计算机图形与图像 HW#2

陈宇轩 **PB16060738**

Question 1

问题描述

实现一个对于中点算法的反走样程序。

算法分析

先讨论在$0 \leq m \leq 1$的情况，参考书本P143：

1. 设直线方程为$y = mx + b$
2. 由于假定了$0 \leq |m| \leq 1$，因此，每次在x方向上加1，y方向上加1或不变需要判断。
3. $y - y_{mid} = [m(x_k + 1) + b] - (y_k + 0.5)$

   通过加上$1 - m$来调整这个计算，得$p = mx_k + b - y_k + 0.5$

   假如p<1-m,则取y_k上的像素，否则取$y_k + 1$上的像素。
4. 同时，p也是直线在像素内部覆盖区域的面积。

   因此，通过计算p值来确定直线的一下个位置，也可以确定对当前像素覆盖区域的百分比。
5. 综上得：

   当$0 \leq m < 1$时, $p = mx_k + b - y_k + 0.5, y = \begin{cases} y & p < 1 - m \\ y + 1 & p \geq 1 - m \end{cases}$

   同理，可推导到其余斜率范围：

当$-1 \leq m < 0$ 时, $p = mx_k + b - y_k + 0.5, y = \begin{cases} y - 1 & p < -m \\ y & p \geq -m \end{cases}$

当$m > 1$ 时, $p = \frac{y_k - b}{m} - x_k + 0.5, x = \begin{cases} x + 1 & p < 1 - \frac{1}{m} \\ x & p \geq 1 - \frac{1}{m} \end{cases}$

当$m < -1$ 时, $p = \frac{y_k - b}{m} - x_k + 0.5, x = \begin{cases} x + 1 & p < 1 + \frac{1}{m} \\ x & p \geq 1 + \frac{1}{m} \end{cases}$

代码实现

使用glfw与OpenGL，编译环境为CLion。

具体代码见附件项目HW2。

对于上面推导得到的p值，我用来作为设置点时采用的alpha值，以达到模糊边缘的效果。

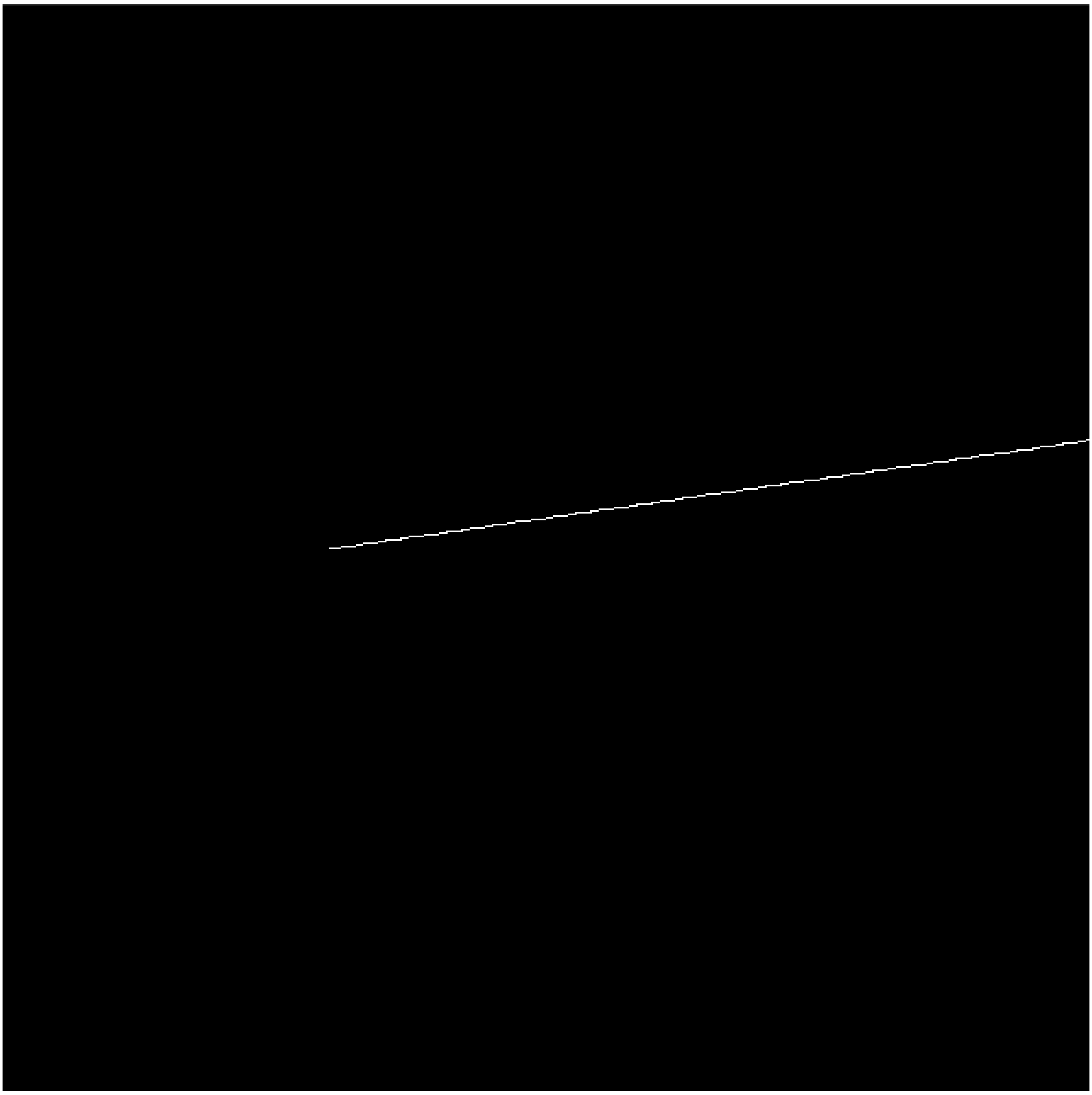
在实际编写过程中，我使用p来对斜率绝对值小于1的情况进行取点判断，得到的结果较好。

但是对于斜率绝对值大于1的情况，我发现画的图像不正确，根据DEBUG查看变量，发现始终$p < 1 - \frac{1}{m}$ (m>1)，导致x不停地取x+1，使得图像画图不正确，我推测可能是面积运算不能只取梯形，由于时间问题没有完全解决，最后对于斜率绝对值大于1的情况，我采用了第一次作图的d来进行取点，但是仍采用了p进行亮度的调整，进行反走样处理。反走样的结果良好。

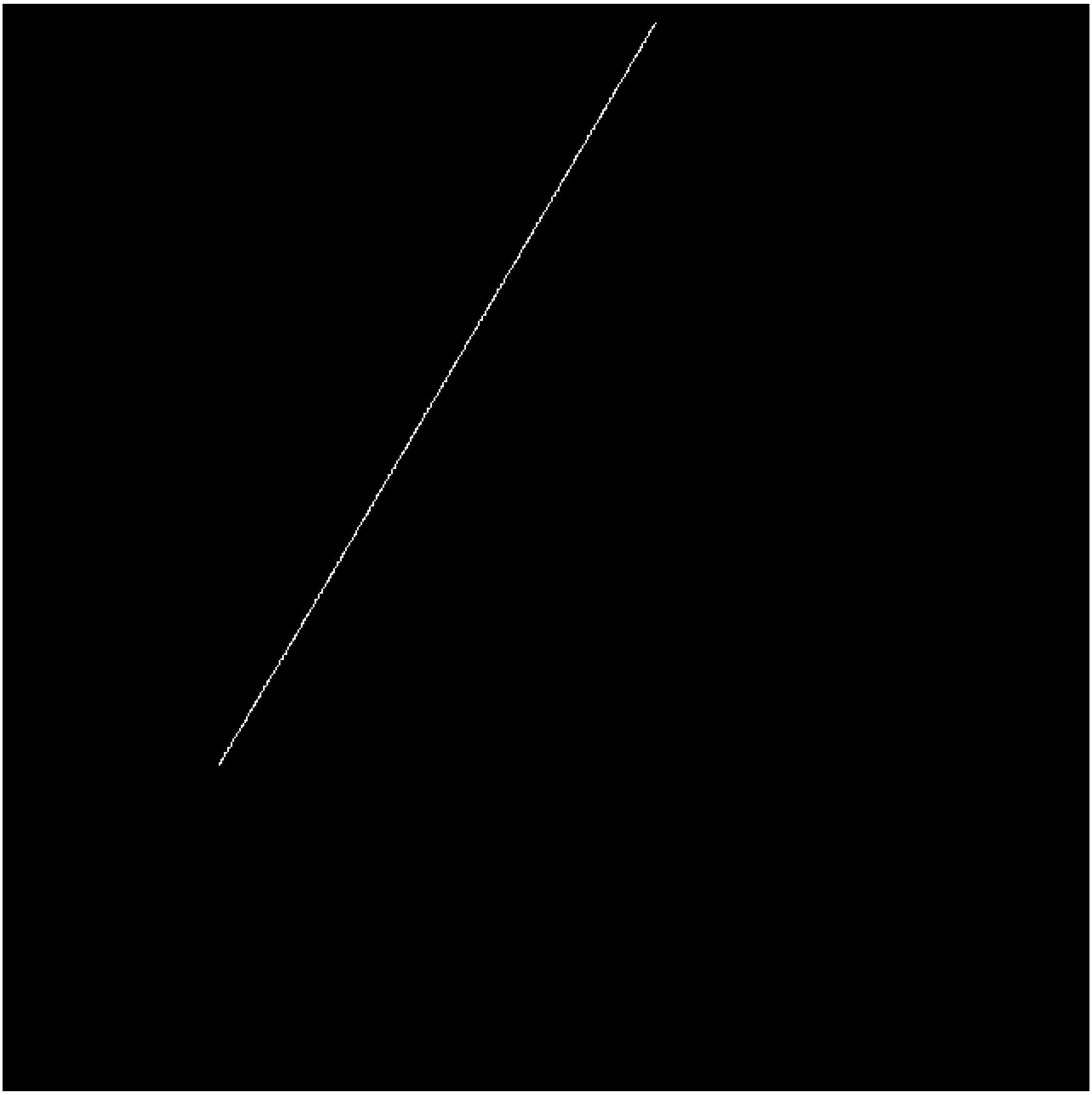
实验结果

在原本直线的周围，会有一层淡淡的白色，形成反走样效果。但我个人觉得效果有限，这种算法效果还是有限，如果想真正比较好的实现反走样，可以直接使用OpenGL里的高频采样函数进行处理。

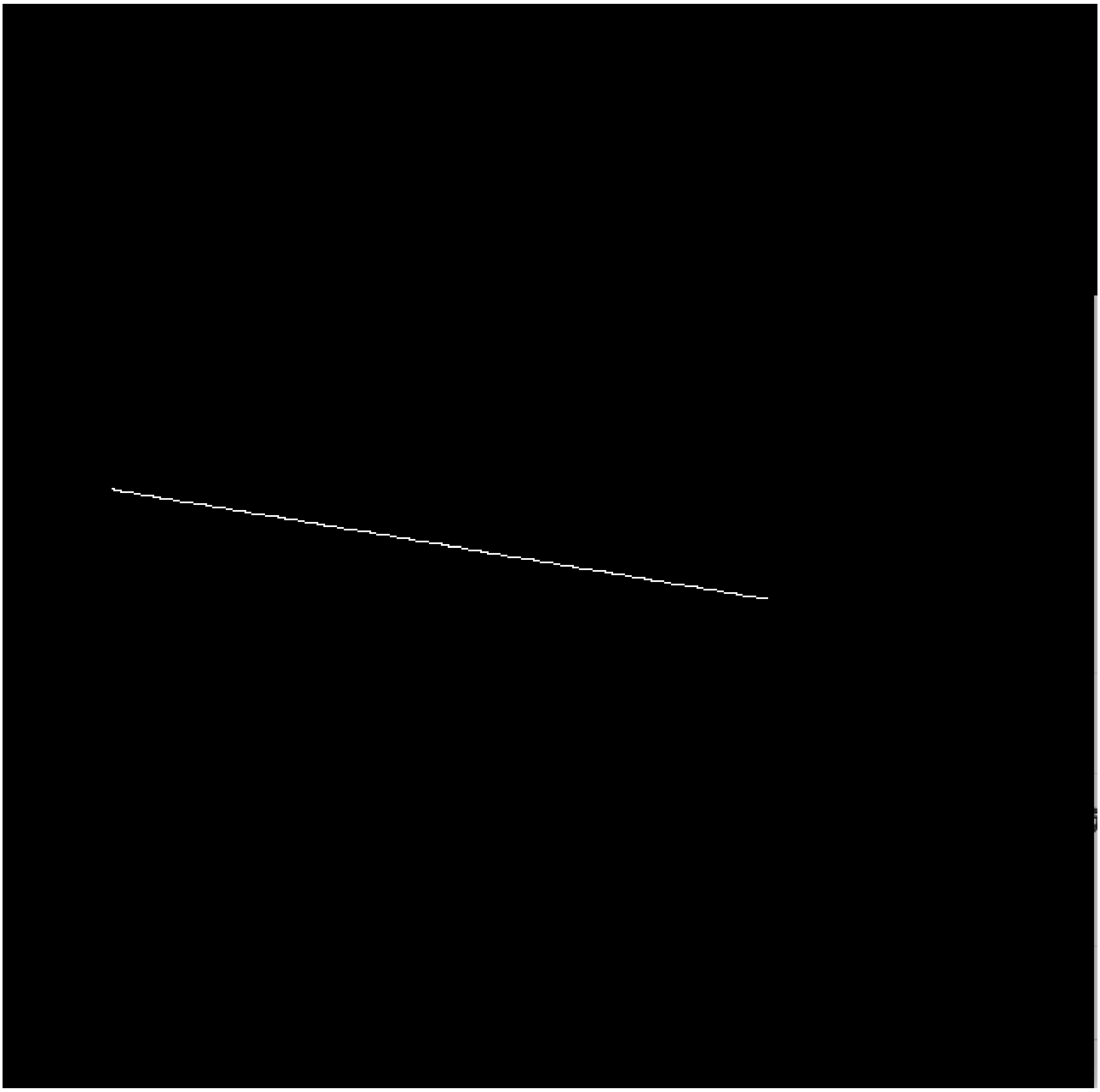
测试样例**1**：(-200,0) (500,100)



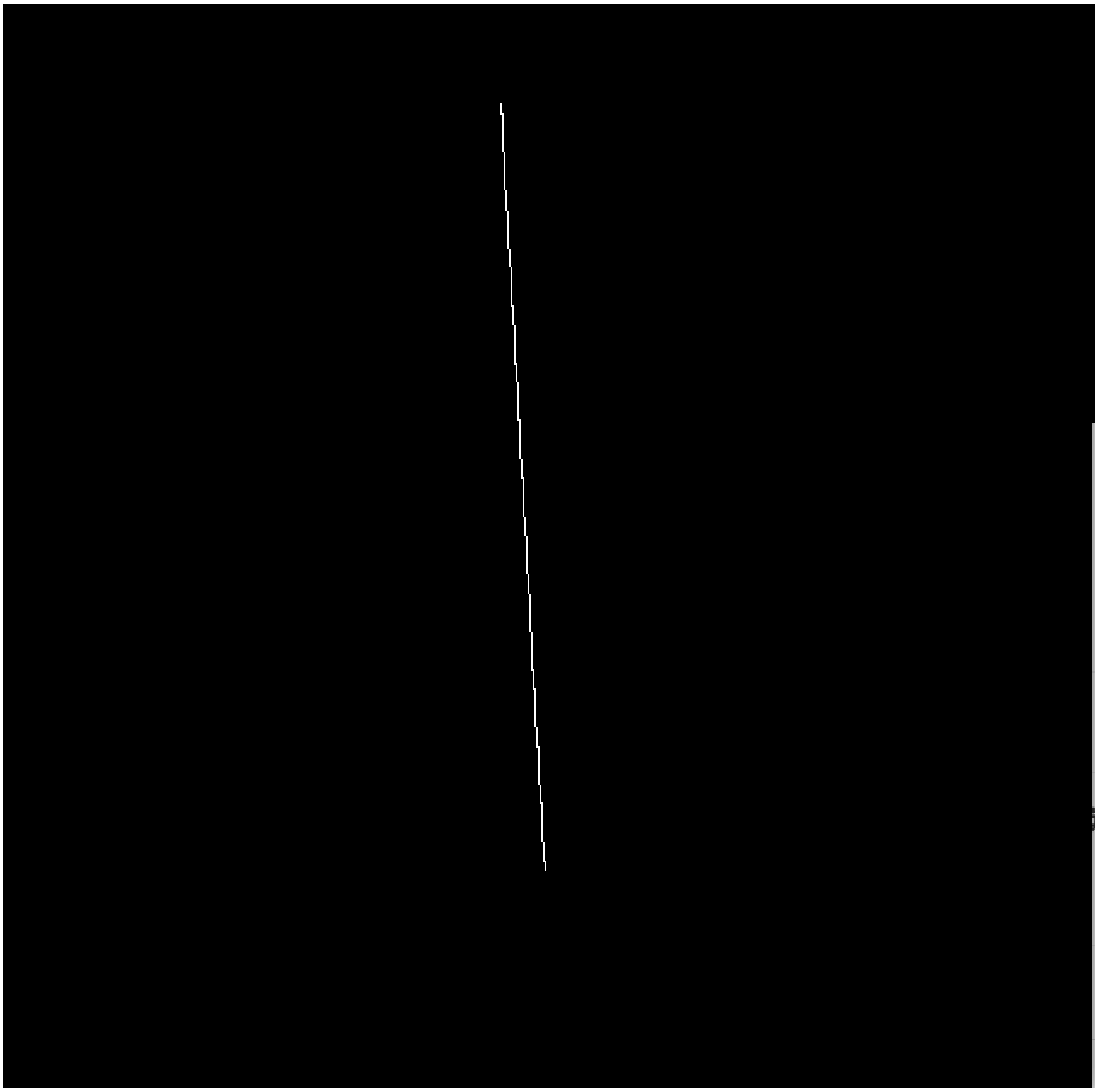
测试样例**2**：(-300,-200) (100,480)



测试样例**3**：(-400,50) (200,-50)



测试样例**4**：(-40,400) (0,-300)

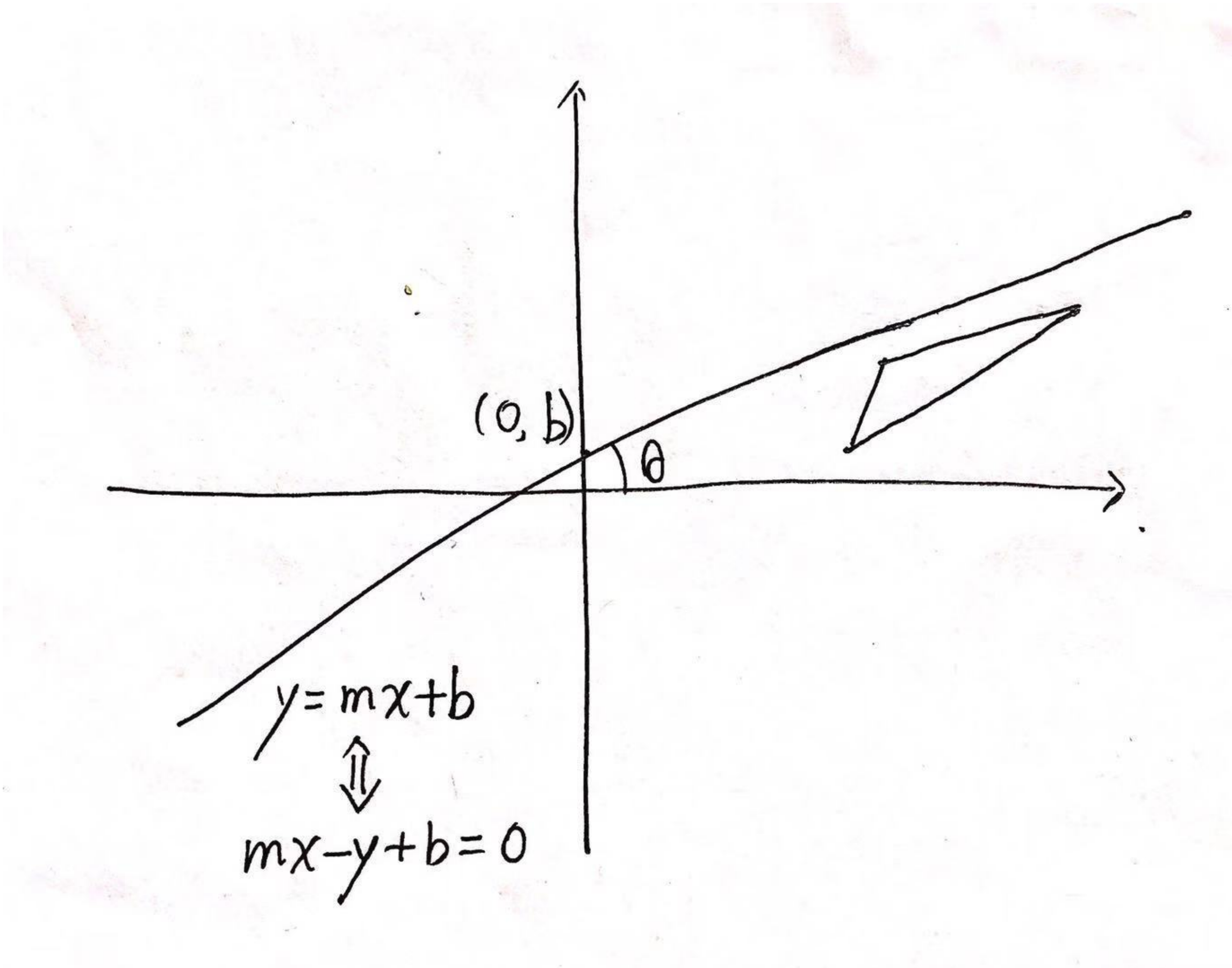


Question 2

问题描述

确定相对于任意直线**y=mx+b**的对称变换矩阵。

问题解答



设直线如图所示，要进行对称变换的图形如图所示。

1. 将直线沿x轴向右平移$\frac{b}{m}$，使其通过原点。
2. 使直线沿坐标原点旋转$-\theta$，使其与x轴重合。
3. 求图形关于x轴的对称变换。
4. 求图形绕原点旋转θ的旋转变换。
5. 求图形向左平移$\frac{b}{m}$的平移变换。
6. 综上，通过三种变换，可得图形关于直线的对称变换矩阵为

$$\begin{bmatrix} \cos 2\theta & \sin 2\theta & \frac{6}{m}(\cos 2\theta - 1) \\ \sin 2\theta & -\cos 2\theta & \frac{6}{m}\sin 2\theta \\ 0 & 0 & 1 \end{bmatrix}$$