山东大学 泰山学堂 学院

数字图像处理 课程实验报告

学号：201500150146	姓名：晁大任	班级： 泰山学堂 2015 级计算机
实验题目：OpenCV 配置及图像基本操作		
实验学时：5	实验日期：2017/10/12	

实验内容：

实验 3.1：直方图均衡化

实现图像的直方图均衡化算法，可以处理 8 位、任意通道数的图像

先统计出每一个灰度值的个数，得到矩阵 H

```
void calc_hist(uchar *data, int width, int height, int step, int H[256])
{
    memset(H, 0, sizeof(H[0]) * 256);
    uchar *row = data;
    for (int yi = 0; yi<height; ++yi, row += step)
        for (int xi = 0; xi<width; ++xi)
            H[row[xi]]++;
}
```

之后通过变换函数得到对应的 S 矩阵，变换函数如下：

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

得到的 S 矩阵对应的函数就是原来灰度值对应的新灰度值，最后把图像的每一个像素点的灰度值进行 S 矩阵对应的函数变换得到新的灰度值赋给像素点

对于多通道的图像，使其每一个通道分别进行上述的变换

我进行了多组实验进行验证，分别是两幅单通道灰度图（一幅太暗一幅太亮）和一幅三通道图，见附件 1 和附件 2

实验 3.2 快速连通域

实现图像的快速连通域算法，可以提取出图像中的连通域，并将不同连通域用不同颜色显示

快速连通域算法本质就是一次扫描得到哪些像素点在一个连通域内，然后合并等价类

在扫描的过程中，扫描到每一个像素点的时候，查看它的前一个像素点以及其上边相邻的像素点，与它本身分别记为 up、left、center。分四种情况：

a 如果 up≠center≠left，给目前像素点一个新的值

b 如果 up=center≠left，将目标像素点的值赋 up 的值

c 如果 up≠center=left，将目标像素点的值赋 left 的值

d 如果 up=center=left，将目标像素点的值赋 up 或者 left 的值

在此需要注意如果 up=center≠left，但 up 与 left 的值不相同，那么记录两个值都指向同一种颜色

同时需要注意没有 up 或者 left 的情形

需要说明的是，找到连通域需要扫描一遍整个矩阵，但扫描过程中不能给图片上色，因此需要再扫描一遍给图片加上颜色
我实现合并等价类的方法是用数组模拟一棵树，每次比较树的树根是否相同即可，合并和查询根的时间复杂度都近似为 O(1)
最终得到的结果在附件 3

硬件环境：
Intel(R) Core(TM) i5-5200U 8GB 64 位操作系统

软件环境：
Windows10 C++语言环境 VS2015
Project 类型：WIN32 控制台应用程序

实验过程中遇到和解决的问题：

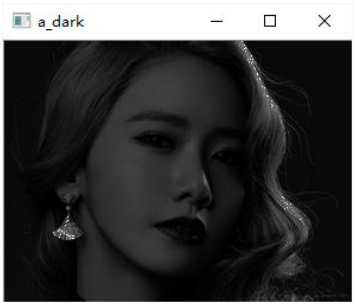
1 首先我不太清楚对于多通道图像应该用什么样的方法进行直方图均值化，有两种方法可以选择，第一种是对每一个通道分别进行均值化，第二种是找到对应的三维矩阵进行均值化。我尝试了第二种方法，这需要建造一个三维矩阵记录每一种 RGB 的密度，但我没有找到很好的计算方法，即不知道如何计算公式中对小于所在点灰度值求和这一步骤（在三维下求和应该是求一个长方体中所有点的和）

2 我在对图片进行双重循环修改每一个点中通道对应的值的时候，总是忘记 n 通道图像每一行对应的像素点是 cols*n

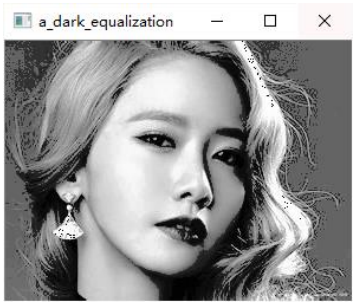
3 在实现快速连通域算法的过程中有很多小错误，比如计算上相邻像素时找错坐标等等。一开始我想使用哈希的方法，先用一个数组记录等价像素，如果左像素等价于上像素但两者不在一个等价类中就将这两个等价类哈希到同一种颜色中，但实际操作中发现需要的颜色太多，于是最后我还是实现了合并等价类方法

结论分析与体会：
通过对直方图均衡化以及快速连通域算法的实现，让我对相关知识有了初步的了解，使我对图像处理的算法充满了好奇与热情，同时锻炼了实践能力。
我发现一开始提出的方法和理论在具体情况下不一定可行，只有实践才能检验理论的可行性。
我相信，在老师的辛勤教导以及我的努力学习下，我一定可以出色的完成这门课并掌握好课程教授内容。

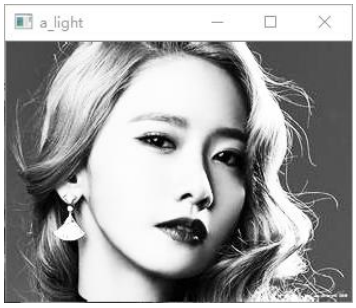
附件 1：



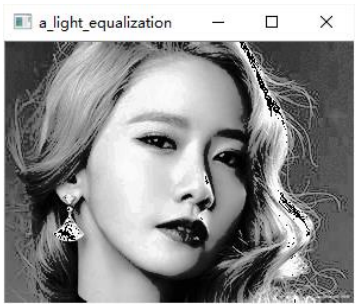
原始图像(太暗)



调整后图像

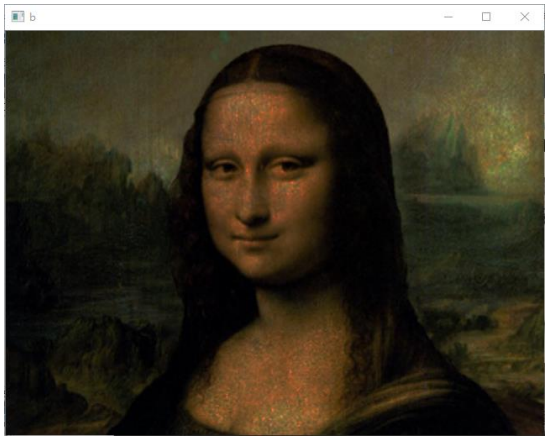


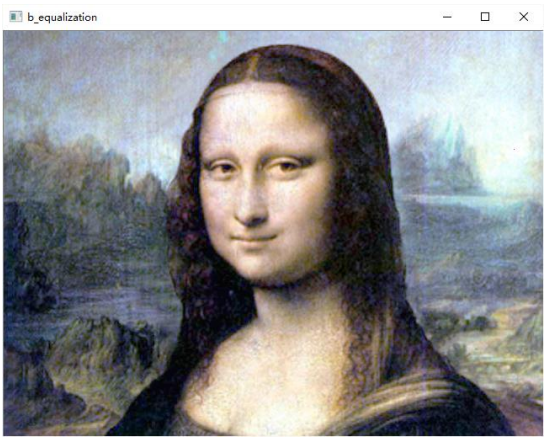
原始图像(太亮)



调整后图像

附件 2：





附件 3：