

# 光栅图形学作业

翁家翌 2016011446

2018-03-20

## 目录

<b>1 基本选题</b>	<b>2</b>
1.1 画线 . . . . .	2
1.2 区域填充 . . . . .	3
<b>2 加分项</b>	<b>4</b>
2.1 反走样 . . . . .	4

# 1 基本选题

基本选题为区域填充，顺便把画线给实现了。编程语言使用的是 python3，使用的第三方库为 opencv。使用如下命令安装依赖包：

```
sudo pip3 install -r requirements.txt
```

所有代码位于 main.py 中，使用命令 `python3 main.py` 即可运行。

## 1.1 画线

实现画线的函数为 `line(p1,p2)` 和 `drawline(a,p1,p2,col)`，作用如下：

- `line(p1,p2)`：输入两个点坐标，默认  $|p_{1x} - p_{2x}| \geq |p_{1y} - p_{2y}|$ ，返回一个从  $p_1$  到  $p_2$  需要着色的点的 list
- `drawline(a,p1,p2,col)`：输入图像矩阵 a、线段起点终点和需要着色的颜色，在 a 中画出该线段

实现效果如图1所示。

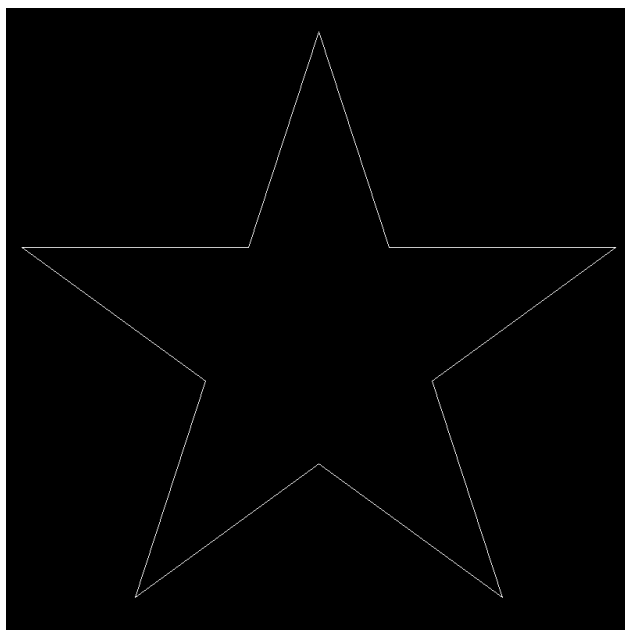


图 1: 一个五角星

可以看到放大之后的毛刺效果，如图2所示。



图 2: 放大之后的线段

## 1.2 区域填充

实现函数为 `colorize(a,p,bg,fg)`，具体为在矩阵  $a$  中，以点  $p$  为起始点，填充颜色  $fg$ ，默认去覆盖颜色  $bg$ 。使用队列实现，效果如图3所示。

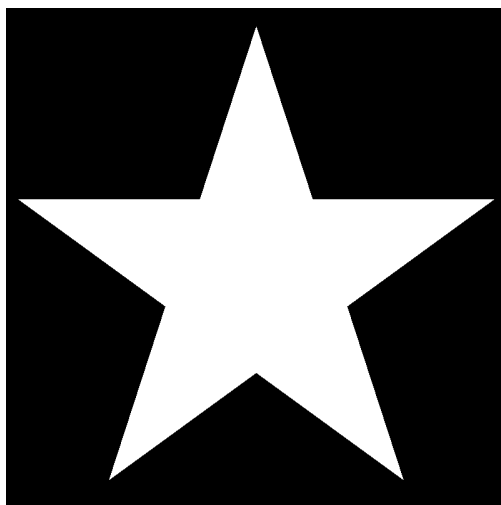


图 3: 一个白白的五角星

可以看到毛刺效果并没有消失，如图4所示。



图 4: 放大之后的边缘

## 2 加分项

此处实现了反走样功能

### 2.1 反走样

反走样使用卷积操作使图像平滑，其中卷积核为

$$\mathfrak{F} = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

我自己手写了一下卷积实现（代码中 91-95 行，已注释），经过测试，运行效率远不如 opencv 中的 `filter2D` 函数来得快。实现反走样之后的效果如图5所示。



图 5: 一个没有毛刺的五角星

可以看到边缘已经平滑，如图6所示。



图 6: 放大之后的边缘

出于美观角度，最后在背景中又人为地加入了一些噪点，能够拥有更好的视觉效果。最终成品如图7所示。

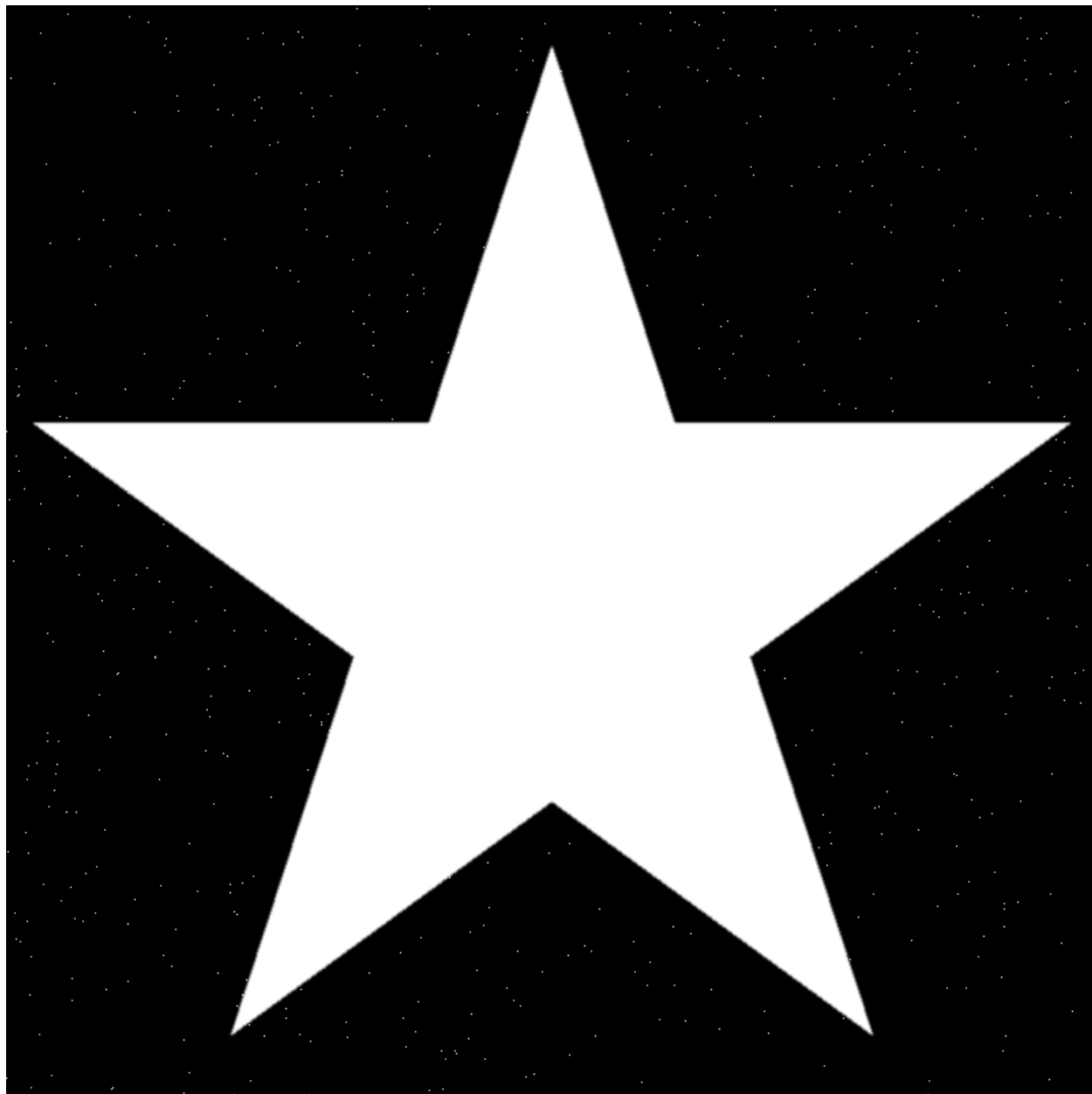


图 7: 一个有背景的五角星