HW5 作业报告

邱荻 2000012852 信息科学技术学院

#基础型 编程实现一种使用直方图技术的色彩迁移算法，自选展示度高和应用性强的样例图像验证和说明该算法的应用场景。

参考这篇博客的方法：https://developer.baidu.com/article/details/3151141

1. 计算源图像和目标图像的直方图
2. 归一化直方图
3. 计算累计分布函数（CDF）
4. 查找映射表

代码如下：

按照上述步骤对每个通道进行色彩迁移的函数

```
function transformed_image = transfer_channel(source_image, target_image)
    % 计算源图像和目标图像的直方图
    source_hist = imhist(source_image);
    target_hist = imhist(target_image);

    % 归一化直方图
    source_hist = normalize(source_hist, 'range');
    target_hist = normalize(target_hist, 'range');

    % 计算CDF
    source_cdf = cumsum(source_hist);
    target_cdf = cumsum(target_hist);

    % 构造映射
    mapping_function = zeros(256, 1);
    for i = 1:256
        [~, index] = min(abs(source_cdf(i) - target_cdf)); % 找到最接近的那个
        mapping_function(i) = index - 1;
    end

    % 应用映射
    transformed_image = mapping_function(source_image + 1);
end
```

对 RGB 每个通道都进行色彩迁移

```
function transformed_image = transfer_color(source_image, target_image)
    transformed_image = zeros(size(source_image));
    for channel = 1:3
        source_channel = source_image(:, :, channel);
        target_channel = target_image(:, :, channel);

        transformed_channel = transfer_channel(source_channel, target_channel);
        transformed_image(:, :, channel) = transformed_channel;
    end
end
```

对 source_image 应用这个函数

```
source_image = imread('food.jpeg');
target_image = imread('football.jpg');

target_image = imresize(target_image, [size(source_image, 1), size(source_image, 2)]);

transformed_image = transfer_color(source_image, target_image);
transformed_image=transformed_image/255; % 记得除255不然很白
imwrite(transformed_image, 'result41.jpg');
```

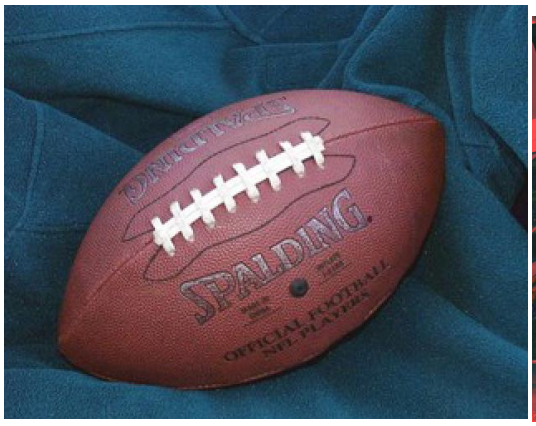
结果：

原图

色彩迁移后







效果还可以。
可以应用于风格转换和图像矫正等场景中。