

2、 Linux(ubuntu)入门及 opencv 的基本使用

Linux(ubuntu)入门

学习要求：

- 1、 掌握 linux 和 ubuntu 关系，ubuntu 发行版本，维护时间等；
- 2、 学习掌握 ubuntu 文件目录结构（含隐藏文件）；
- 3、 熟练掌握 linux 快捷键和常见终端命令：如快捷打开一个终端，
pwd、date、history、cd（系列）、rm、pwd、vi、ls、cp、mkdir、
cat、clear、poweroff、shutdown、chomd、source 等等
- 4、 一小时入门 vim；（vim 快捷键很多，只要求大家学会 vim 简单使用，ps：vim 需自行安装）
- 5、 学习掌握 ubuntu 其他软件，如录屏软件、科学上网等。（非必须）
- 6、 linux 脚本（非必须）

以上学习内容，需使用 CSDN/github 做笔记记录，提交链接。

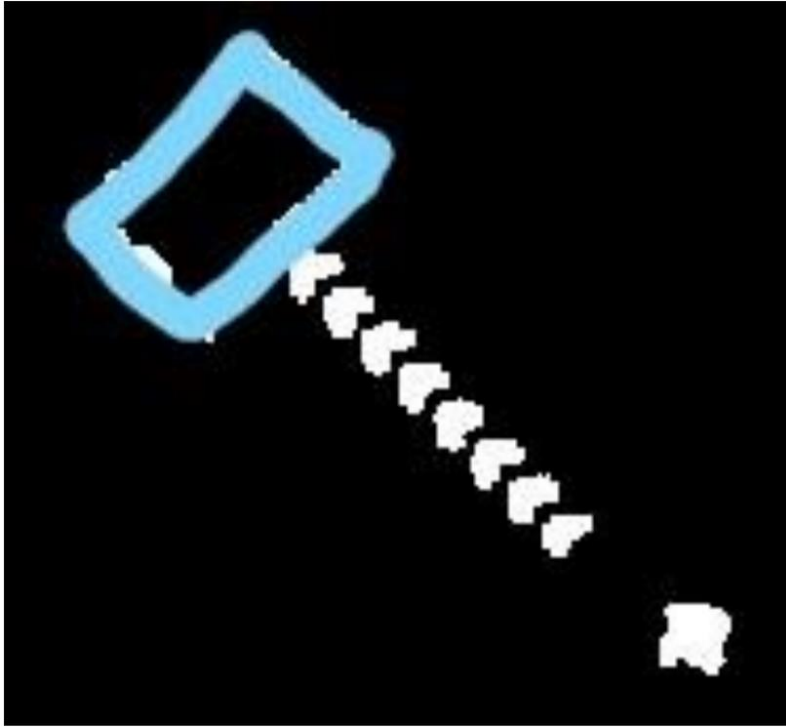
Opencv 基本使用

任务 1：简单的图像处理任务

具体内容：

这是一幅已经二值化的图像，你需要提取出其中的装甲板，即下图所示的蓝色框的内部的四边形。画出找到的四边形并保存为‘result1.jpg’。

注：这里允许直接调用 opencv 的 api，同时提供一下思路：1、利用 findcontours 函数并基于轮廓间关系去识别；2、基于找出的轮廓拟合最小包围矩形并添加约束；3、对图像二次处理，能够抠出内部的区域（可以考虑分水岭填充等算法）。



任务 2：在这个任务中你需要掌握仿射变换和透视变换，了解图像分割的基本理论。

具体内容：

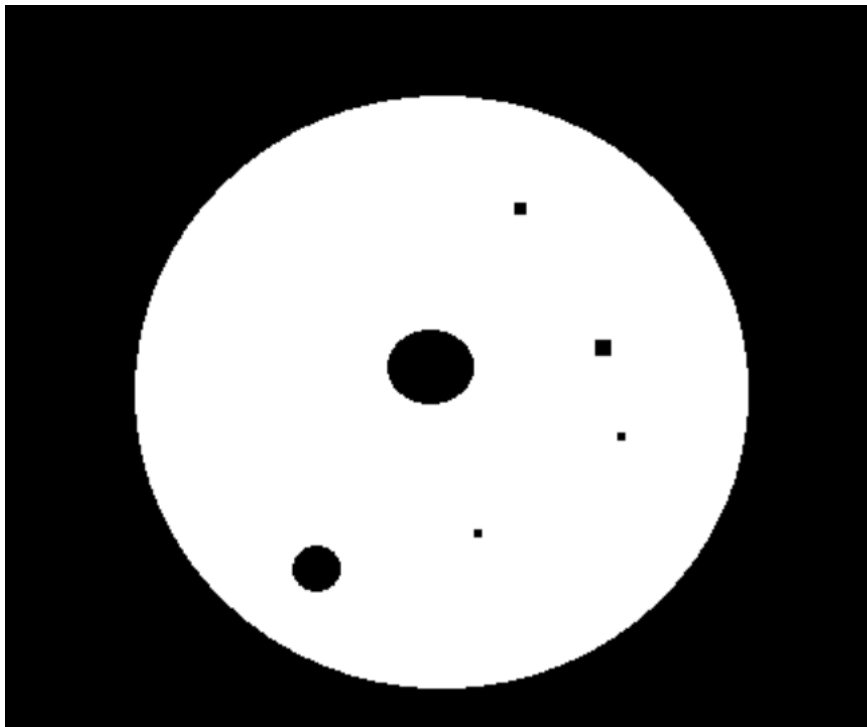
- 1、 识别一张倾斜拍摄的纸张，找出轮廓，提取出该纸张的位置（你可能需要用到边缘提取、霍夫变换或 OpenCV 里的轮廓提取函数，这都取决于你）。
- 2、 假设你已通过图像处理的算法找到发生形变的纸张的位置，那么对这个倾斜纸张进行变换，得到纸张的垂直视图，实现文档校准。（如果你在第一步遇到了问题，那么在这一步你可以进行手动标注纸张边界）



任务 3:

具体内容：下图是一张已经二值化的图片

- (1) 显示白色圆环上最大黑色区域的像素点数量；
- (2) **自行编写代码**过滤掉白色圆环上的黑色小点，并保存处理后的图片为'result3.jpg'



任务 4:

具体内容：

opencv 提供了很多有意思的示例代码, 这些示例程序都在你安装 opencv 的文件夹下面, 找到它们, 并选择一到两个较为复杂的代码尝试分析其底层实现方式。

程序全部采用 C/C++ 编写, 符合基本软件规范, 变量命名做到见名知意, 需要的地方做好注释。

Opencv 部分提交方式: word 报告+源码

参考资料:

- 1、《opencv3 编程入门》
- 2、《学习 opencv3》
- 3、《opencv4 快速入门》