

计算机视觉 课程实验报告

学号：201822130233	姓名：李云龙
-----------------	--------

实验题目：图像匹配 1

实验过程中遇到和解决的问题：

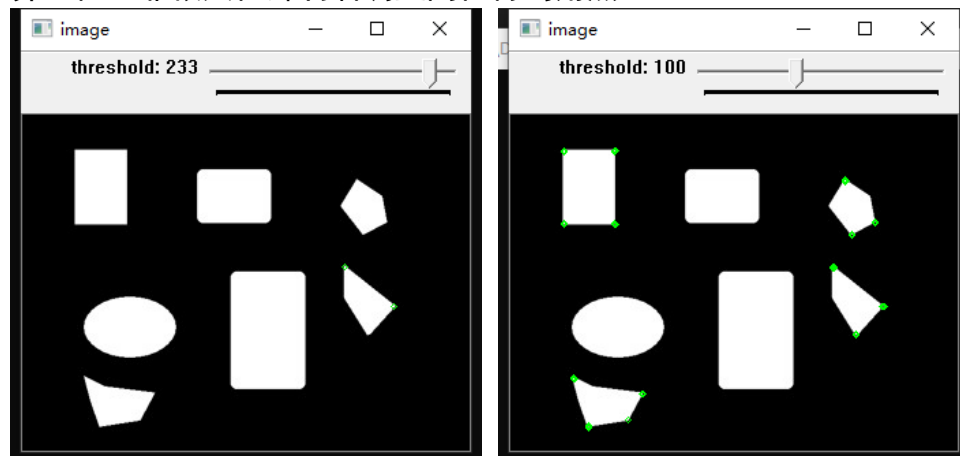
（记录实验过程中遇到的问题，以及解决过程和实验结果。可以适当配以关键代码辅助说明，但不要大段贴代码。）

实现 Harris 角点检测算法，并与 OpenCV 的 cornerHarris 函数的结果进行比较。

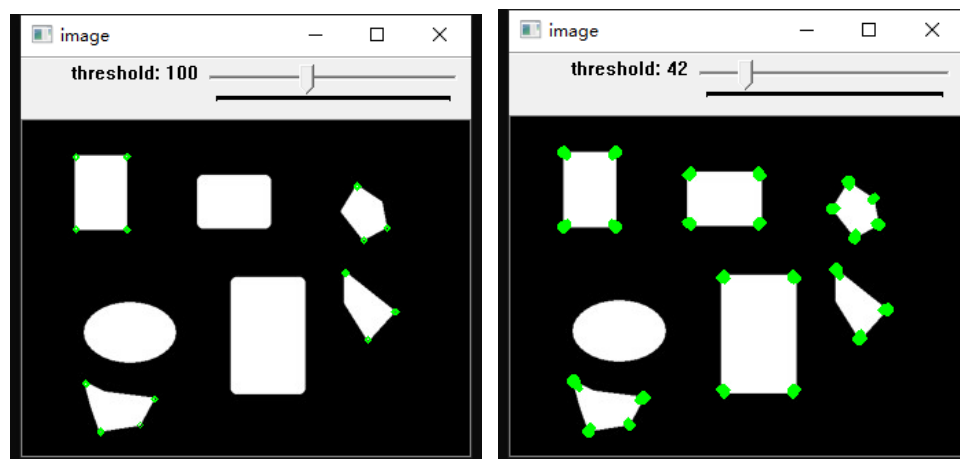
运用 Sobel 算子计算 x 与 y 方向的梯度得到矩阵 I_x 和 I_y ，定义窗口大小

$wsize$ ，在这个窗口内计算 M ，
$$M = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix}$$
，即分别计算 $I_x * I_x$ ， $I_y * I_y$ ，

$I_x * I_y$ 在窗口内的取值并求和。根据 $R = \det(M) - \alpha \text{trace}(M)^2 = \lambda_1 \lambda_2 - \alpha (\lambda_1 + \lambda_2)^2$ 计算 r 值。非极大值抑制后调整阈值得到角点。



如果 $w(x, y)$ 使用高斯函数，先把 I_{xx} ， I_{yy} ， I_{xy} 矩阵用 GaussianBlur 处理再计算 R 。



OpenCV 函数 `void cornerHarris(InputArray src, OutputArray dst, int blockSize, int ksize, double k, int borderType=BORDER_DEFAULT)`

src: 输入图像，即源图像，填 Mat 类的对象即可，且需为单通道 8 位或者浮点型图像

dst: 用于存放 Harris 角点检测的输出结果，和源图片有一样的尺寸和类型

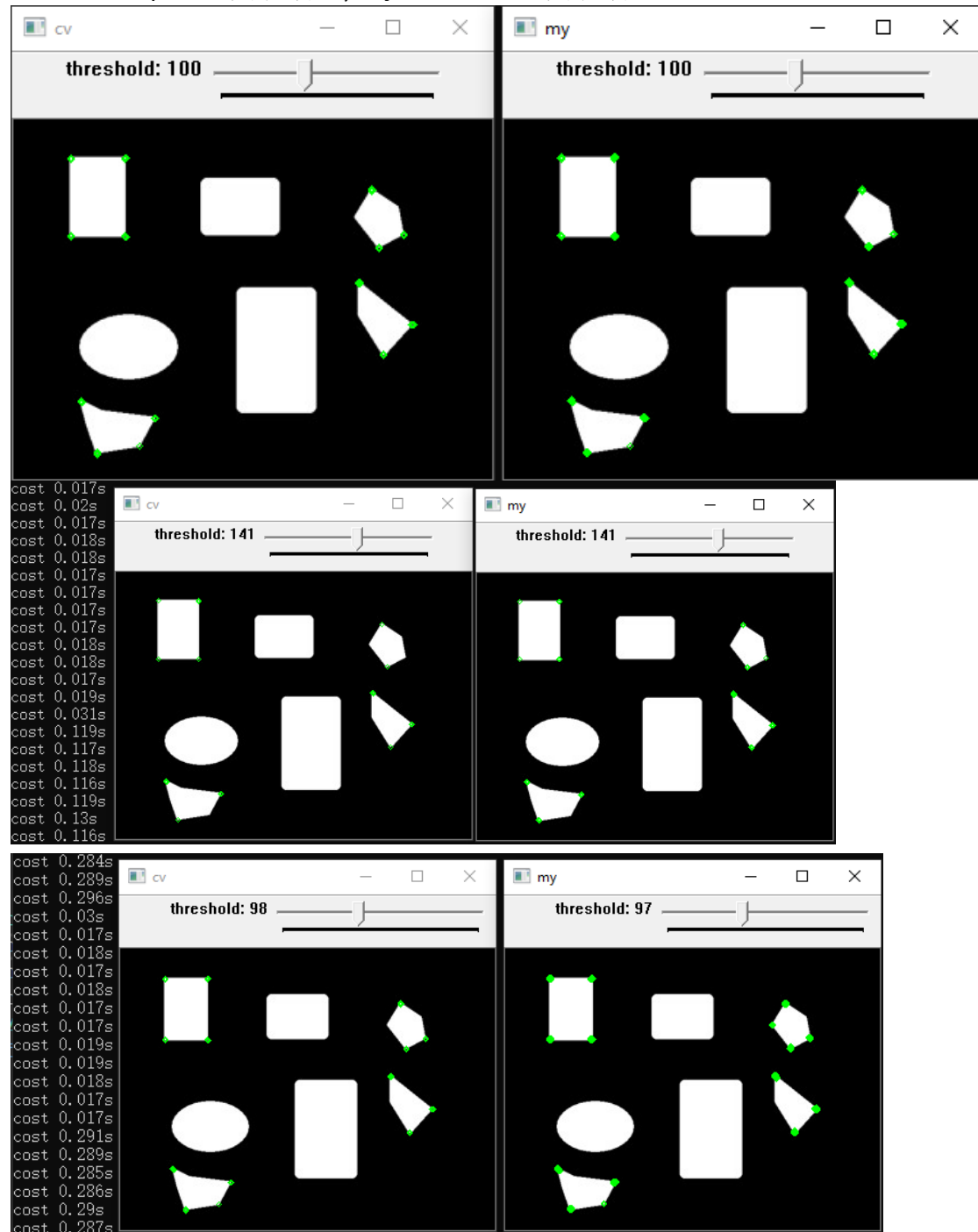
blockSize: 表示邻域的大小

ksize: 表示 Sobel() 算子的孔径大小

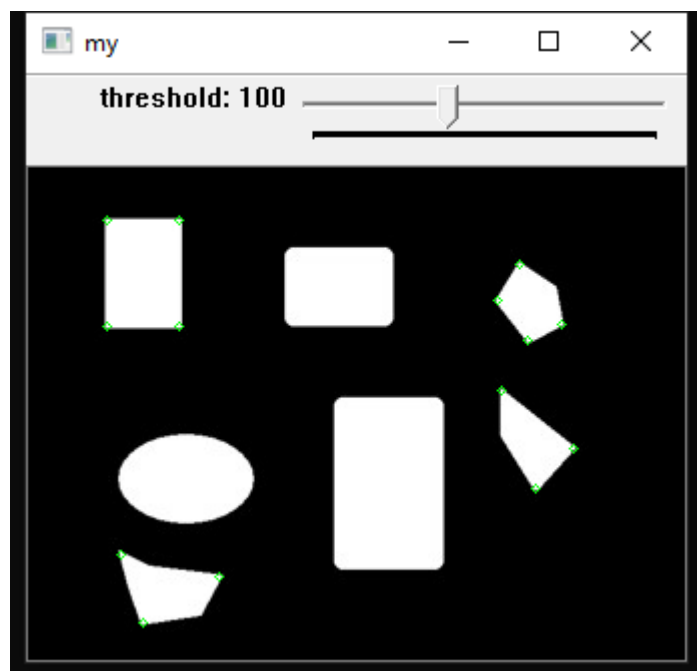
k: Harris 计算公式中的参数

borderType: 图像像素的边界模式，有默认值 BORDER_DEFAULT

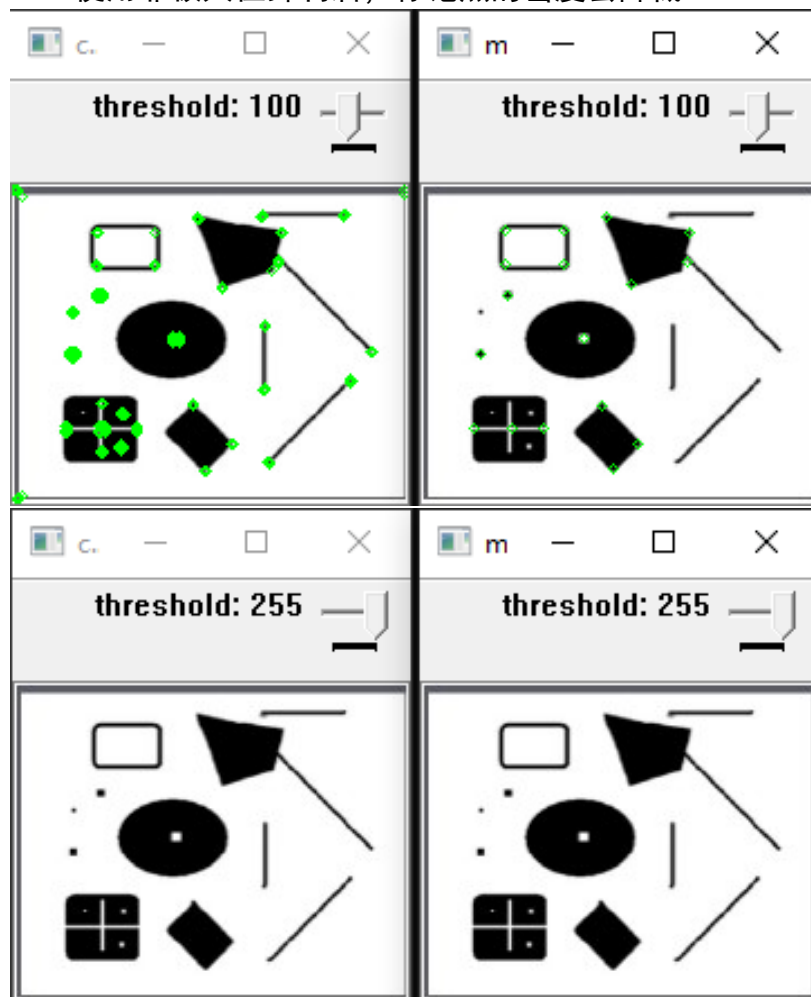
cv 窗口为 OpenCV 实现效果，my 窗口为自己实现效果

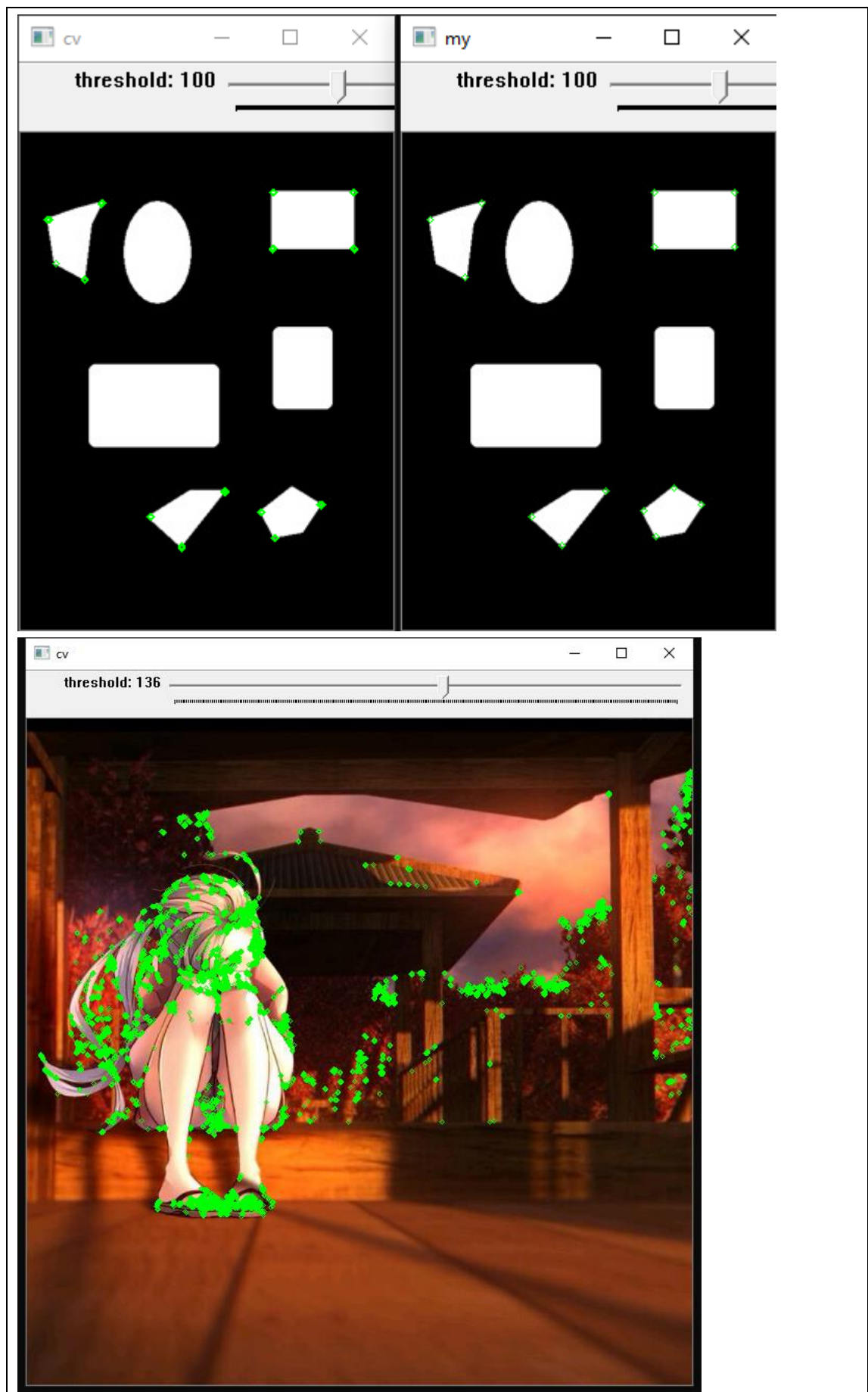


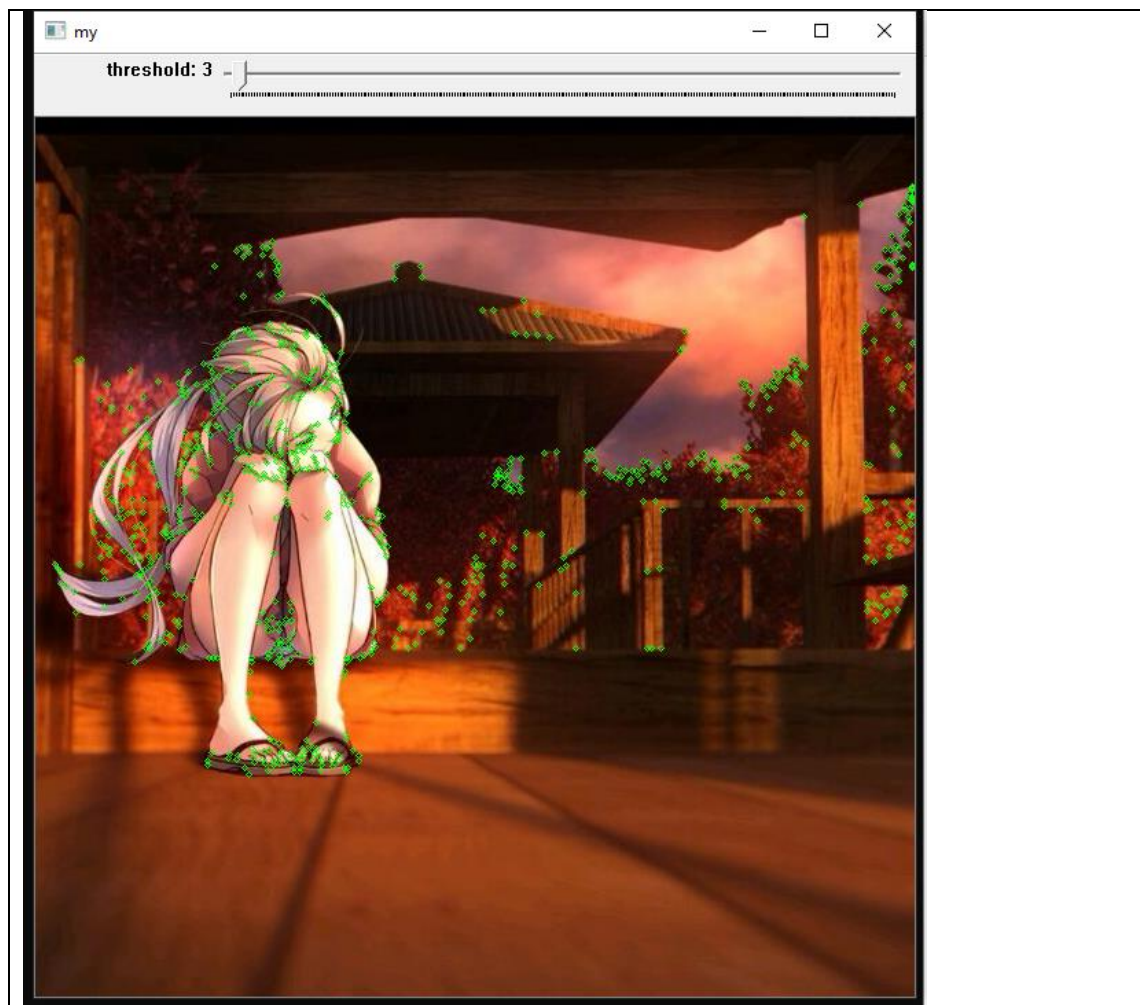
OpenCV 函数的时间消耗在 0.018s 左右,自己实现的函数时间消耗在 0.118s 左右,如果使用 Gauss 则时间在 0.28s 左右。



使用非极大值抑制后,标记点的密度会降低。







结果分析与体会：

本次实验中，实现了 Harris 角点检测算法，并与 OpenCV 的 `cornerHarris` 函数的结果进行了比较。