****

2023-2024（2）

《C++程序开发实训》

**第5周周报**

# 该周应该完成的任务

1. 优化系统性能，提升识别准确率和实时性。
2. 完善用户界面和交互体验。
3. 整理代码，准备完成实训报告的编写。

# 该周已完成的工作

1. 对系统性能进行优化，提高识别准确率和实时性
   1. **采用“多线程技术”进行优化**

在前几周完成的工作的基础上，人脸识别门禁系统已经能够初步使用，但在对其测试的过程中发现了问题：在学生的刷脸过程中经常会出现识别过程中系统卡顿的现象，基于这个问题，提高阅读代码进行分析，发现了较为耗时的操作：



为了解决上述所说的问题，考虑使用**Qt框架的多线程功能**：创建了一个新的线程，并将QFaceObject对象fobj移动到该线程中执行，而不是在主线程（也称为GUI线程）中运行。因为QFaceObject 对象涉及到耗时操作，如图像处理，将其移动到线程中执行有利于节省时间。在多线程环境中，这有助于保持主线程的响应性。

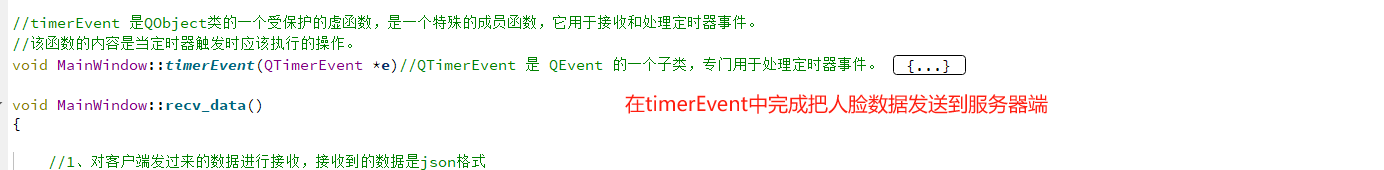


使用多线程技术后，系统卡顿问题得到缓解，识别的准确性和实时性也有了提高，说明系统性能得到了优化。

* 1. **采用“标志变量”进行优化**

学生在人脸识别系统刷脸时，客户端通过启动定时器，在定时器事件中来把人脸数据发送到服务器端，在进行测试过程中，发现同一个学生人脸进入刷脸系统后，在短时间内会进行多次人脸查询，多次学生信息显示。针对这个现象，考虑对系统进行优化。





首先考虑的是把触发定时器的间隔时间调大一些，但间隔时间调大后，采集并发送人脸数据的频率变低，使得人脸识别系统的实时性降低，故该方法不采纳。

在保证系统实时性的同时要避免短时间内对同一张人脸的多次识别，考虑**使用一个标志变量flag**来记录当前客户端采集的人脸与刚刚识别出来的人脸是否属于同一个人脸。根据flag的值来确定是否发送数据到服务器端，避免多次发送与识别人脸。

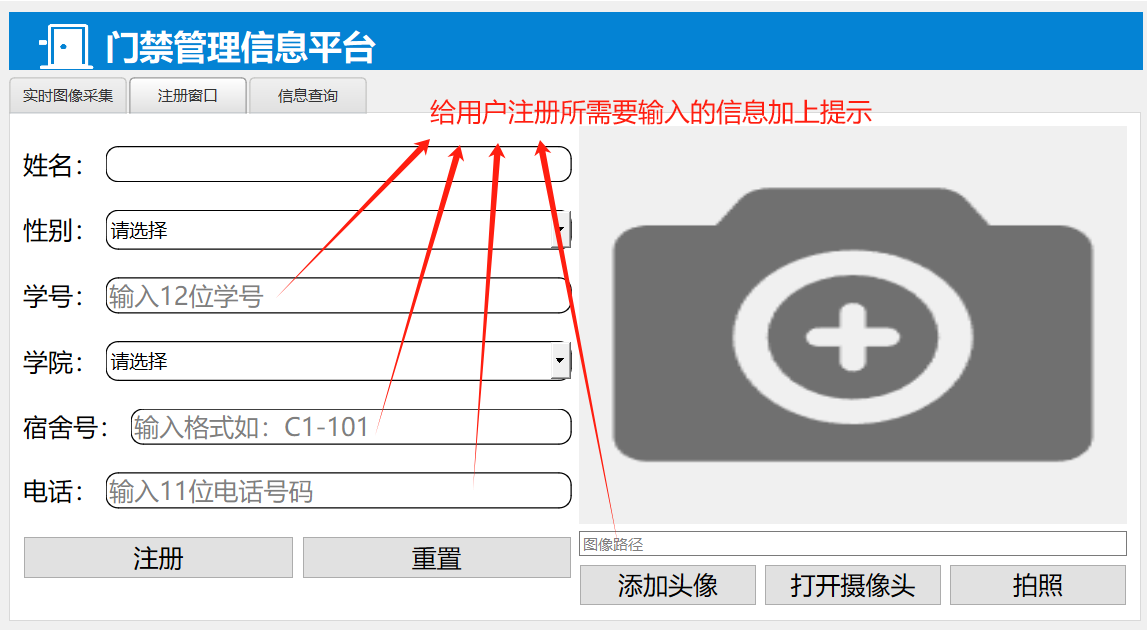




使用标志变量后，系统的性能得到优化，实时性也有一定的提高。

1. 完善用户界面和交互体验。

在前几周设计的界面的基础上进行完善，在需要用户手动输入的内容加上提示，以便用户能够输入正确格式的数据。



同时，当用户输入信息不完整，或者输入了不符合要求的数据，会显示一个警告消息框（QMessageBox）来提示用户。

**信息填写不完整**



**未采集图片**



**注册时采集的图像中不包含人脸时**

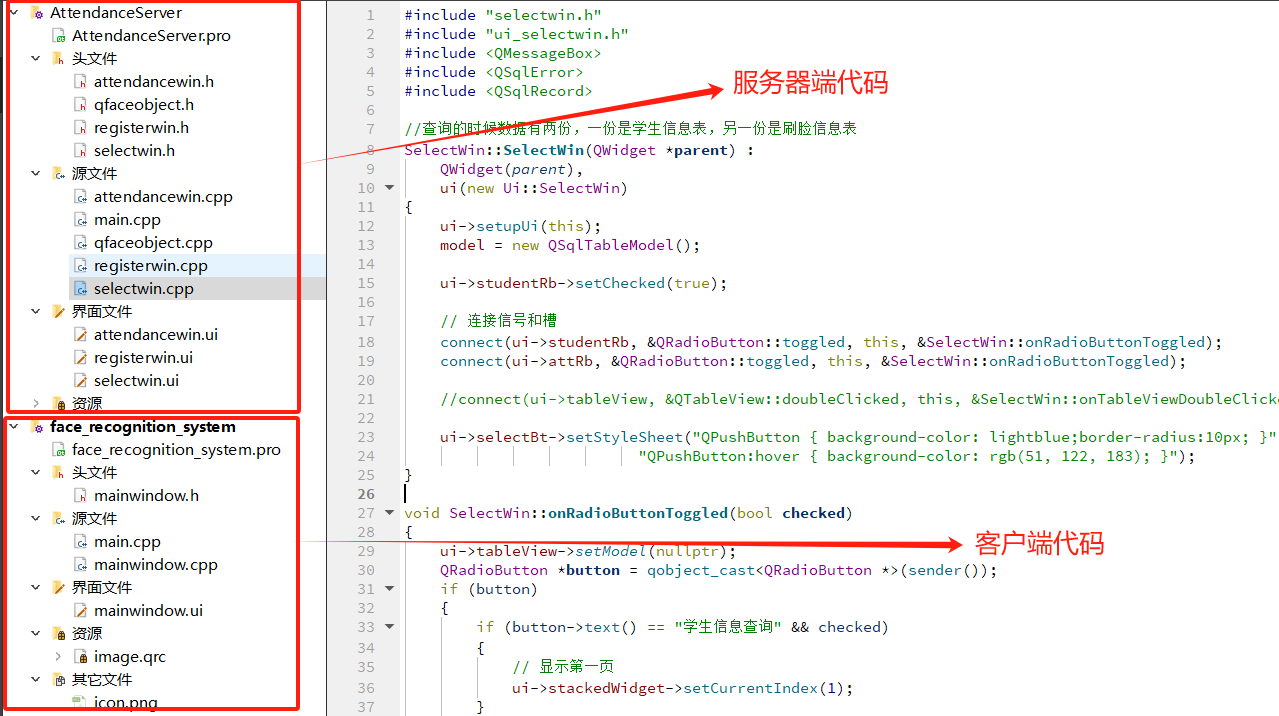


**输入信息格式错误**



**至此，用户界面完善工作完成，提高了用户的交互体验。**

1. 完成代码整理，为后续编写实训报告做好准备



# 目标达成情况

1、 优化系统性能：目标已达成，系统的识别准确率和实时性得到提升。

2、完善用户界面和交互体验：目标已达成，为用户界面加上了提示信息，便于用户输入准确的数据。

3、整理代码，为实训报告的编写做准备：目标达成。