

一、彩色图像人脸检测

基于颜色空间做人脸检测，即给定一张图片，将其中的人脸框出来。

1. 原理

主要基于文献 [1]中总结的三种方法进行人脸检测，实际上都是在不同颜色空间中做阈值变换。

在不同颜色空间中的人脸检测需要满足以下条件：

- RGB颜色空间：

$$\begin{cases} R > 95 \wedge G > 40 \wedge B > 20 \wedge (\max\{R, G, B\} - \min\{R, G, B\}) > 15 \\ \wedge |R - G| > 15 \wedge R > G \wedge R > B & \text{均匀照明} \\ R > 20 \wedge G > 210 \wedge B > 170 \wedge |R - G| < 15 \wedge R > G \wedge R > B & \text{潜在照明} \end{cases}$$

- HSV颜色空间：

$$V \geq 40 \wedge 0.2 \leq S \leq 0.6 \wedge 0 \leq H \leq 0.25$$

- YCbCr颜色空间：

$$75 < Cb < 250 \wedge 10 < Cr < 100 \wedge Y > 80$$

整体流程包括以下几步：

1. 图像预处理
2. 图像颜色空间的转换
3. 基于阈值的图像分割
4. 连通分量分析
5. 标注出人脸并显示

2. 实验结果与分析

可以发现基于阈值的划分方法其实并没有办法很好地区分人脸和其他人体部分。

二、直方图均衡化

观察对RGB图像的三个通道分别做直方图均衡，与对整个图像做直方图均衡是否一致。

1. 原理

直方图均衡化在第一个Proj中已经完成，主要分为以下几个步骤：

- 对原图片的像素进行统计，得到 $[0, 255]$ 中每一个灰度值的频数 n_k ，进而得到原图像的直方图
- 对原图像直方图进行归一化，由频数变为概率 $p_r(r_k) = n_k/n$ ，实即概率质量函数(PMF)¹
- 由概率质量函数求得累积质量函数(CMF)

$$P(r_k) = \sum_{j=0}^k p_r(r_j) = \sum_{j=0}^k \frac{n_j}{n}$$

- 将原图像的灰度值作为 x 输入，从CMF中得到对应的输出值 y ，并且对 y 重新恢复尺度 $[0, 255]$ ，得到新图片中该位置的灰度值，即

$$s_k = T(r_k) = 255 \cdot \sum_{j=0}^k \frac{n_j}{n}$$

注意需要对 s_k 进行取整

- 对原图像中的每一个像素都这么操作，可以得到直方图归一化后的新图像

在这个实验中，则直接将其封装为一个函数，并分别对各通道和全图进行应用。

2. 实验结果与分析

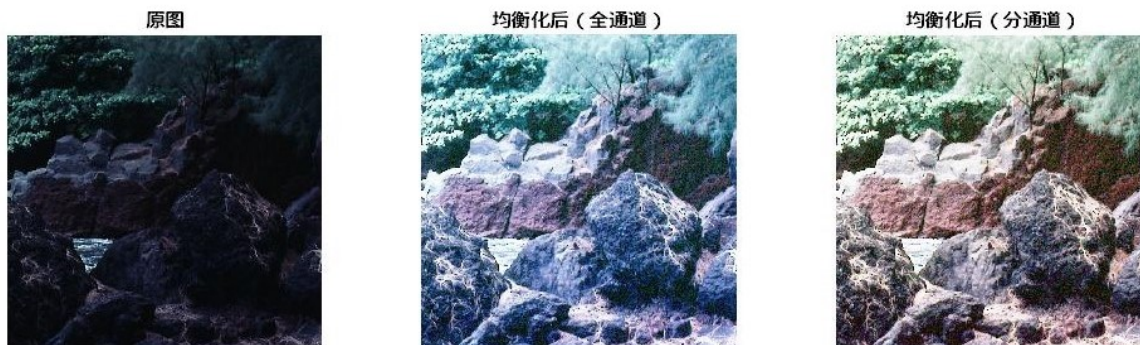
我主要采用了两幅图像进行直方图均衡化。首先第一幅图是Lenna的人像，由于原本图像的对比度已经很清晰，因此经过全通道均衡化后的图片变化不大（但细看确实对比图提升了）；而分通道均衡化后的图像则与原图会有巨大差异，主要体现在颜色上，如背景的浅红色经过分通道均衡化后被消除了。



第二幅图则是课本中比较暗的石头图片，这幅图片经过全通道直方图均衡化后，可以明显看出按部都被明显提升了；而分通道的均衡化同样也将对比度提升，同时图片不会偏蓝，会更

¹注：由于是离散情况，故不是概率密度函数

加自然。



三、带阻滤波器

利用带阻滤波器消除周期性噪声。

1. 原理

仅仅阻断中间一圆环的频率分量，而其他分量均可正常通过。下面是二维 n 阶巴特沃斯带阻滤波器的例子

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

其中 $D(u, v)$ 为距离中心点的距离， D_0 为截止频率， W 为带阻宽度。

2. 实验结果与分析

原图被sin的周期性噪声干扰。通过调整圆环的位置及宽度，我们可以看到经过带阻滤波器后周期性噪声被有效消除了。

Fig.5.16(a)原图

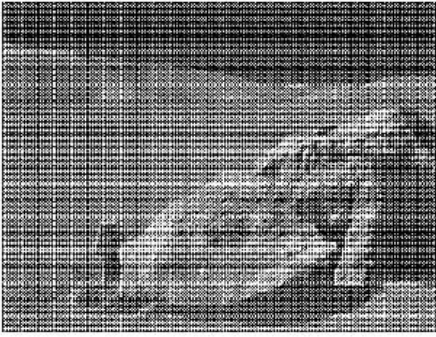
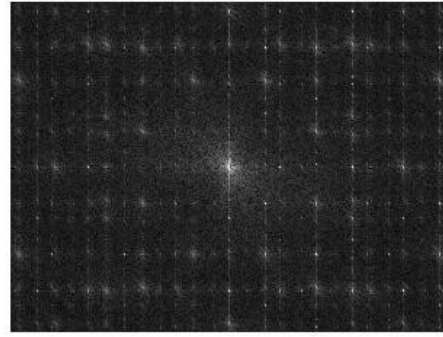
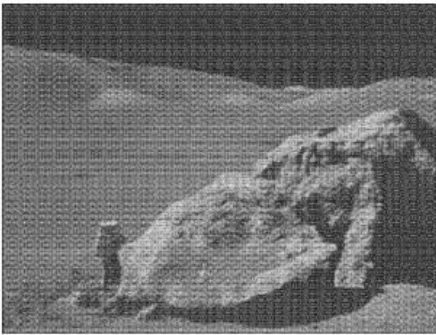


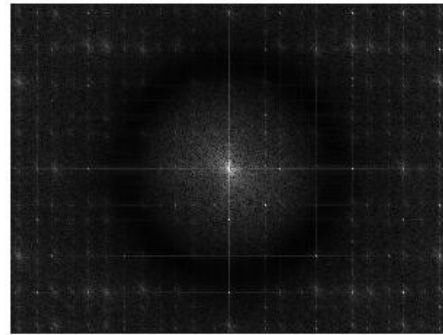
Fig.4.18(a)傅里叶谱



带阻滤波后结果



带阻滤波后傅里叶谱



附录 A. 图像人脸检测代码

附录 B. 直方图均衡化代码

```
close all;clear all;clc;
% I = imread('fig/Lenna.png');
I = imread('fig/Fig0635.tif');
[m,n,k] = size(I);

% all channels
newI = histeq(I);

% different channels
newI2 = zeros(m,n,3);
newI2(:, :, 1) = histeq(I(:, :, 1));
newI2(:, :, 2) = histeq(I(:, :, 2));
newI2(:, :, 3) = histeq(I(:, :, 3));

figure,
subplot(131),imshow(uint8(I));
title('原图')
```

```

subplot(132),imshow(uint8(newI));
title('均衡化后 (全通道)')
subplot(133),imshow(uint8(newI2));
title('均衡化后 (分通道)')

function g = histeq(img)
    A = zeros(1,256);
    for i = 1:256
        A(i) = sum(img == (i-1),'all');
    end
    A = double(A);
    A = A / prod(size(img));
    cumulation = zeros(1,256);
    for i = 2:256
        cumulation(i) = cumulation(i-1) + A(i);
    end
    g = uint8(cumulation(img+1)*255);
end

```

附录 C. 带阻滤波器

```

close all;clear all;clc;

I = imread('fig/Fig0516.tif');
I = double(I);
newI = filter(I);

figure,
subplot(221),imshow(uint8(I));
title('Fig.5.16(a)原图')
subplot(222),imshow(uint8(abs(fft2(centerize(I))).^0.4),[]); % (log(1 + sp),[]);
title('Fig.5.16(a)傅里叶谱')
subplot(223),imshow(uint8(newI));
title('带阻滤波后结果')
subplot(224),imshow(uint8(abs(fft2(centerize(newI))).^0.4),[]); % (log(1 + sp),[])
    ↪ ;
title('带阻滤波后傅里叶谱')

function g = filter(img)
    [M,N] = size(img);
    P = 2 * M; Q = 2 * N; % remember to do extension
    Iext = zeros(P,Q);
    Iext(1:M,1:N) = img(1:M,1:N);
    [Y,X] = meshgrid(1:Q,1:P);
    center_x = P/2; center_y = Q/2;

```

```
D = (X - center_x).^2 + (Y - center_y).^2;
D0 = 300^2;
W = 400;
n = 1;
% H = 1 - exp(-0.5*(D-55^2)/((sqrt(D)*5).^2));
H = 1 ./ (1 + ((sqrt(D) * W)./(D - D0)).^(2*n));
cimg = centerize(Iext);
f = fft2(cimg);
g = centerize(real(ifft2(H.*f)));
g = g(1:M,1:N);
end

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

参考文献

- [1] Rosali Mohanty, M.V Raghunadh, *Skin Color Segmentation based Face Detection using Multi-Color Space*, International Journal of Advanced Research in Computer and Communication Engineering.
- [2] Stanford EE368/CS232, https://web.stanford.edu/class/ee368/Project_03/project_03.html