

接口设计说明书

一、前言

本文档的编写，旨在方便系统开发人员或者系统维护人员能够更好的了解系统的结构、各个系统层次之间的交互。根据该文档，读者可以很好地了解到，系统中的每个接口的作用，输入和输出要求。

二、系统接口概述

基于 opencv 的人脸识别考勤系统分为客户端与服务器端。客户端通过人脸识别模块对人脸进行图像采集。服务器端的人脸模块则负责将当前图像与数据库图像进行匹配返回匹配结果。两端的通信通过 socket 编程实现。同时服务器端可并行接收客户端的连接请求。

三、服务器端系统

3.1、服务器端初始化，建立监听线程并接受客户端连接

1、函数名：Server()

【函数原型】Server()

【接口说明】服务器端 UI 初始化

【输入参数】void

【输出参数】无返回

【调用权限】public

2、函数名：btnStarServer_Click()

【函数原型】void btnStarServer_Click(object sender, EventArgs e)

【接口说明】服务器 socket 初始化，并接收客户端连接

【输入参数】无

【输出参数】无

【调用权限】private

3、函数名：StartAccept()

【函数原型】void StartAccept()

【接口说明】服务器监听线程

【输入参数】无

【输出参数】无

【调用权限】private

3.2、服务器端接收客户端的数据包并进行响应

1、函数名：Startlisten()

【函数原型】void Startlisten(Server f, ClientInfo client)

【接口说明】为每个客户端连接建立接收线程

【输入参数】数据结构 ClientInfo（保存与客户端建立的套接字描述符）。

【输出参数】无

【调用权限】private

- 2、函数名: ReCall()
 - 【函数原型】void ReCall(string t)
 - 【接口说明】多线程的 UI 操作
 - 【输入参数】字符串 t
 - 【输出参数】无
 - 【调用权限】private
- 3、函数名: SendToClient()
 - 【函数原型】void SendToClient(ClientInfo client, string Msg)
 - 【接口说明】向客户发送数据包
 - 【输入参数】数据结构 ClientInfo 和消息 Msg
 - 【输出参数】无
 - 【调用权限】private
- 4、函数名: MsgRecv()
 - 【函数原型】void MsgRecv(ParserPack msg, ClientInfo client)
 - 【接口说明】将接收到的协议包交付处理
 - 【输入参数】协议包 msg 与数据结构 client
 - 【输出参数】无
 - 【调用权限】private
- 5、函数名: StructToBytes()
 - 【函数原型】byte[] StructToBytes(object structObj)
 - 【接口说明】将目标对象转化为字节数组
 - 【输入参数】对象 structObj
 - 【输出参数】byte[]
 - 【调用权限】private
- 6、函数名: BytesToStruct()
 - 【函数原型】object BytesToStruct(byte[] bytes, Type strucType)
 - 【接口说明】解析客户端发送来的数据流, 将接收缓冲区中的数据解析成 Parserpack 类型。
 - 【输入参数】byte[], 类型 strucType
 - 【输出参数】转化目标类型
 - 【调用权限】private

3.3、服务器端的数据库处理

- 1、函数名: connectSQL()
 - 【函数原型】void connectSQL()
 - 【接口说明】连接数据库
 - 【输入参数】无
 - 【输出参数】无
 - 【调用权限】private
- 2、函数名: enroll()
 - 【函数原型】void enroll(string ID, string name, string password)
 - 【接口说明】用户注册操作
 - 【输入参数】ID, Name, password

【输出参数】无
【调用权限】private

3.4、服务器端的模式识别

1、函数名：CreateLib()

【函数原型】void CreateLib(CvMat* M, IplImage **p, int x, int y, int start, int end)

【接口说明】读取图像数组 p 中的图像信息，将像素信息保存至矩阵 M

【输入参数】p 是一个图像数组指针，x 与 y 是图像的大小，start 与 end 是该指针的首末位置，M 为提取后的像素矩阵

【输出参数】无

【调用权限】public

2、函数名：Mean()

【函数原型】bool Mean(CvMat* M, CvMat* O, int x, int y, int flag)

【接口说明】求矩阵的平均值，矩阵 O 为输入矩阵，矩阵 M 为输出矩阵，

【输入参数】矩阵 O 为输入矩阵，矩阵 M 为输出矩阵，x 与 y 为矩阵的大小，flag=0，按行求平均，flag=1，按列求平均

【输出参数】成功返回 1，失败返回 0

【调用权限】public

3、函数名：Lda()

【函数原型】int Lda(int C)

【接口说明】LDA 算法，实现人脸图像匹配

【输入参数】C 为样本库中样本的个数

【输出参数】返回值为匹配的结果

【调用权限】public

四、客户端系统

4.1、客户端初始化

1、函数名：StartConnect()

【函数原型】void CtestDlg::StartConnect()

【接口说明】开启后台操作连接客户端

【输入参数】无

【输出参数】连接失败弹出错误提示框

【调用权限】public

4.2、客户端功能

- 1、函数名： OnBnClickedButton1()
 - 【函数原型】 void Enroll::OnBnClickedButton1()
 - 【接口说明】 实现客户端的注册功能
 - 【输入参数】 无
 - 【输出参数】 无
 - 【调用权限】 public
- 2、函数名： OnBnClickedOk()
 - 【函数原型】 void Login::OnBnClickedOk()
 - 【接口说明】 实现客户端的登录功能
 - 【输入参数】 无
 - 【输出参数】 无
 - 【调用权限】 public
- 3、函数名： OnBnClickedCheck()
 - 【函数原型】 void Sys::OnBnClickedCheck()
 - 【接口说明】 上班考勤操作
 - 【输入参数】 无
 - 【输出参数】 无
 - 【调用权限】 public
- 4、函数名： OnBnClickedCheck2()
 - 【函数原型】 void Sys::OnBnClickedCheck2()
 - 【接口说明】 下班考勤操作
 - 【输入参数】 无
 - 【输出参数】 无
 - 【调用权限】 public
- 5、函数名： OnBnClickedQuery()
 - 【函数原型】 void Sys::OnBnClickedQuery()
 - 【接口说明】 考勤查询
 - 【输入参数】 无
 - 【输出参数】 无
 - 【调用权限】 public

4.3、和服务器交互的数据包的操作

- 1、函数名： MsgToParser()
 - 【函数原型】 int MsgToParser(ParserPack *msg, char *pdu)
 - 【接口说明】 将数据类型转化为字节数组
 - 【输入参数】 *msg, *pdu

【输出参数】int -1 为错误, 0 为正确

【调用权限】public

2、函数名: ParserToMsg()

【函数原型】int ParserToMsg(char *pdu, ParserPack *msg)

【接口说明】将字节数组转化为数据类型

【输入参数】*pdu, *msg

【输出参数】int -1 为错误, 0 为正确

【调用权限】public

3、函数名: GetM()

【函数原型】char* GetM(CString cstr)

【接口说明】将 CString 转化为 char*

【输入参数】cstr

【输出参数】char*

【调用权限】public

4、函数名: SendPic()

【函数原型】void SendPic(CString PathName, CSocket &socket,
ParserPack *msg)

【接口说明】传输图片函数

【输入参数】图片路径, socket, *msg

【输出参数】无

【调用权限】public

4.4、客户端的图像处理

1、函数名: ViewFace()

【函数原型】afx_msg void ViewFace()

【接口说明】显示人脸即时视频

【输入参数】无

【输出参数】输出窗口显示摄像头捕捉的图像

【调用权限】public

2、函数名: OutputFrame()

【函数原型】afx_msg void OutputFrame(IplImage *frame)

【接口说明】显示 frame 图像

【输入参数】IplImage *frame 指向二维图像的指针

【输出参数】在制定窗口输出某一帧图像, frame 指向该图像帧

【调用权限】public

3、函数名: FaceDetect()

【函数原型】afx_msg int FaceDetect(IplImage *pframe)

【接口说明】在图像中找寻人脸区域

【输入参数】IplImage *frame 指向二维图像的指针

【输出参数】返回值大于 0 表示检测成功。该函数改变 3 个全局变量：人脸中心横坐标 face_x、人脸中心纵坐标 face_y、人脸半径 face_r, 通过这三个变量确定人脸区域

【调用权限】public

4、函数名：RGB2Gray ()

【函数原型】afx_msg void RGB2Gray(IplImage *pframe)

【接口说明】将 pframe 所指向的图像转换为灰度图像，并且进行中值滤波

【输入参数】IplImage *frame 指向二维图像的指针

【输出参数】无

【调用权限】public

5、函数名：Histeq ()

【函数原型】Histeq(IplImage *pframe, int h, int w)

【接口说明】对 pframe 所指向的图像进行直方图均衡化

【输入参数】IplImage *frame 指向二维图像的指针，h 为图像高度，w 为图像宽度

【输出参数】无

【调用权限】public

6、函数名：ImResize ()

【函数原型】ImResize (IplImage *pframe, int h, int w)

【接口说明】对 pframe 所指向的图像调整大小

【输入参数】IplImage *frame 指向二维图像的指针，h 为调整后图像高度，w 为调整后图像宽度

【输出参数】无

【调用权限】public

7、函数名：CreateGray()

【函数原型】afx_msg void CreateGray()

【接口说明】初始化灰度矩阵

【输入参数】无

【输出参数】无

【调用权限】public

8、函数名：CreateRGB()

【函数原型】afx_msg void CreateRGB()

【接口说明】初始化 RGB 像素矩阵

【输入参数】无

【输出参数】无

【调用权限】public