1);%频率分量将会移到坐标中心
subplot(612);
plot(abs(y1));%画出原始语音信号的频谱图，这里保证了 x 轴的点数必须和 y 轴点数一致
title('原始采样频率 44k 的语音信号的频谱');

x=resample(x1,1,2);
sound(x,Fs/2);
y=fft(x);
```

```
y=fftshift(y);%频率分量将会移到坐标中心
subplot(613);
plot(abs(y));%画出原始语音信号的频谱图，这里保证了x轴的点数必须和y轴点数一致
title('采样频率 22k 的语音信号的频谱');


x=resample(x,1,2);
sound(x,Fs/4);
y=fft(x);
y=fftshift(y);%频率分量将会移到坐标中心
subplot(614);
plot(abs(y));%画出原始语音信号的频谱图，这里保证了x轴的点数必须和y轴点数一致
title('采样频率 11k 的语音信号的频谱');

x=resample(x,1,2);
sound(x,Fs/8);
y=fft(x);
y=fftshift(y);%频率分量将会移到坐标中心
subplot(615);
plot(abs(y));%画出原始语音信号的频谱图，这里保证了x轴的点数必须和y轴点数一致
title('采样频率 5.5k 的语音信号的频谱');

x=resample(x,1,2);
sound(x,Fs/16);
y=fft(x);
y=fftshift(y);%频率分量将会移到坐标中心
subplot(616);
plot(abs(y));%画出原始语音信号的频谱图，这里保证了x轴的点数必须和y轴点数一致
title('采样频率 2.75k 的语音信号的频谱');

3
```

Input

3-in-girl.png

3-in-mark.png

Output

3-output-watermarked.png

```
clc;
clear;
close all;
alpha = 1;

im = double(imread('3-in-girl.png')) / 255;
mark = double(imread('3-in-mark.png')) / 255;
imsize = size(im);

TH = zeros(imsize(1) * 0.5, imsize(2), imsize(3));
TH1 = TH;
TH1(1:size(mark, 1), 1:size(mark, 2), :) = mark;
M = randperm(0.5 * imsize(1));
N = randperm(imsize(2));

for i = 1:imsize(1) * 0.5

    for j = 1:imsize(2)
        TH(i, j, :) = TH1(M(i), N(j), :);
    end

end

imsize = size(im);
%random
```

```
TH = zeros(imsize(1) * 0.5, imsize(2), imsize(3));
TH1 = TH;
TH1(1:size(mark, 1), 1:size(mark, 2), :) = mark;
M = randperm(0.5 * imsize(1));
N = randperm(imsize(2));

for i = 1:imsize(1) * 0.5

    for j = 1:imsize(2)
        TH(i, j, :) = TH1(M(i), N(j), :);
    end

end

mark_ = zeros(imsize(1), imsize(2), imsize(3));
mark_(1:imsize(1) * 0.5, 1:imsize(2), :) = TH;

for i = 1:imsize(1) * 0.5

    for j = 1:imsize(2)
        mark_(imsize(1) + 1 - i, imsize(2) + 1 - j, :) = TH(i, j, :);
    end

end

FA = fft2(im);
FB = FA + alpha * double(mark_);
FAO = ifft2(FB);
figure, imshow(FAO);
title('watermarked image');
imwrite(abs(FAO), '3-output-watermarked.png');
```

实验总结

熟悉了基于 MATLAB 的频域手段处理图像音频等多媒体信号。

参考文献

https://zhuanlan.zhihu.com/p/31917473

https://ww2.mathworks.cn/help/matlab/ref/imread.html

https://ww2.mathworks.cn/help/matlab/ref/imwrite.html

https://ww2.mathworks.cn/help/matlab/ref/fftshift.html

https://ww2.mathworks.cn/help/matlab/ref/fft.html