计算机视觉第二次作业

16340219 王亮岛

作业说明：

我在尝试了作业 2 的 code0 后发现自己不能得出很好的实验结果，最好的结果是对 lena.bmp 有较好的边缘检测结果，而其他 3 个实验数据得出的实验结果只能是 3 张毫无形状的结果图，在经过多次尝试后发现自己能力有限。因此我选择改写 code2 中的 canny 算法。

Code2：

测试环境：windows 下的 C++环境。

测试过程：

代码改写：

需要把 Mat 以及用 openCV 处理的函数和变量换成用 CImg 类处理的函数和变量即可，openCV 的数据处理过程和 CImg 的数据处理过程类似。要注意的是在进行双阈值处理的时候如果像素点的位置等于图片长度或宽度就视为出界。

算法分析：

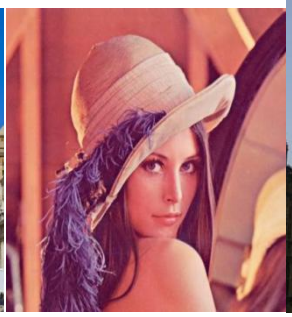
第一步：使用高斯滤波平滑图像，这里要用到二维高斯卷积核，因此要先生成卷积核。

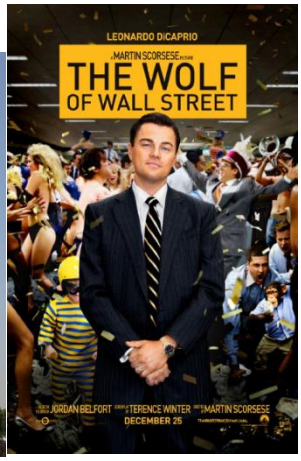
第二步：使用索贝尔算子将边缘返回到水平和垂直方向上，由此可以确定边缘的梯度和方向。

第三步：用非最大化抑制将边缘细化

第四步：用双阈值滤除具有弱边缘梯度的像素点，去除噪声。

测试数据如下：

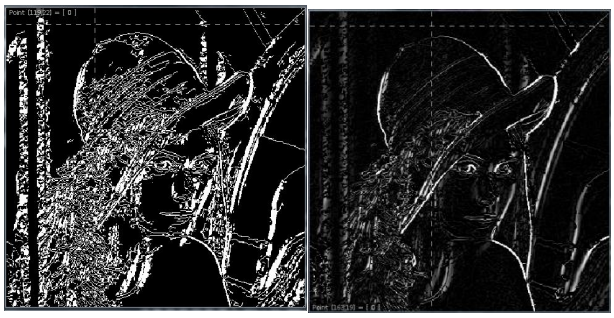


测试结果：从左到右，取 sigma 值为 1，高阈值为 40，低阈值为 20.

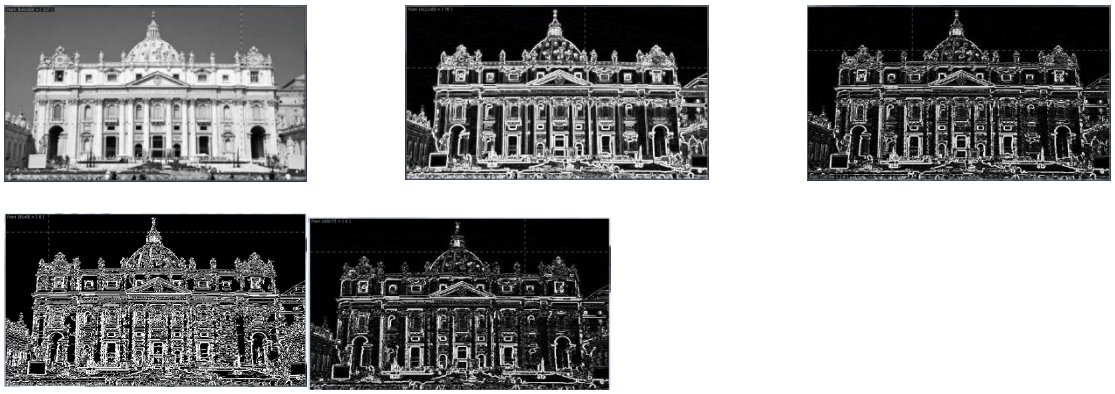
高斯平滑图像，用索贝尔算子过滤，非最大化抑制，双阈值处理，边缘连接：

对于 lena.bmp：

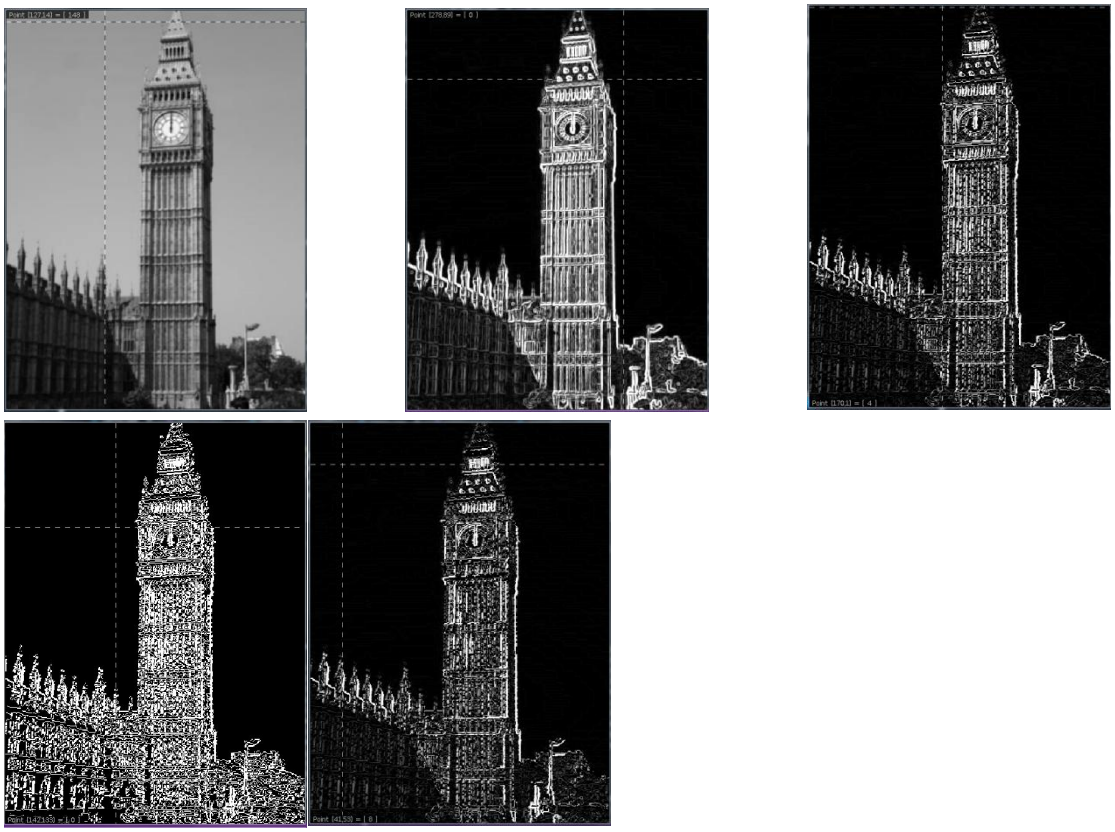




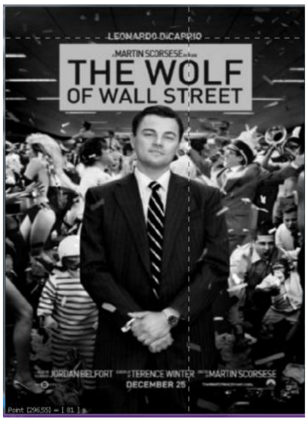
对于 stpietro.bmp：

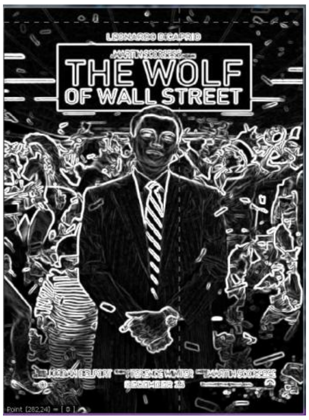


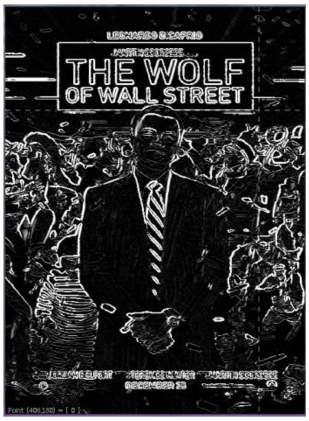
对于 bigben.bmp：

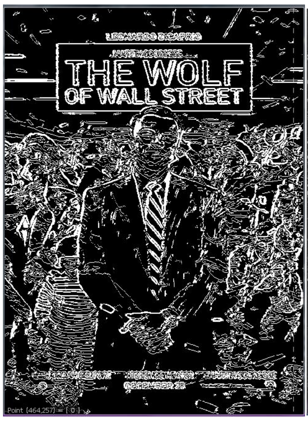


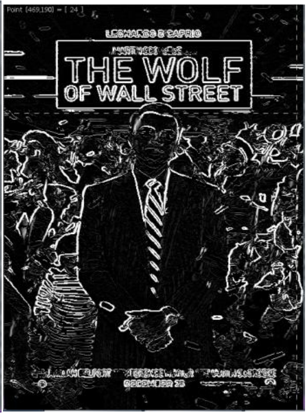
对于 twows.bmp：





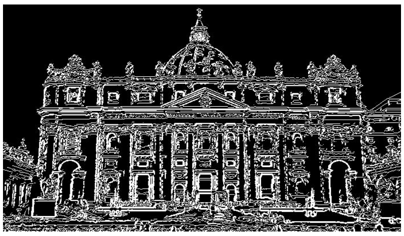


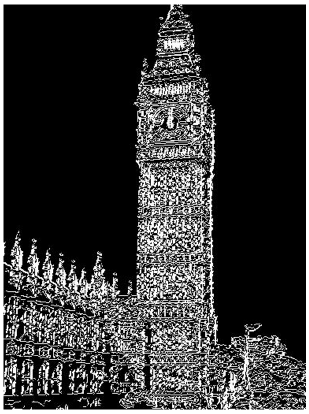
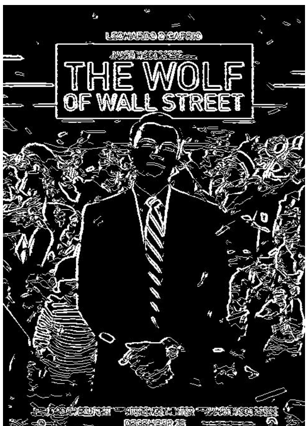




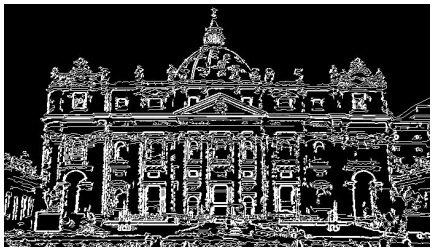
经过多次测试后一些较好的测试结果：

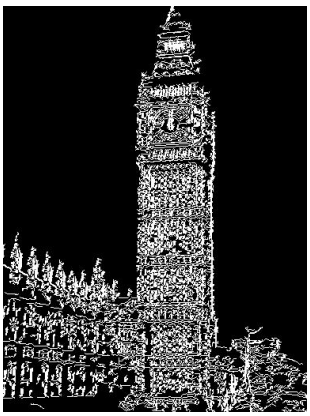
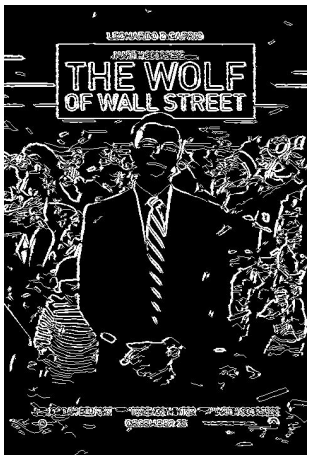
经过双阈值处理后：



经过连接边缘并去掉长度小于 20 的线条后：



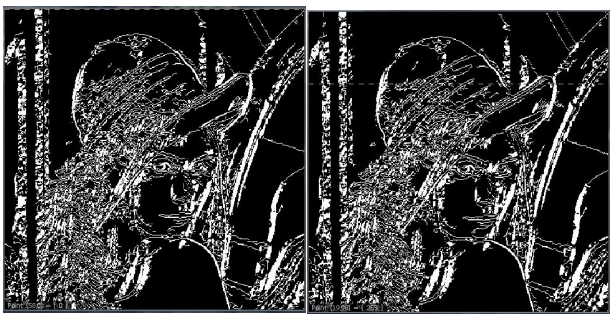
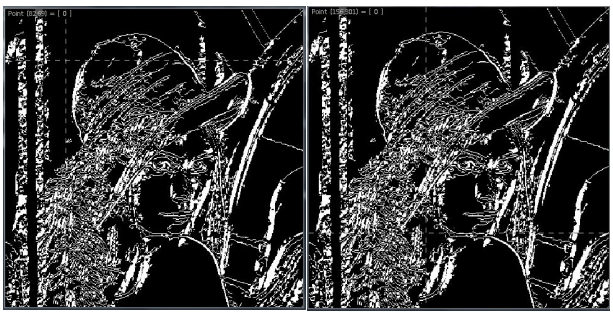



可以看出经过边缘连接后图像的轮廓更为清晰。由于噪声的影响减少了。这里我先看一下图片的轮廓与颜色的分布和变化，然后选取一个差较大的高低阈值，这样在连接边缘的时候会把一些噪声去除，同时将一些因为长度小于 20 的强边缘和周围的弱边缘连接形成更为完整的边缘。如果效果不好再缩小高低阈值之间的差值。

参数设置与分析：这里只比较双阈值处理后的图像。

此处以 lena.bmp 和 twows.bmp 为例：sigma 的取值为[1,5,50,100],低阈值的取值为[10,50,100,200],高阈值的取值为[80,120,160,200]。

对于 sigma：

可以看出 sigma 的取值对于图像的边缘提取影响不大。原因在于 sigma 对于高斯卷积核的影响在两部分，指数部分 sigma 增大与分母部分 sigma 的增大效果相抵消了，因此对卷积核的值没有很大影响，在图片上不能得到清晰的显示。但 sigma 的值不能小于 1。

对于低阈值：这里保证低阈值与高阈值的差为 30.

可以看出随着低阈值的增大，一些强边缘的像素会丢失，但阈值过小又会导致噪声的增加，使本可以剔除的像素点成为边缘像素点。

对于高阈值：这里选取低阈值为 10。Sigma 为 5.

可以看出高阈值的减小会增加一些强边缘像素，而高阈值的增加会将部分的强边缘像素点转换成弱边缘像素，因而丢失一部分边缘像素点。

边缘连接算法设计：

思路：主要实现方式为使用深度搜索算法。先找到一个弱边缘像素点，然后在其 8 邻域内找到其他弱像素点，进一步根据这些弱像素点及其 8 邻域找到更多弱像素点直到没有办法在一个弱像素点的 8 邻域内找到一个弱像素点。这里要注意的是如果这些弱像素点及其 8 邻域内存在强像素点，即可将这些弱像素点转化为强像素点。否则取消对这些弱像素点的标记，这样可以便于判断这些弱像素点中是否可以成为另一条边缘。

具体步骤：

1. 遍历每个像素点找到弱像素点
2. 找到弱像素点的 8 邻域
3. 在 8 邻域中查找是否有强边缘像素点同时记录所有弱边缘像素点
4. 若有强边缘像素点则弱边缘像素点=强边缘像素点，否则取消标记
5. 获取能通过 8 邻域找到的强像素点的个数，判断是否小于 20，若是，则变为非边缘像素点。