

狼牙战队2024赛季算法组考题整体说明

笔记本: My Notebook

创建时间: 2023/9/29 11:43

更新时间: 2023/9/29 11:50

作者: Miracle_ngu

目录

1. 初阶任务一: 排序
 2. 进阶任务一: 矩阵类
 3. 进阶任务二: 利用自制矩阵类求坐标变换
 4. 进阶任务三: OpenCV的安装与轮廓检测
 5. 进阶任务四: 灯条识别
 - armor.png 图片识别
 - 1b2r_armors.mp4 影片识别
-

初阶任务一: 排序

程序: Sorts.cpp

简述:

对数据进行排序的方法有很多种, 我选做了基础等级的冒泡排序和插入排序, 以及稍微进阶的快速排序作为本次考题的开胃前菜, 它们各自的内存需求及时间复杂度不多叙述。

设计思路: 冒泡排序, 字如其名, 由首位元素依次与后一位元素比较大小, 大数往后移, 小数往前移, 犹如泡泡在水里一般缓慢上浮, 时间复杂度最大。

插入排序, 半字如其名, 思路为划定已排序好的区域, 要求区域内元素按小到大排序, 首先把第一位元素划入到此区, 随后由第二位元素开始与前述区域内元素逐个比较大小, 从而找到适合自己的位置并插入其中。

快速排序, 设置两个哨兵 i, j 分别指向首元素和尾元素, 并另设置一缓冲区 key 用于储存元素, j 从右向左移动直到找到比 key 小的元素停下, 交换 j 和 key , i 从左往右移动直到找到比 key 大的元素停下, 交换 i 和 key , 一直执行上述操作直至 i 和 j 相遇, 此时交换 key 和 i 或 j , key 左方均是比 key 小的数, key 右方均是比 key 大的数, 然后分别对前

半段和后半段进行上述排序便可。

注意事项请详阅相关 ReadMe.pdf 说明文档。

进阶任务一：矩阵类

程序：Matrix.cpp

简述：

程序定义了一个类 Matrix, 私有成员包括行数 rows, 列数 cols, 以及 double 类型的二维指针 data 用以储存矩阵元素; 公共成员包括能动态申请内存的构造函数 Matrix() 设置矩阵元素的 setValue()、获取矩阵元素的 getValue(), 重载了 +、-、* 运算符, 实现矩阵的加法、减法及乘法运算, 此外还有进行矩阵求逆的 inversion() 函数, 以及其辅助-用于构建单位矩阵的 identity() 函数, 而通过 print() 函数能够按矩阵的形状输出其内容。

主函数中会提示用户输入两个初始矩阵的行、列数并赋值, 可按照相关任务文件夹中的 ReadMe 的测试样例输入。

注意事项请详阅相关 ReadMe.pdf 说明文档。

进阶任务二：利用自制矩阵类求坐标变换

程序：Vector_transform.cpp

简述：

通过对《视觉SLAM十四讲》的“三维空间刚体运动”一章的学习, 以及 B 站上相关影片的介绍令我对该章节有了基本的认识, 大致程序结构沿用上一任务点“进阶任务一：矩阵类”, 不同点在于把原本的手动输入矩阵行列元素改为自动设置为任务要求的大小及内容, 分别为矢量 a、旋转矩阵 R 以及平移矢量 t, 对三个矩阵进行运算前的预处理后按照“a 依次经过 R、t 变换”的顺序进行矩阵相乘、相加等运算得到结果, 对其进行转换得到最终结果并输出。

注意事项请详阅相关 ReadMe.pdf 说明文档。

进阶任务三：OpenCV的安装与轮廓检测

程序：Contour_detect.py

简述：

OpenCV的安装：

已于暑假安装了 vscode 并配置了 OpenCV (python及C++) 进行学习, 当中的流程重点大概为设置环境变量, 在 CMake 指定 gcc 及 g++ 路径, 并利用其生成 Makefiles 等文件配置, 及设置 .vscode 中的 c_cpp_properties.json 、 launch.json 和 task.json 三个文件。

轮廓检测：

想要侦测轮廓首先要把图像转为 HSV 图, 利用 Canny() 函数测轮廓, 此处为了便于观察过程, 便在原图的复制本 imgContour 上画出了原图中的所有轮廓并计算出轮廓面积和周长, 随后把所获轮廓进行 approxPolyDP 函数的近似轮廓处理 (其中需要进行调参), 分辨轮廓的顶点数量及位置坐标, 最后通过对轮廓顶点、面积和坐标进行限制得出二维码左上角 "双层正方形" 中内层的正方形, 并在原图 img 上用蓝色画出了其外轮廓并展示。

注意事项请详阅相关 ReadMe.pdf 说明文档。

进阶任务四：灯条识别

简述：

图片"armor.png"灯条识别：

程序：Light_recognition.cpp

在该程序中你需要在 TrackBar 视窗中的Mode栏滑动滑条进行模式选择, 当中有 "0", "1", "2" 分别对应 "使用你自己调整的HSV参数进行后续识别", "使用已调整好的红色HSV参数进行后续识别", 以及 "使用已调整好的蓝色HSV参数进行后续识别" 三个模式, 在程序的输出区域也会输出上述提示。

相关操作教程请详阅相关 ReadMe.pdf 说明文档。

影片"1b2r_armors.mp4"灯条识别:

程序: mp4_Light_recognition.cpp

首先对影片的亮度、饱和度及对比度进行调整便于后续识别,随后可以在 TrackBar 视窗中选择识别模式,与上一程序不同的是本程序省去了让你自由发挥的选项,模式选择栏 Mode 只有 "0" --- "使用调整好的红色 HSV 参数进行识别",以及 "1" --- "使用调整好的蓝色 HSV 参数进行识别" 两个方向,图像处理流程基本沿用上一程序,但由于影片中小蓝和小红与摄影机距离较远,故其中有微调判定参数。

注意事项请详阅相关 ReadMe.pdf 说明文档。