

微型无人机载光电吊舱显控软件的设计与实现

王天澳¹, 于 洵², 朱 镭³, 王 刚⁴, 张 杨¹, 黄书佳¹

(1. 西安工业大学 光电工程学院, 西安 710021; 2. 西安工业大学 兵器科学与技术学院, 西安 710021;
3. 西安应用光学研究所, 西安 710021; 4. 中国航天空气动力技术研究院, 北京 100074)

摘要: 针对微型光电吊舱的特点, 使用 Qt 框架和 C/C++ 语言设计并实现了某型机载微型光电吊舱的显示与控制软件; 该软件在设计上采用了多线程的方式, 分别实现了基于 QtSerialPort 模块的串口数据收发; 基于开源编解码库 FFmpeg 的视频流接收和解码; 基于 YOLOv3 模型检测算法的目标检测和识别等多项任务; 各模块内部独立使用线程, 并利用 Qt 的信号槽机制传递消息, 使软件程序层次清晰, 可读性强, 并有利于后续功能的修改和添加; 在地面高性能设备中使用 YOLOv3 算法做目标识别, 在视频延时符合要求时, 明显提升了检测速度和识别精度; 最后, 通过多次实验, 验证了该显控软件的可靠性和有效性。

关键词: 光电吊舱; 显控软件; Qt; 目标识别; 多线程; C++; YOLOv3

Design and Implementation of Display-console Software for Micro Unmanned Electro-optical Pod

WANG Tianao¹, YU Xun², ZHU Lei³, WANG Gang⁴, ZHANG Yang¹, HUANG Shujia¹

(1. School of Optoelectronic Engineering, Xi'an Technological University, Xi'an 710021, China;

2. School of Ordnance Science and Technology, Xi'an Technological University, Xi'an 710021, China;

3. Xi'an Institute of Applied Optics, Xi'an 710021, China;

4. China Academy of Aerospace Aerodynamics, Beijing 100074, China)

Abstract: According to the characteristics of micro electro-optical pod, the Qt framework and C/C++ language are used to design and implement the display and control software of a certain type of airborne micro-photoelectric pod; the software is designed in a multi-threaded way to realize the serial ports based on the data transmission and reception of the QtSerialPort module, and the video stream reception and decoding based on the open source decoding library FFmpeg; the multiple tasks of the target detection and recognition based on the YOLOv3 model detection algorithm; each module uses threads independently, and the Qt's signal slot mechanism is used to transmit messages and make the levels of the software clear and the readability strong, and it is conducive to the modification and addition of the subsequent functions; the YOLOv3 algorithm is used for the target recognition in the ground high-performance equipment, and the detection speed and recognition accuracy are significantly improved when the video delay meets the requirements; finally, the reliability and effectiveness of the display and control software are verified after many experiments.

Keywords: electro-optical pod; display console software; Qt; target recognition; multi-threading; C++; YOLOv3

0 引言

机载光电吊舱挂载于固定翼飞机、多轴旋翼飞机等常见机型, 其集成了光电探测、稳像搜索、激光测照、光电图像处理等技术, 可以完成目标跟踪/定位、疆场态势感知、军械引导与打击效果评估等作战任务^[1]。当前, 光电吊舱集多传感器于一体, 并向着全天候、高分辨率、远距离、综合化、小型化发展^[2]。微型光电吊舱正是这种发展趋势下的产物, 在敌情侦察、边疆巡检、火灾监控、海上救援等军民领域都发挥着重要的作用^[3]。

目前, 国内外的光电领域研究人员大多着重于光电吊

舱的结构优化、光学性能提升等技术应用及发展趋势。如: 王晨宇^[4]等提出光电吊舱的分数阶鲁棒控制器设计与仿真, 优化了光电吊舱控制系统的控制性能; 王鹏鹏^[5]等提出某无人机光电吊舱隔振机构设计和试验研究方法, 提升了光电吊舱的隔振效果; 魏志强^[6]等研究了光电吊舱双路视频压缩存储方法, 提高了双路视频压缩传输效率; 张孟陶^[7]等对光电吊舱的发展趋势做了深入的研究。

而与光电吊舱配套的地面显控软件则得到的关注较少, 合理的地面显控软件同样对光电吊舱性能的发挥起着至关重要的作用^[8]。地面操控人员通过地面显控软件, 可以实

收稿日期: 2021-12-22; 修回日期: 2022-02-01。

作者简介: 王天澳(1993-), 男, 江苏徐州人, 硕士, 主要从事机载光电吊舱显控软件设计方向的研究。

于 洵(1964-), 男, 陕西西安人, 教授, 主要从事光电信息系统与技术的教学方向的研究。

引用格式: 王天澳, 于 洵, 朱 镭. 微型无人机载光电吊舱显控软件的设计与实现[J]. 计算机测量与控制, 2022, 30(3): 145-150, 178.

时了解光电吊舱内部光电传感器的运行状态,电视/红外传感器实时拍摄的效果,并实现对光电吊舱的实时调整,以此提高任务的执行效率。

因此,开发一款高效的光电吊舱显控软件成为一种必要的需求。本文针对某微小型机载无人光电吊舱,设计了一款基于跨平台框架 Qt 的多模块集成光电吊舱显控软件。

1 显控软件系统概述

1.1 功能设计

本地面显控软件配套的微小型光电吊舱,主要挂载于微小型无人机、巡飞弹等平台,通过无线数传/图传构建数据链,将地面显控软件和机载光电吊舱相连接。根据微小型光电吊舱的任务需求,显控软件主要包含以下功能。

1) 实时视频显示功能:通过千兆网口连接无线图传,使用 UDP 协议接收视频流,实时对视频流进行解码显示,并根据项目需要,在图像上叠加显示十字光标、跟踪框、测距值等辅助标识。

2) 光电吊舱控制功能:获取地面人员的操作信息(调节电视、红外参数、电源通断、摇杆位移等),并转换为控制指令,发送给光电吊舱。

3) 光电设备状态显示功能:地面设备连接无线数传,获取光电传感器的实时状态数据,软件实时解析数据,然后以文字、提示灯等多种形式展示。

4) 数据记录功能:显控软件可以实时保存视频裸流和光电吊舱状态数据,以便后续的分析。

系统总体框图如图 1 所示。

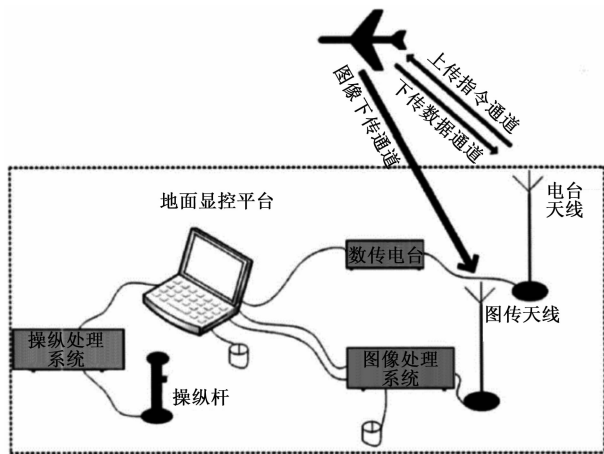


图 1 地面显控平台系统框图

1.2 模块设计

根据上述系统的功能要求,对显控软件的功能模块进行设计,具体设计如图 2 所示。同时,表 1 详细介绍了微小型光电吊舱显控软件组成模块的具体功能。

机载光电吊舱通过数据链路下行的视频流数据将被存入数据缓存,然后从缓存提取视频流,通过开源解码库 FFmpeg 的相关方法对获取的视频流进行解码,解码后得到 YUV420P 格式图像,此格式无法直接在 Qt 中使用控件显

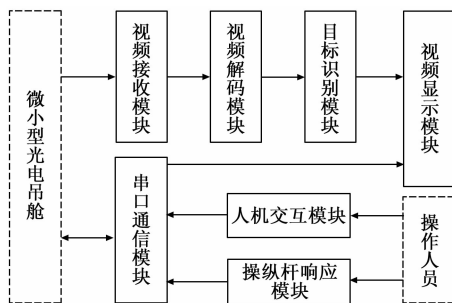


图 2 功能模块图

表 1 显控软件模块功能表

| 模块名称 | 模块功能 |
|---------|---|
| 视频流接收模块 | 选用 UDP 协议进行网络通信,接收光电载荷实时下传的 H.264 编码视频流 |
| 视频解码模块 | 对实时接收的视频流进行软解码,并转换为 Qt 支持的图像格式 |
| 串口通信模块 | 实现与光电载荷的串口通信,实现控制指令发送和载荷状态接收 |
| 人机交互模块 | 响应地面人员在软件界面的人机交互操作,并生成对应的控制指令 |
| 摇杆响应模块 | 获取外接摇杆的动作类型和输入,生成对应的控制指令 |
| 目标识别模块 | 基于 YOLOv3 算法,对解码后的图像进行目标识别 |
| 显示模块 | 在 Qt 显示控件上,显示红外视频、可见光视频及传感器的状态信息 |

示,故将其格式转变为 RGB32^[9],然后目标识别模块对图像中的特定目标进行标记,最后在 Qt 的控件上定时刷新显示。

光电吊舱通过无线数传下传光电传感器的状态信息,显控软件通过串口连接到地面无线数传,串口收发模块接收数据并根据相关算法进行解算,然后以数值或标记灯的形式在界面上显示。人机交互模块和摇杆响应模块,获取地面操作人员在显控软件上的操作(调节电视、红外参数、电源通断等)或摇杆移动动作,填入到传输报文中,然后通过串口收发模块发送到无线数传中,上行到光电吊舱,实现对光电吊舱的操控。

2 软件设计与实现

2.1 Qt 开发环境简介

Qt 是一款基于 C++ 语言的跨平台图形界面开发,基于它开发的相关软件,源代码无需修改就可以在市场上绝大部分平台上运行,而且会自动适应相应操作系统的图形界面风格^[10]。另一方面,Qt 其独有的对象间信号槽通信机制取代了传统的回调函数机制,这降低了对对象之间的耦合度,保证了传递参数的准确性。Qt 同时有着全面又完善的 C++ 图形库,其集成的模块有 SQL 模块、脚本模块、Network 模块、XML 模块等,并且这些模块都支持多线程操作,这显著提高了 Qt 开发复杂大规模工程的能力^[11]。

QtCreator 是由诺基亚公司研发的一款新的集成开发环境, 功能齐全, 开发界面简洁, 环境配置简单, 可以快捷的完成 Qt 项目的开发^[12]。Qt Creator 包括带语法标识的 C++ 语言编辑器、GDB 调试功能、源代码版本控制软件等。本文显控软件的设计基于 Qt5.12.1 版本下开发的, 版本高包且包含模块丰富, 能够满足项目开发的需要。

2.2 软件界面设计

本文基于 QMainWindow 主窗口类设计了显控软件。考虑到软件操作的便捷性和扩展性, 在主窗口中添加了菜单栏、状态栏和工具栏等。在主窗口中添加界面基类 QWidget, 并将具体的功能组件, 如 QLabel、QGroupBox、QSpinBox 等放入 QWidget 类中来实现具体的功能。另外, 还使用了 QTabWidget 分页组件来切换不同传感器的控制页面, 显控软件的主界面如图 3 所示。



图 3 显控软件界面设计

该显控软件的主界面主要考虑到了易用性、合理性、可靠性等几方面的设计理念。显控软件主要分为: ①控制分区: 对小型光电吊舱(包括电视/红外传感器、吊舱伺服、记录仪/激光等组件)的实时控制; ②视频显示分区: 显示电视/红外视频(也可以开启目标识别、双击视频全屏等功能), 并根据需要添加瞄准十字线、激光测距值、方位角等信息; ③光电吊舱状态显示分区: 显示光电吊舱的方位角、横滚角、系统模式、传感器选择、目标状态等信息; ④操杆选项分区: 配置操杆, 选择启用实体操杆还是虚拟操杆等 4 个主要区域, 合理的功能分区大大提高了显控软件的易用性。

2.3 多线程设计

本次设计显控软件采用多线程方式, 从而实现各个模块的并行运行, 软件设计流程图如图 4 所示。

根据 Qt 的开发规则, 主界面只能放在主线程中, 所以将视频显示和人机交互放在主线程中。运行软件后, 主线程启动, 在其中创建 5 个子线程并启动, 以此达到并行运行的效果。线程之前的通信, 使用 Qt 独有的信号槽机制, 并加互斥锁, 保护共享数据。软件结束后, 主线程中的析构函数内停止线程、销毁线程, 释放内存资源^[13]。

每个线程都对应一个模块的类, 在相应的类里面启动线程, 运行线程对应功能。以视频解码模块为例子, 首先

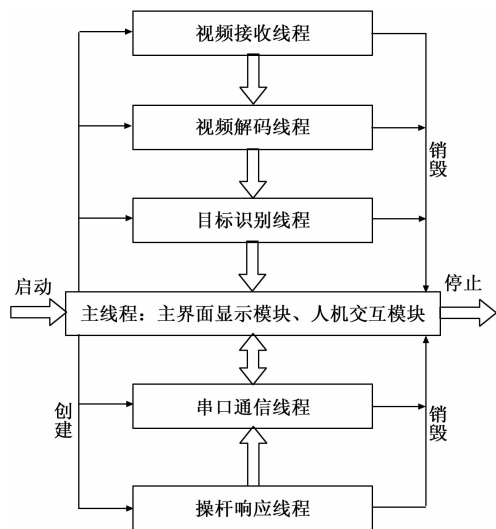


图 4 显控软件多线程设计

创建一个解码类 VideoDecoder, 继承于 QObject, 具有基类的特性。在解码类头文件中, 定义所涉及到的变量和函数, 包括解码起止函数 decoderStart()、decoderStop() 等。另外, 需要注意的是要定义一个变量, 如 mflag 来标记线程的状态, 根据这个标记变量的值, 运行线程需要执行的内容。定义的线程继承线程基类的开始和结束函数 QThread::start() 和 QThread::stop()^[14], 通过信号槽关联到解码类中的起止函数。在 VideoDecoder 类的构造函数中需包含以下代码:

```
connect(&thread, &QThread::started, this, &VideoDecoder::decoderStart);
connect(&thread, &QThread::finished, this, &VideoDecoder::decoderStop);
moveToThread(&thread);
```

2.4 串口通信

显控软件通过串口与地面无线数传/图传相连接, 天空端和地面端的无线数传/图传构成无线数据链路, 实现数据的远程传输。考虑到需要实时上行和下行数据, 采用 RS422 串口与无线数传相连进行通信。QT5 提供了串口通信模块 QtSerialPort 模块, 为硬件和虚拟的串口提供了统一接口^[15]。使用 Qt 自带的串口模块, 在开发需要串口通信模块的软件时, 大大缩短了开发周期^[16]。

在使用 QtSerialPort 模块时, 只需在工程文件中加上代码 Qt += serialport。QtSerialPort 提供了访问串口的接口函数, 首先在主函数中创建一个 QSerialPort 类对象 mySerialPort, 其次, 初始化串口, 获取当前连接串口号; 再次, 设置串口连接模式为读写; 最后将串口读取信号和对应的槽函数 ReadData() 相关联^[17]。每当串口有新数据可从设备的当前读取通道读取时, 该信号就会发出一次, 接收到信号后, 对应槽函数开始对串口数据进行读取。串口通信流程图如图 5 所示。

串口相关代码如下:

```
mySerialPort = new QSerialPort(this); //新建一个串口对象, 并初始化
```


4) 视频流解码: 打开解码器后, 使用 `av_read_frame` 函数读取一包未解码数据, 把它解码成一帧图像, 然后通过 `av_send_packet` 函数发送解码后图像, `av_receive_frame` 函数接收解码后图像。

5) 视频显示: 调用 `sws_scale` 函数把解码后图像从原始格式转换成 Qt 支持的 RGB 格式, 再调用 OpenCV4.5.1 视觉库在图像上叠加辅助线、文字等信息, 最后通过定时刷新, 实现视频解码后的播放。

2.6 目标检测和识别

传统光电吊舱中, 目标检测和识别模块放在机载嵌入式模块里, 而微小型光电吊舱体积小, 重量轻, 无法放置高性能目标识别模块, 故将目标识别模块放在地面显控软件中实现^[20]。地面站中专用服务器的算力相较机载嵌入式设备有着显著的提高, 配合上目前市面上比较成熟稳定的目标检测和识别算法 YOLOv3^[21], 显著提升了目标的检测速度和识别准确率。

2.6.1 Yolov3 算法简介

Yolov3 算法的基础网络由 Darknet-53 架构组成, Darknet-53 网络结构是在 Darknet-19 和残差网络的两个网络的基础上做了改动。由表 3 可以看出, 改进后的 Darknet-53 网络在精确率方面高于 Darknet-19 和 Resnet-10 网络, 不过与 Resnet-152 网络相比略低, 但是此网络每秒测量浮点运算的速度最高, 比其他网络更快。

表 3 Darknet-53 网络与其他网络对比

| Backbone | Accuracy | Bn Ops | BFLOP/s | FPS |
|------------|----------|--------|---------|-----|
| Darknet-19 | 73.1% | 7.29 | 1246 | 171 |
| ResNet-101 | 77.1% | 19.7 | 1039 | 53 |
| ResNet-152 | 77.6% | 29.4 | 1090 | 37 |
| Darknet-53 | 77.2% | 18.7 | 1457 | 78 |

2.6.2 软件中实现

由于本次项目针对的目标为民用车, 官方训练的模型已经包括了 this 类别, 而且识别精度比较高, 故直接使用了官方给的权重文件, 这有效缩短了开发周期。在开发系统中配置 CUDA 环境, 调用 OpenCV4.5.1 图像处理库读取 YOLOv3 训练后的权重文件, 开启 GPU 加速, 然后对实时接收、解码后的视频进行目标检测和识别。

```
// 载入权重文件
```

```
Net net = readNetFromDarknet(modelConfiguration, modelWeights);
```

```
net.setPreferableTarget(DNN_TARGET_CUDA); // 使用 GPU 加速
```

```
net.setPreferableBackend(DNN_BACKEND_CUDA);
```

```
cv::rectangle(fpicture, Point(left, top - 2 * labelSize.height - baseLine), Point(left + round(labelSize.width), top - baseLine + 2), scalar, FILLED); // 检测完成标记目标
```

```
putText(frame, label, Point(left, top - 0.5 * labelSize.height - baseLine - 1), FONT_HERSHEY_SIMPLEX, 0.8, Scalar(0, 0, 0), 1);
```

上述代码通过 OpenCV 读取 Yolov3 权重文件和对应标签文件, 对视频图像逐帧检测, 最后使用 OpenCV 中的 `rectangle`、`putText` 等函数对识别后的目标进行框选。

2.7 摇杆操控模块

机载伺服控制模块通过获取的位移量去控制光电吊舱的横滚和俯仰角度。目前项目中, 使用实体摇杆获取位移量发送给天空端。位移量的数值代表移动的速度, 正负代表移动方向。同时可以在摇杆中定义一些快捷键, 如跟踪和锁定, 这样使地面人员的操作更加便捷。

该软件中摇杆控制使用的 `winmm.dll` 是 Windows 多媒体相关应用程序接口。如图 8, 通过 `QLibrary` 类, 在程序运行时加载 `window` 动态链接库 `winmm.dll`, 通过 `resolve()` 解析并调用库中的具体方法, 这样再下面的程序中可以直接调用 `qJoyGetPos` 这些解析过的方法了。摇杆数据获取编程设计流程如图 8 所示。

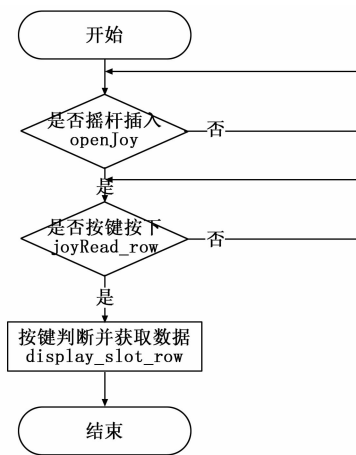


图 8 摇杆数据获取流程

从图 8 可知, 先检测当前系统是否有摇杆插入, 然后检测是否按键按下, 不然就一直循环查询, 知道获取到数据为止, 利用 QT 独有的槽函数, 将摇杆数据结构 `joyinfoex_tag` 发射到主线程的 `display_slot_row(joyinfoex_tag-state_row)` 函数中。从 `joyinfoex_tag` 结构体, 获取各种按键的信息, 判断出是哪一个按键按下, 再根据预先定义的功能, 生成相应的指令。

3 软件测试

显控软件开发完成后, 与某型微小型光电吊舱配套, 在外场搭载于巡飞弹、某小型无人机等飞行器。对电视/红外探测器控制、激光照射等传感器指标进行完整测试, 并基于目标的检测识别, 对目标进行锁定、跟踪等测试。测试中, 光电吊舱与地面站通过无线数传/图传相连接, 选用的无线数传/图传最大支持 6 Mb/s 的传输速率。下面对各个模块进行一一测试。

3.1 串口通信测试

在测试中, 将 RS422 USB 转串口连接电脑, 波特率设置为 115 200, 数据位 8 位, 停止位 1, 没有校验位和流控

