

程序分析文档

程序总体逻辑

预处理

1. 使用pillow库，读取指定数量的图片
2. 对于每一个读取到的图像，将其转化为灰度图像，并对大小进行统一（500*333）
3. 将大小调整后的图片存储在temp文件夹内

绘制灰度直方图

1. 对于给定的图片，先统计其每一个灰度值的数量
2. 使用matplotlib进行柱状图的绘制

直方图均衡函数

1. 统计图片中每一个灰度值的数量
2. 计算灰度值累加数组
3. 根据累加的灰度直方图得到输入灰度级和输出灰度级之间的映射关系
4. 得到直方图均衡化后的图像

基于点处理的增强

遍历图像矩阵，对每一个点的灰度值做映射

基于邻域处理的增强

1. 除了最值、中值滤波，全部使用卷积变换
2. 卷积过程：
 1. 根据卷积核的大小对图片做填充，周围全部填充0
 2. 遍历图片对每一个子图片块做卷积
3. 根据输入的不同卷积核进行不同操作

开发中遇到的困难

- 灰度值溢出：有可能经过映射后的灰度值大于255，程序默认的处理方式是将其对256取模，这样处理不符合映射的目的。因此应该将大于255的灰度值规范至255
- 均值滤波后的结果可能是float类型，而灰度统计函数不支持float，故需要对均值滤波的结果做向下取整
- plot问题：当图像中某一灰度值的数量远远大于其余灰度值的数量时（如图像是黑色背景则灰度值为0的数量将极大），绘制出的灰度直方图只能观测到最大的一条竖线，而对于其余的灰度无法准确观测。处理方法是，当最大值远大于次最大值时，将最大值以一定比例缩小