

人脸识别项目

学生姓名：郝裕玮&张闯

截止日期：11.16

上交日期：11.15

摘要

本次实验内容为根据肤色实现人脸识别。

小组分工：

郝裕玮和张闯共同完成代码和报告的撰写以及 PPT 的制作

基于肤色的人脸识别

(1) 技术讨论：

膨胀：A 被 B 膨胀定义为：

$$A \oplus B = \{z | (\hat{B})_z \cap A \neq \emptyset\}$$

腐蚀：A 被 B 腐蚀定义为：

$$A \ominus B = \{z | (B)_z \subseteq A\}$$

开操作：开操作一般使对象的轮廓变得光滑，断开狭窄的间断和消除细的突出物。使用结构元素 B 对集合 A 进行开操作的定义为：

$$A \circ B = (A \ominus B) \oplus B$$

肤色检测主要方法：

```
从一篇文章了解到，肤色检测主要有以下七种方法：
1. RGB color space
2. YcrCb之cr分量+otsu阈值化
3. YCrCb中133<=Cr<=173 77<=Cb<=127
4. HSV中 7<H<20 28<S<256 50<V<256
5. 基于椭圆皮肤模型的皮肤检测
6. opencv自带肤色检测类AdaptiveSkinDetector

不过经作者自己检验，用RGB的方法受光线影响比较大，鲁棒性太低了，所以我们这次就不实验它了，留下一个判别式就好。
```

RGB 色彩模型和 YCbCr 色彩模型的互化变换公式：

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.1687 & -0.3313 & 0.500 \\ 0.500 & -0.4187 & -0.0813 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix}$$

肤色范围内的 Y，Cb，Cr 的最大值和最小值如下表所示：

像素色彩值	最大值	最小值
Y	230	87
Cb	123	93
Cr	173	134

项目流程：



代码如下所示（具体分析已包含在代码注释中）：

```
image = imread('C:\Users\93508\Desktop\3.png');% 读入图片
[row,col,rgb] = size(image);%记录图片尺寸大小(忽略亮度信息 rgb)
%展示原图
figure(1)
imshow(image);
title('原图');

%将图像从 RGB 格式转为 YCbCr 格式
YCbCr = rgb2ycbcr(image);
%提取图片的 YCbCr 分量：Y 亮度分量，Cb 蓝色分量，Cr 红色分量
Y = YCbCr(:,:,1);
Cb = YCbCr(:,:,2);
Cr = YCbCr(:,:,3);
image_gray = zeros(row,col);

%查阅资料可知，肤色点的 Y 值范围为(87,230)，Cb 值范围为(77,127)，Cr 值范围为
(133,173)
for i = 1:row
    for j = 1:col
        if(Y(i,j) > 87 && Y(i,j) < 230 && Cb(i,j) > 77 && Cb(i,j) < 127
&& Cr(i,j) > 133 && Cr(i,j) < 173)
            image_gray(i,j) = 255;
            %若为肤色点，则将其像素值置为 255（白色）
        else
            image_gray(i,j) = 0;
            %若不是肤色点，则将其像素值置为 0（黑色）
        end
    end
end
%展示肤色覆盖范围（白色表示肤色，黑色表示背景）
figure(2)
imshow(image_gray);
title('肤色覆盖范围黑白图');
```

```

%开运算：先腐蚀后膨胀
%开运算用来消除小物体、在纤细点处分离物体、平滑较大物体的边界的同时并不明显改变其面积
%图像腐蚀
SE = strel('disk',10);
image_erode = imerode(image_gray,SE);
figure(3)
imshow(image_erode);
title('图像腐蚀');
%图像膨胀
image_dilate=imdilate(image_erode,SE);
figure(4)
imshow(image_dilate);
title('图像膨胀');

%确定人脸边界
x_min = 10000;
x_max = 0;
y_min = 10000;
y_max = 0;
%找到横向的肤色边界值 x_min 和 x_max
for i = 1:row
    for j = 1:col
        if(image_dilate(i,j) == 255 && j < x_min)
            x_min = j;
        end
        if(image_dilate(i,j) == 255 && j > x_max)
            x_max = j;
        end
    end
end
%找到纵向的肤色边界值 y_min 和 y_max
for i = 1:row
    for j = 1:col
        if(image_dilate(i,j) == 255 && i < y_min)
            y_min = i;
        end
        if(image_dilate(i,j) == 255 && i > y_max)
            y_max = i;
        end
    end
end
end

```

```

%根据刚才确定的边界，对原图进行画框，标出人脸所在范围，框线颜色为红色
(R:240,G:65,B:85)
image(y_min:y_max,x_min,1)=uint8(240);
image(y_min:y_max,x_min,2)=uint8(65);
image(y_min:y_max,x_min,3)=uint8(85);

image(y_min:y_max,x_max,1)=uint8(240);
image(y_min:y_max,x_max,2)=uint8(65);
image(y_min:y_max,x_max,3)=uint8(85);

image(y_min,x_min:x_max,1)=uint8(240);
image(y_min,x_min:x_max,2)=uint8(65);
image(y_min,x_min:x_max,3)=uint8(85);

image(y_max,x_min:x_max,1)=uint8(240);
image(y_max,x_min:x_max,2)=uint8(65);
image(y_max,x_min:x_max,3)=uint8(85);

%画图
figure(5)
imshow(image);
title('人脸识别区域');

```

(2) 结果讨论：

原图像 1



肤色检测（见下页）：

肤色覆盖范围黑白图



图像腐蚀:

图像腐蚀



图像膨胀:

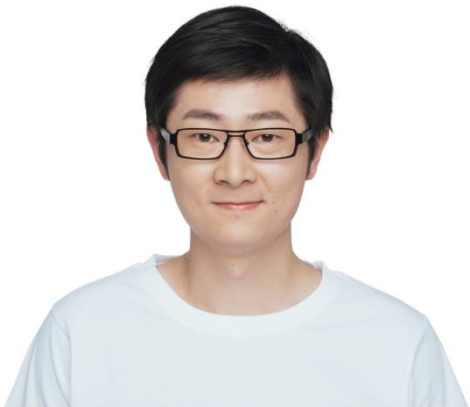
图像膨胀



图像定位（识别人脸区域）:



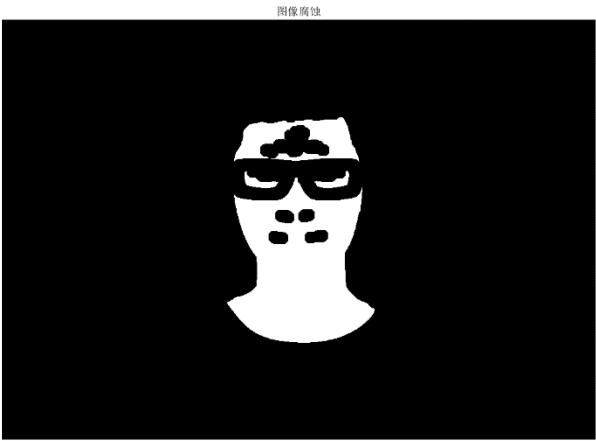
原图像 2



肤色检测:



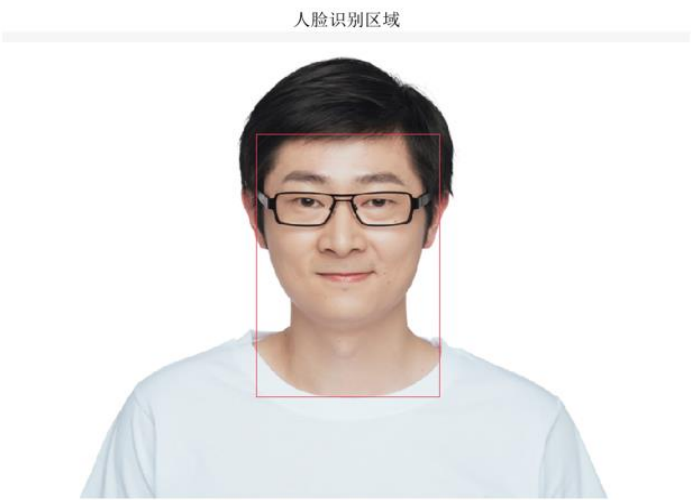
图像腐蚀:



图像膨胀:



图像定位 (识别人脸区域):



原图像 3



肤色检测:



图像腐蚀:



图像膨胀：



图像定位（识别人脸区域）：

