

学号： 2020090917003

姓名： 陈驰

专业方向： 软件工程（数字动漫）

单位名称： 电子科技大学

毕设课题名称：基于基板卡的摄像机阵列控制三维重建扫描系统

校外导师： 匡平

校内导师： 匡平

**信息与软件工程学院**

**本科毕业设计（论文）**

**初期报告**

**第一章 毕设概况**

1.1 毕设题目与内容

本毕业设计的题目是《基于基板卡的摄像机阵列控制三维重建扫描系统》，主要内容是基于主流嵌入式平台与工业RGBD摄像头，设计并实现用于实现三维重建扫描的基于基板卡的摄像机阵列控制系统，要求实现一个PC端的用户GUI，通过用户设定参数自动完成对物体的拍摄，保证图像质量满足后端对三维重建的需求。

## 1.2 毕设所涉及到的主要领域与相关知识

本次毕业设计涉及到多个方面的知识和技术。在嵌入式平台方面，需要掌握Linux操作系统的相关知识，包括Linux的基本命令、文件系统管理、进程管理等。还需要了解Linux下的设备驱动开发和嵌入式系统的搭建。在工业摄像头的控制方面，需要掌握C++编程语言，并熟悉对应品牌摄像头的SDK（软件开发工具包）。通过使用SDK，可以实现对摄像头的控制、图像采集和处理等功能。同时，还需要了解摄像头的基本原理和图像处理算法。在前端界面方面，需要掌握Qt/C++相关的知识。Qt是一个跨平台的应用程序开发框架，可以用于开发图形用户界面（GUI）应用程序。通过使用Qt，可以创建用户友好的界面，并与后端逻辑进行交互。此外，系统各个模块之间通过局域网进行通信，需要了解计算机网络的相关知识。包括网络协议、IP地址、端口等基本概念，以及网络编程的基本原理和技术，如使用套接字（socket）进行网络通信。

# 前期准备

## 2.1 可行性研究

三维重建是目前计算机视觉、机器学习、计算机图形学等领域的热门方向。三维重建的过程中离不开所要重建物体的各个角度的高清图像信息。本次毕设的主要工作就是为这一需求服务，即在合适的角度上，控制多个高清摄像机同时对重建物体进行拍照。为了获取清晰的图像，在拍照时需要调整摄像机的焦距、光圈、曝光时间等参数。同时，系统对拍照的同时性也有着非常高的要求。

经济可行性方面，目前系统仅使用5台相机与嵌入式控制设备来进行初期开发，开销如下表所示。

|  |  |  |  |
| --- | --- | --- | --- |
| 设备类别 | 型号 | 数量 | 单价 |
| 工业相机 | 海康威视MV-CU050-90UC | 5 | 1400 |
| 工业镜头 | 海康威视MVL-HF0624M-10MP | 5 | 400 |
| 嵌入式设备 | OrangePi3B （2G） | 5 | 240 |

表2-1 硬件开销

技术可行性方面，本次毕业设计用到的主要技术有：1. 嵌入式C++开发。2. 工业相机、镜头的使用。3. Qt/C++开发。4. OpenCV库的使用。

## 2.1 硬件准备

目前，系统所需要的硬件已经全部准备齐全。

嵌入式设备：香橙派2B 开发板。

工业摄像头：海康机器人MV-CU050-90UC型工业相机。

镜头：海康威视MVL-HF0624M-10MP型工业镜头。

PC终端一台。

## 2.2 开发环境准备

Visual Studio 2022 Community

QT6 Creater

OpenCV 4.8.0

Ubuntu 20.04

# 第三章 当前进度

## 3.1 硬件相关知识的学习

首先是工业相机。工业相机相比较于日常中普通的相机比，其精度更高，帧率更高，同时暴露给使用者以控制其参数的接口也更多。其基本使用流程如下图所示。

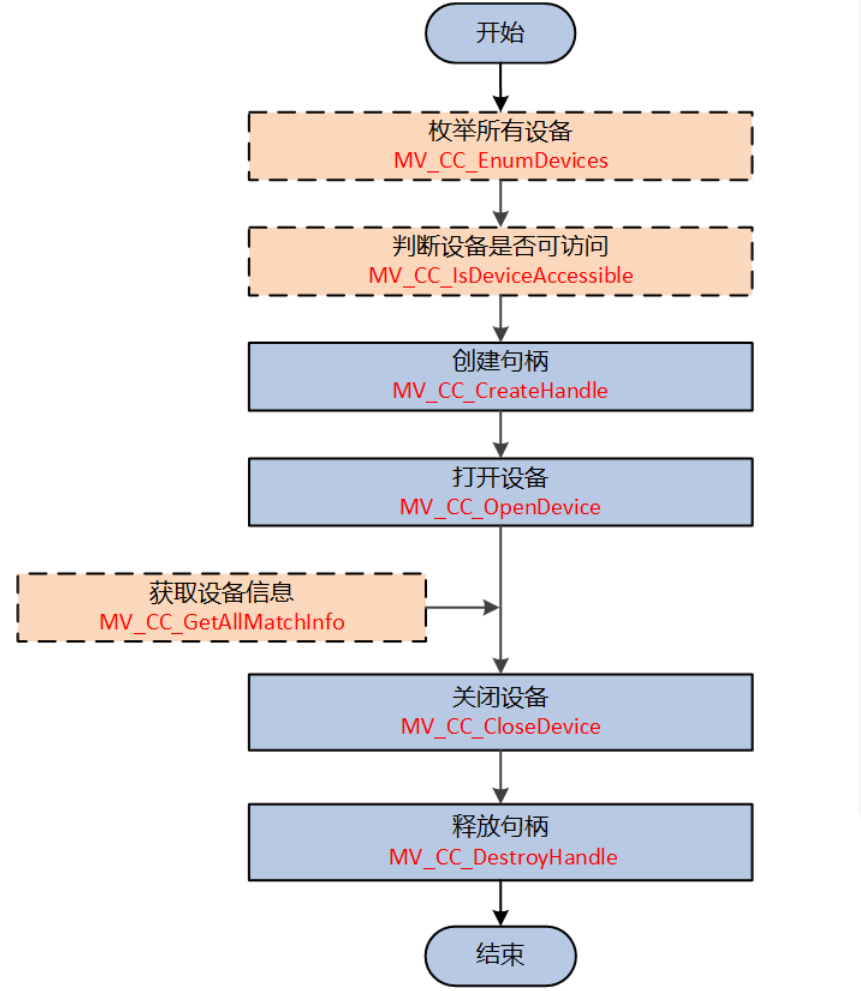
****

图3-1 工业相机的使用流程

1. 调用 MV\_CC\_EnumDevices() 枚举子网内指定传输协议对应的所有设备。 可通过nTLayerType在结构 MV\_CC\_DEVICE\_INFO() 中获取设备信息。

2. 在打开指定设备前，调用 MV\_CC\_IsDeviceAccessible() 检查指定设备是否可访问。

3. 调用 MV\_CC\_CreateHandle() 创建设备句柄。

4. 调用 MV\_CC\_OpenDevice() 打开设备。

5. 调用 MV\_CC\_GetAllMatchInfo()以获取设备信息。

6. 调用 MV\_CC\_CloseDevice() 关闭设备。

7. 调用 MV\_CC\_DestroyHandle() 销毁句柄并释放资源

海康机器人摄像头的SDK中有着丰富的控制接口，但是对于我们目前的系统，很多功能都无需使用与了解，因此需要编写一层相机的封装来方便系统使用相机，相机的封装代码如下。

1. #pragma once
2. #include "Camera.h"
3. #include <cstdint>
4. #include "MvCameraControl.h"
6. **class** HIKCamera : **public** Camera
7. {
8. **public**:
9. ~HIKCamera();
10. HIKCamera() = **delete**;
11. HIKCamera(MV\_CC\_DEVICE\_INFO\* pDeviceInfo);
13. **bool** Open();
14. **bool** Close();
15. **bool** IsOpen();
16. CameraFrame GetFrame();
17. **private**:
18. **void**\* m\_handle;
19. MV\_CC\_DEVICE\_INFO m\_info;
20. **bool** m\_isOpen;
21. };
23. #endif // !HIKCAMERA\_H

代码块3-1 相机类的封装 C++接口

不同品牌的相机都派生自Camera抽象基类，对调用者提供打开并初始化相机接口Open()，关闭并释放资源接口Close()，判断相机是否已经打开的接口IsOpen()，还有最关键的使用当前相机获取一帧图像的接口GetFrame()。经过这样的封装，不仅在编写代码时，无需关心特定品牌（当前是海康）的相机SDK的复杂实现方式，也能将不同品牌的相机视为同一对象，进行统一使用和管理，体现了面向对象编程的特点。

1. **int** main(**int** argc, **char**\*\* argv)
2. {
3. CameraManager\* s\_CameraManager = CameraManager::GetInstance();
4. std::shared\_ptr<Camera> cam = s\_CameraManager->GetOrOpenCamera();
6. CameraFrame frame(cam->GetFrame());
7. cv::imshow("Camera", frame);

10. cv::destroyAllWindows();
11. **return** 0;
13. }

代码块3-2 主程序

如上所示，经过封装后的相机在使用时只需先打开，即可调用其提供的接口来获得一帧CameraFrame图像。

**初期报告主要从以下三个方面进行阐述（参考毕业论文正文格式：宋体小4号，英文字体为Times New Rome，行间距固定20磅），2000字以上：**

1. **毕设项目的主要内容和指标**
2. **毕设项目实施方案简述（在多学科环境中，从软件开发、工程管理、以及经济决策等方面初步考虑项目组织和实施方案）**
3. **毕设项目可行性研究简述（在社会、健康、安全、法律、文化以及环境等约束条件下，对设计方案的可行性进行简单描述）**
4. **前期(项目)基础（毕设项目已经具备的前期技术基础，以及相关开发环境的准备工作情况）**