



高级语言程序设计(进阶)

2022-2023 学年第二学期项目作业展示

姓名: 闫浩扬 学号: 2253156 邮箱: 2253156@tongji.edu.cn



国豪工科精英班

展示项目: 矩阵运算及应用

项目介绍

该项目是一个矩阵运算和图像处理的小程序，整合了矩阵的基本操作，包括加法、数乘、转置、乘法、Hadamard乘积和卷积等。此外，项目还利用矩阵卷积进行了基础的图像处理。加分项部分实现了OTSU算法对图像进行二值化，并利用该算法提取出图像的主体部分。

设计思路

首先将完整功能分割为以下几个部分，并按照该顺序进行

(1) 矩阵的构造

受前期作业启发，决定封装一个矩阵类，其中包括矩阵的数据以及一些矩阵运算需要使用的必要函数等。提高了可读性，安全性以及可维护性。

(2) 矩阵的基础运算

重载 $+$, $*$, $-$, $/$, $+=$, $-=$, $==$, $!=$ 等运算符，一是为了巩固所学知识，刻意使用，二是简化代码，直观易维护。同时对矩阵运算的各种条件进行一定的判断，以处理运行的各种情况。

(3) 矩阵的进阶运算

矩阵的卷积：首先理解各种参数的作用，以及卷积的流程。然后按照流程对进行运算。

(4) 图像的卷积

①准备输入矩阵（将图像转化为灰度图像并储存在矩阵中）和卷积核（滤波器）②图像填充（补边）避免在卷积过程中出现边界像素的丢失。③开始卷积，将图像矩阵与卷积核进行卷积操作，并将结果矩阵转化为灰度图像进行输出。同时运用了多个类型转化的自定义函数实现各类型之间的互相转换。

(5) OTSU算法的二值化

OTSU算法基本思路：①计算灰度直方图②计算总像素数以及各个灰度的权重③计算类内方差以及最佳阈值并实时更新④根据阈值分割图像，进行二值化④输出根据最佳阈值二值化后的图像

(6) 利用OTSU算法提取主体

观察后发现与二值化存在细小区别。提取主体要求保留原有灰度，并将背景变为纯黑色即可。

成果展示



实现难点及解决方案

(1) 矩阵数据结构的选择不合适

前几项矩阵都为 10×10 以内，所以最初使用了静态数组的形式，但是到图片处理时，由于图片大小较大，矩阵大小变化大，因此考虑使用动态内存申请的方法，动态为矩阵分配所占用的内存，极大地节省了空间，灵活度高。

(2) 矩阵卷积有误

矩阵卷积时，为了对边缘像素也进行处理，应当进行填充，使用填充矩阵进行卷积操作，保证对边缘像素也进行处理。

(3) 图片比例不对

矩阵的输出图片比例太大导致显示不全，思考发现是`mat.at<>()`成员函数赋值时将`j`变为`3*j`。

(4) 灰度图像太锐利，颜色有问题

灰度值判断出现错误，导致每个像素的灰度出现问题。在灰度赋值时添加判断大于 255 值置 255，小于 0 置 0；

(5) OTSU算法计算阈值错误

最佳阈值的计算应当同时考虑类内方差和类间方差，遍历所有可能的阈值 T ，计算每个阈值的类间方差，并选择使类间方差最大化的阈值作为最佳阈值。

(6) imshow显示多个图像时有些在边缘

如果图像的大小超过了屏幕的可见区域，可能会出现部分图像显示在屏幕边缘的情况。这是因为`imshow`默认将图像显示在屏幕左上角的位置，并根据图像大小调整显示窗口的大小。

实现心得

由于基础部分前期作业已经写过，所以写大作业时较为轻松，直接使用矩阵类，运算符重载等，耗时少。在进行卷积时，由于没有接触过此类概念，上网搜索自主学习的过程了解了许多有趣的知识，虽然遇到许多问题，例如未填充边界等，但是在不断搜索尝试的过程中对矩阵卷积操作的认识也更为深刻。并且本次作业并没有知识数字的枯燥乏味，卷积应用使得成果肉眼可见，OTSU算法进行图像的二值化处理以及分离主体也让我感到成就感满满，既了解到图像处理的基础知识，又在不断尝试中学到了编程技巧，提升了自己的能力。