

数字图像处理

基于肤色的人脸检测

中山大学计算机学院 计算机科学与技术
19335174 施天予

本人实现的是 RGB 颜色空间的人脸检测。

一、原理

RGB 颜色空间的人脸检测，即给定一张图片，将其中的人脸框出来。我实现的方法主要基于 RGB 颜色空间人脸检测的阈值，将原图像做阈值变换转为二值图，对应人脸的部分为 1，其他部分为 0。在用方框把人脸框出来。

在 RGB 颜色空间中的人脸检测需要满足以下条件：

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

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

二、实验结果与分析

最终的人脸及图像标注如下图所示，可以看出在 RGB 空间还是能够比较好地将皮肤部分提取出来，但 RGB 空间会损失较多细节。同时我们也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，因此基于阈值的划分方法只能做简单的人脸检测处理，并且其阈值也是需要经过精细调整才能达到比较好的效果。

基于RGB的人脸检测



分割图



基于RGB的人脸检测



分割图



基于RGB的人脸检测



分割图



附录 A. 图像人脸检测代码

```

close all;clear all;clc;

I1 = imread('1.jpeg');
BW_rgb1 = rgb_face(I1);
bb_rgb1 = get_bb(BW_rgb1);
I2 = imread('2.jpeg');
BW_rgb2 = rgb_face(I2);
bb_rgb2 = get_bb(BW_rgb2);
I3 = imread('3.jpeg');
BW_rgb3 = rgb_face(I3);
bb_rgb3 = get_bb(BW_rgb3);

figure, subplot(1,2,2);
imshow(BW_rgb1)
title('分割图');
subplot(1,2,1);
imshow(I1);
rectangle('Position',bb_rgb1,'EdgeColor','r');
title('基于RGB的人脸检测');

figure, subplot(1,2,2);
imshow(BW_rgb2)
title('分割图');
subplot(1,2,1);
imshow(I2);
rectangle('Position',bb_rgb2,'EdgeColor','r');
title('基于RGB的人脸检测');

figure, subplot(1,2,2);
imshow(BW_rgb3)
title('分割图');
subplot(1,2,1);
imshow(I3);
rectangle('Position',bb_rgb3,'EdgeColor','r');
title('基于RGB的人脸检测');

function bw = rgb_face(I)
    [m,n,c] = size(I); % 获取图像
    BW = zeros(m,n);
    for i = 1:m
        for j = 1:n
            R = I(i,j,1);
            G = I(i,j,2);

```

```

        B = I(i,j,3);
        v = [R,G,B];
        if (R > 95 && G > 40 && B > 20) && ((max(v) - min(v) > 15 && abs(R-G) >
            ↪ 15 && R > G && R > B)) % 均匀照明 (白天)
            % if R > 20 && G > 210 && B > 170 && abs(R-G) < 15 && R > G && % R > B)
            ↪ % 潜在照明 (黑夜)
            BW(i,j) = 1;
        end
    end
end
bw = BW;
end

function g = get_bb(BW)
    L = bwlabel(BW,8); % 返回和BW相同大小的8连通矩阵
    BB = regionprops(L,'BoundingBox'); % 返回最小矩阵的结构体
    BB1 = struct2cell(BB); % 将结构体转换为元胞数组
    BB2 = cell2mat(BB1); % 元胞数组转为矩阵
    % 二值图所以连通只有1, 大小1*4n, 每4个[x,y,w,h]代表矩形的从点(x,y)开始绘制一个宽w
    ↪ 高h的矩形。

    [s1,s2] = size(BB2);
    max_area = 0;
    j = 3;
    for k = 3:4:s2-1
        area_bb = BB2(1,k) * BB2(1,k+1); % 计算值矩阵大小
        if area_bb > max_area && (BB2(1,k) / BB2(1,k+1)) < 1.8
            max_area = area_bb;
            j = k;
        end
    end
    g = [BB2(1,j-2),BB2(1,j-1),BB2(1,j),BB2(1,j+1)];
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.