

# 基于机器学习的自动点名系统

## 基于机器学习的自动点名系统

- 1.初定实现方案
  - 1.1数据来源(Input)
  - 1.2数据处理
  - 1.3结果输出(Output)
  - 1.4前端交互
- 2.开发语言、工具选用
  - 2.1开发环境
  - 2.2图像识别工具
  - 2.3开发语言
- 3.开发流程
  - 3.1开发清单

## 1.初定实现方案

### 1.1数据来源(Input)

实现的自动点名系统的数据来源有2个：

- 1、将静态照片(.jpg等格式)作为初始数据输入到系统中
- 2、将一个视频片段作为初始数据输入到系统中

论述：

本质上，对视频片段的处理也是对静态图片的处理。相比于直接使用静态图片进行处理，使用视频需要先按照时间顺序抽出几帧图片，对各个图片分别进行处理得出结果后进行汇总。

### 1.2数据处理

数据的处理方案主要有2个：

- 1、使用当前各AI平台开放的人物图像提取API
- 2、使用OpenCV进行手动切割提取

将各人脸图像从图片中提取出来之后，使用API进行其于数据库中的人脸图像进行比对，最终得出结果。

论述：

- 1、从人脸图像从相片中提取出来这部分功能，目前暂时只能通过封装好的API或OpenCV中自带的模型进行提取
- 2、人脸图像比对可以使用开源的API实现，也可以使用自己实现的图像比对算法实现

## 1.3结果输出(Output)

服务器后端通过提取人脸图像并比对，得出每个人脸匹配的相似度结果并得出一个相对可靠的人数。

前端可通过动态渲染将人数和相似度信息直接显示出来，也可将其显示在新的页面。

## 1.4前端交互

前端选用微信小程序，用户通过微信小程序启动点名程序并通过调用移动设备的摄像头拍摄需要点名的相片或视频。小程序获取到相片或视频后将其发送到服务器后端，服务器计算出人数和人脸相似度结果之后返回微信小程序，小程序将结果信息展现给用户。

论述：

- 1、为了有更好的用户体验，应尽量减少小程序与服务端HTTP交互的数据量。当用户选择拍摄视频时，小程序可以在本地将视频分割后提取几张相片，再发往服务端。
- 2、计算比对人脸相似度的过程由于人数多可能耗时较长，所以可以分批返回比对结果。另外，统计人数的重要性大于人脸相似度的重要性，优先返回人数值。

## 2.开发语言、工具选用

### 2.1开发环境

**操作系统：**Windows 10(64bit)、Ubuntu 16.04

**IDE：**IntelliJ、PyCharm

### 2.2图像识别工具

**AI平台：**阿里AI、百度AI、旷视Face++、腾讯AI(效果不太好)

**开源工具包：**OpenCV、dlib

### 2.3开发语言

Python 3、Java、JavaScript、CSS

## 3.开发流程

### 3.1开发清单

