

# 基于深度学习的移动端空中手写数字识别



安卓AI小分队

# MENU

---

1/ 概述

2/ 技术组成

3/ 系统构成

4/ 总 结

# 概述

# 1

PART 1

1.1 项目目标

1.2 应用需求

# 1.1 项目目标

---

01

线性加速度传感器

手机传感器，采集空中手写数字的加速度数据

02

深度学习的方法

建立模型用于空中手写数字识别，准确率达到99%

03

初步应用

将模型初步进行应用，已实现人机验证、紧急求助这两个应用

## 1.2 应用需求

### 人机验证

验证码可  
票、抢座、灌  
特定用户或网  
击。传统验证

### 紧急情况

多起  
人可能会监视  
可能会引起嫌  
地情况下向外



力破解密码、刷  
意用户对某些  
不断的非法攻  
挑战。



嫌疑  
清? 机的敏感操作  
引起嫌疑人警觉  
重点。

# 技术组成

## PART 2

2.1 数据采集



2.2 数字分割



2.3 数字识别

## 2.1 数据采集

---

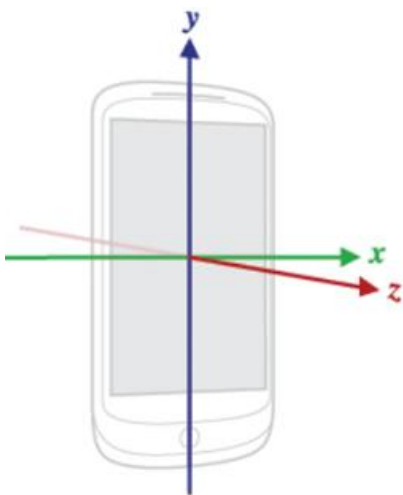


图2-1 线性加速度



图2-2 竖直持手机

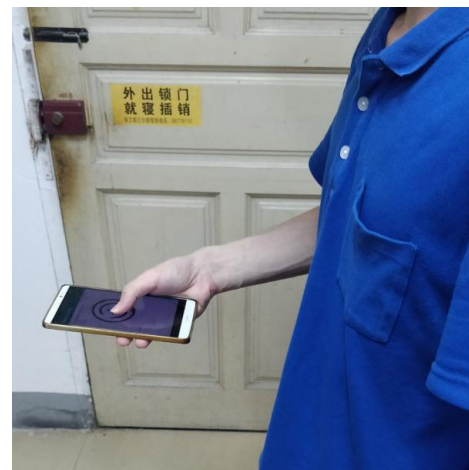


图2-3 水平持手机

## 2.1 数据采集

---

数据集内容	数字个数
竖直持手机数字数据库	16740
水平持手机数据库	15390

表2-1 训练数据集

传感器采样频率：100Hz

每10个数字长度：约2000个采样点



## 2.1 数据采集

---

书写数字的笔画有很多种，采取主流的书写笔画

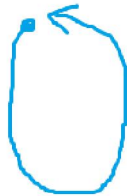

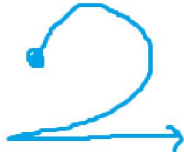

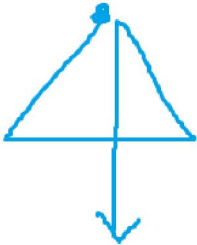


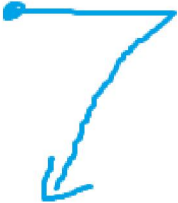


数字 0	数字 1	数字 2	数字 3	数字 4
				
数字 5	数字 6	数字 7	数字 8	数字 9
				

表2-2 数字笔画表

## 2.1 数据采集

---

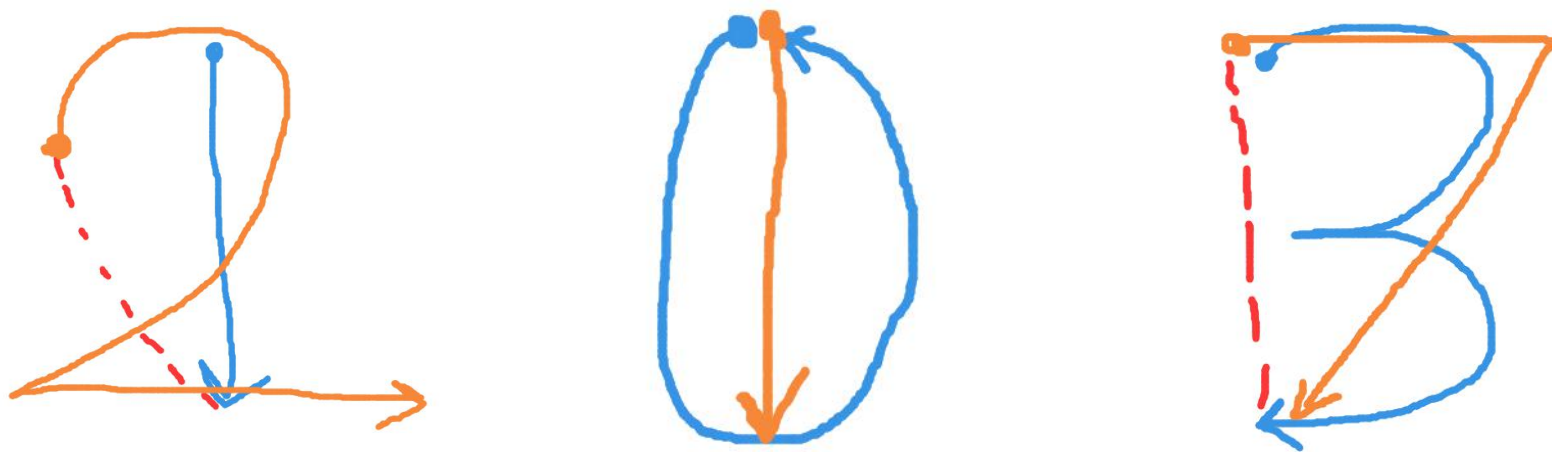


图2-4 连续书写数字的连笔

## 2.1 数据采集

书写多个数字时，数字和数字间会存在连笔，在数据库中包含连笔信息。数字连笔本有 $10*10=100$ 种，按照笔画起止位置进行归类，只需要采集 $3*6=18$ 种。

开始位置	结束位置
	中下： 1 4
正上： 0 1 4 5	右下： 2
左上： 2 3 7	左下： 3 6 7 9
正中： 8 9	右上： 5
	正中： 8
	正上： 0

表2-3 数字开始和结束位置

## 2.2 数字分割

---

01

### 目标

将连续书写的几个数字分割成单个数字段

02

### 难点

1. 只凭加速度数据，难以判断何时在书写数字
2. 用户书写用户手抖以及传感器噪声带来的影响
3. 数字分割不成功，很容易降低后期的数字识别准确率

## 2.2 数字分割

数字分割方法	具体实现	效果
能量阈值	类似音频中的能量阈值划分	连续书写分割后识别准确率85%
移动方差阈值	单/双阈值	难以确定阈值，无法区分
RNN/LSTM	在训练集上达到1的准确率	无法准确识别静止段
SVM	将能量、移动方差、加速度变化率、加速度均值作特征	连续书写分割后识别准确率82%

表2-4 数字分割方法

## 2.2 数字分割

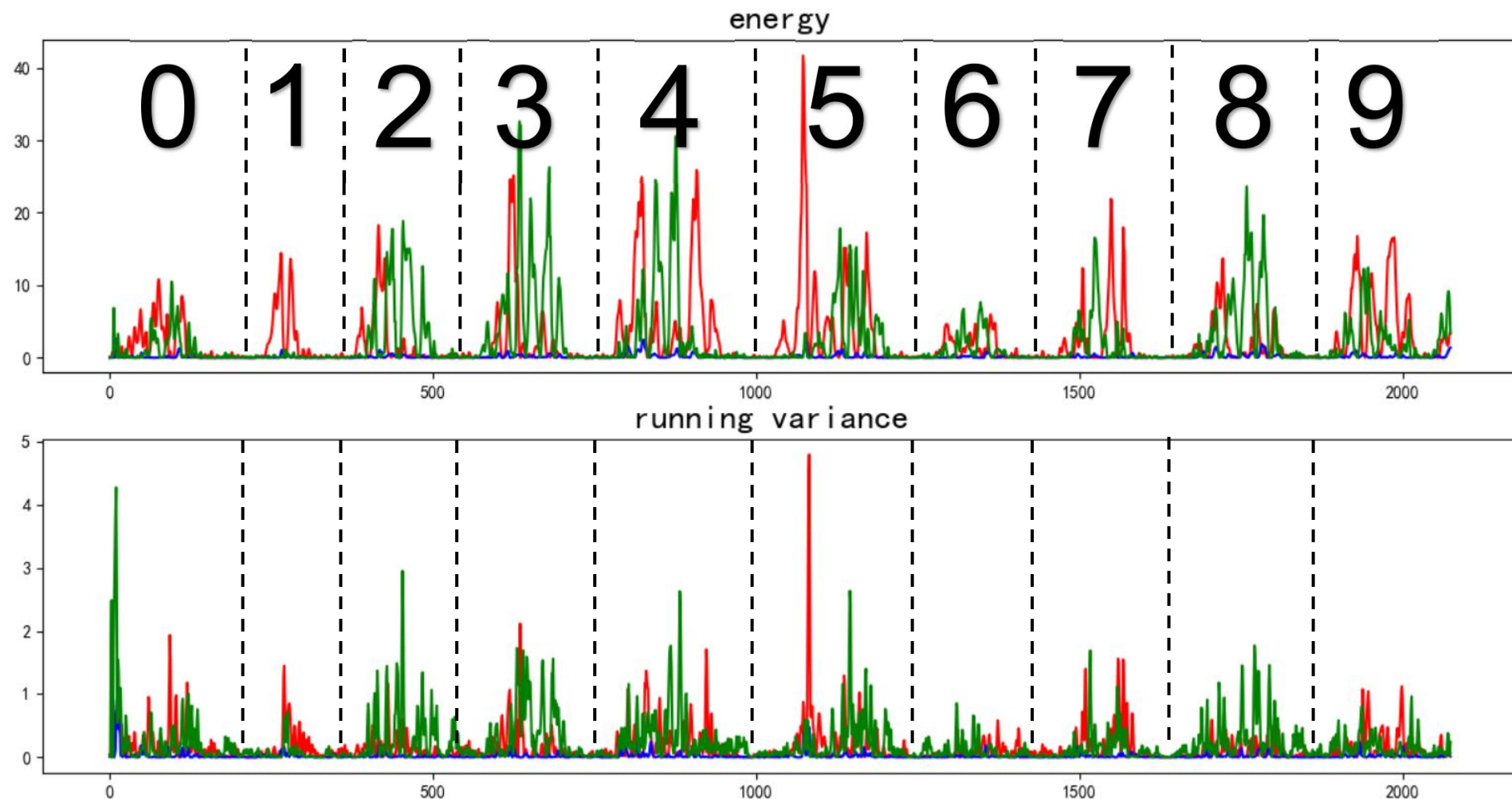


图2-5 数据中能量和移动方差

## 2.2 数字分割

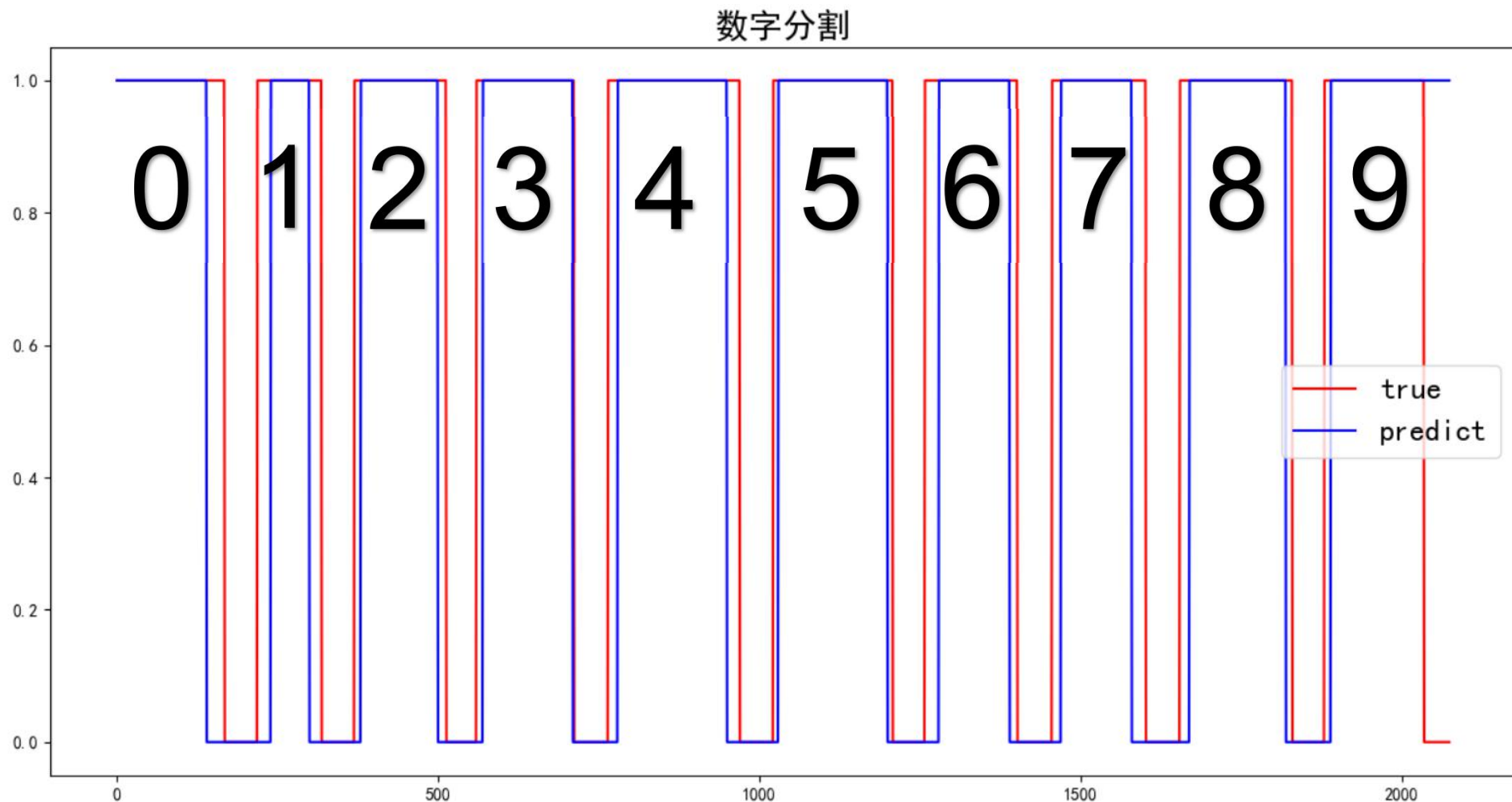


图2-6 分割后的数据

## 2.3 数字识别

---

识别方法	准确率
RNN	90.4%
LSTM	91.8%
CNN	95.3%
CNN-BLSTM	99.4%

表2-5 数字识别及准确率



## 2.3 数字识别

---

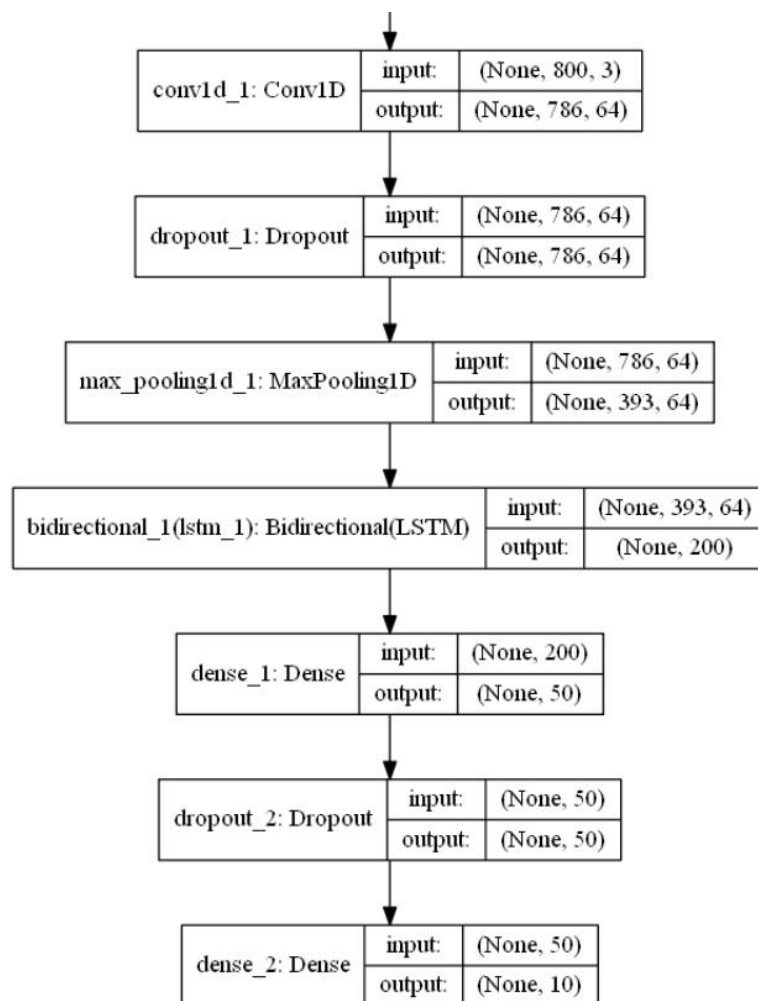


图2-7 模型结构

## 2.3 数字识别

### ● CNN-BLSTM模型训练

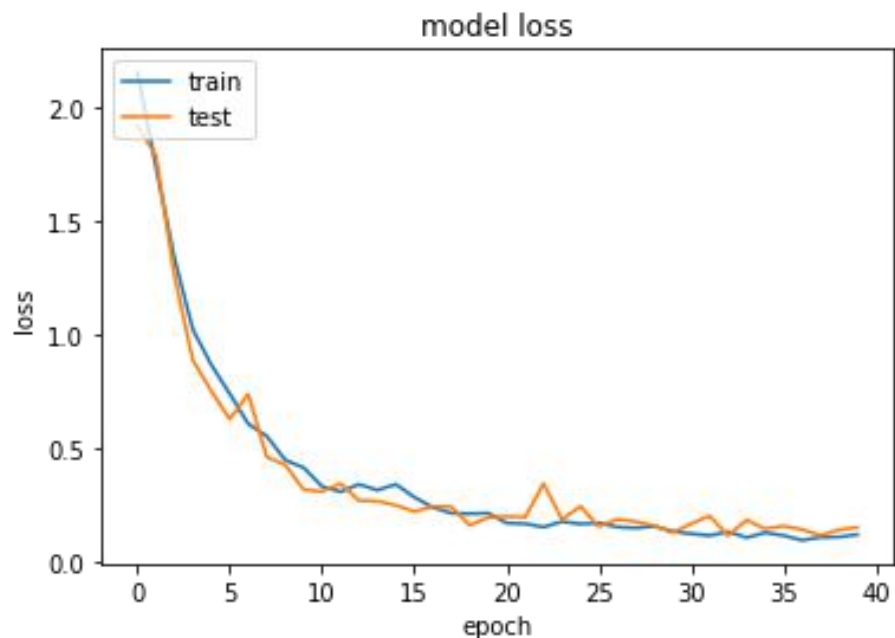


图2-8 训练集和验证集损失

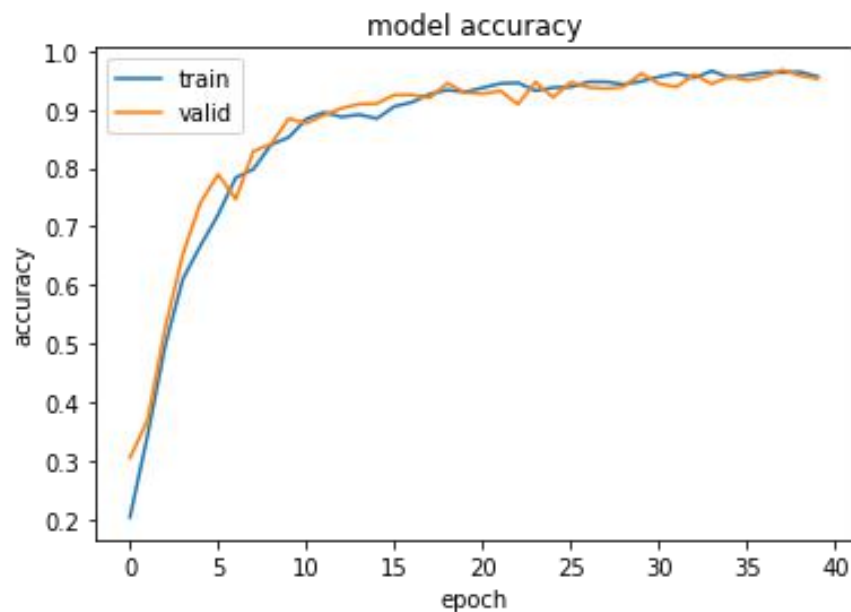


图2-9 训练集和验证集准确率

## 2.3 数字识别

### ● CNN-BLSTM模型训练

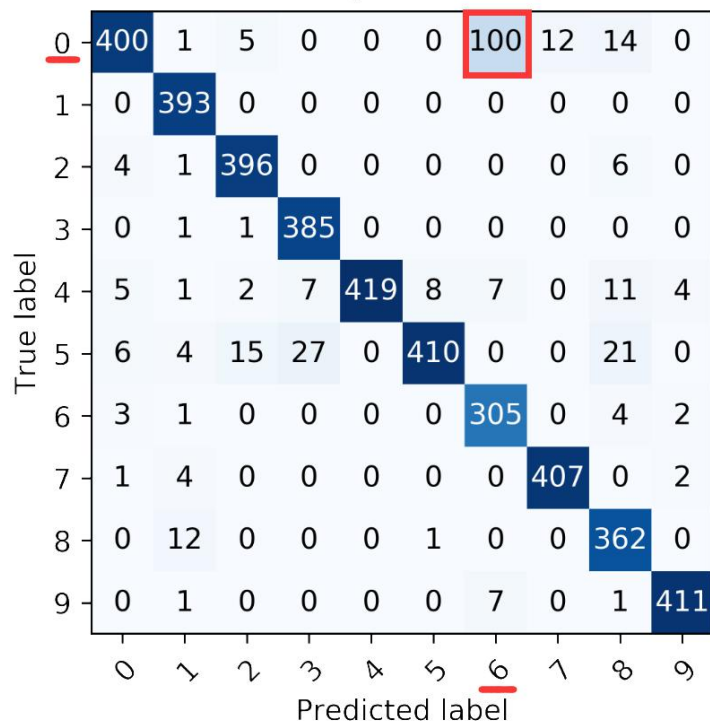


图2-10 0与6分辨不清

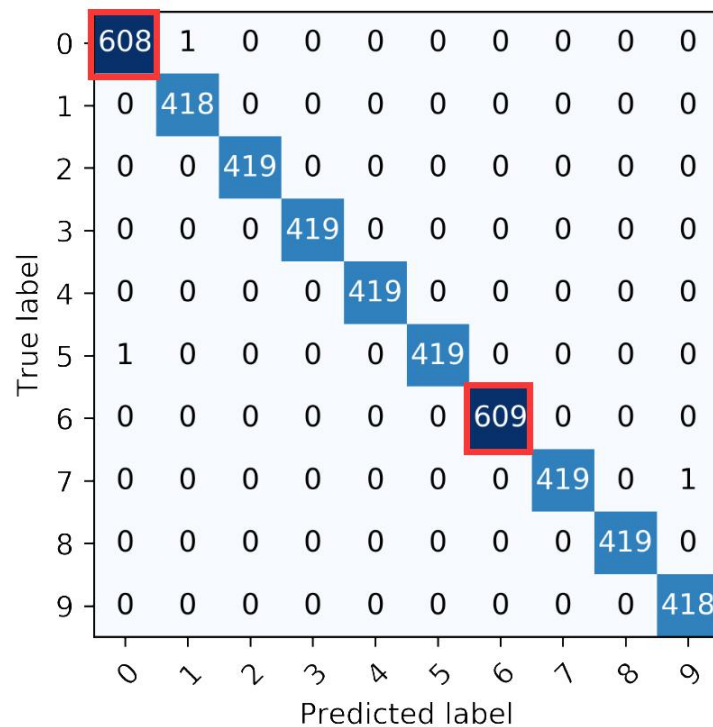


图2-11 0与6分辨清晰

## 2.3 数字识别

### ● CNN-BLSTM模型测试

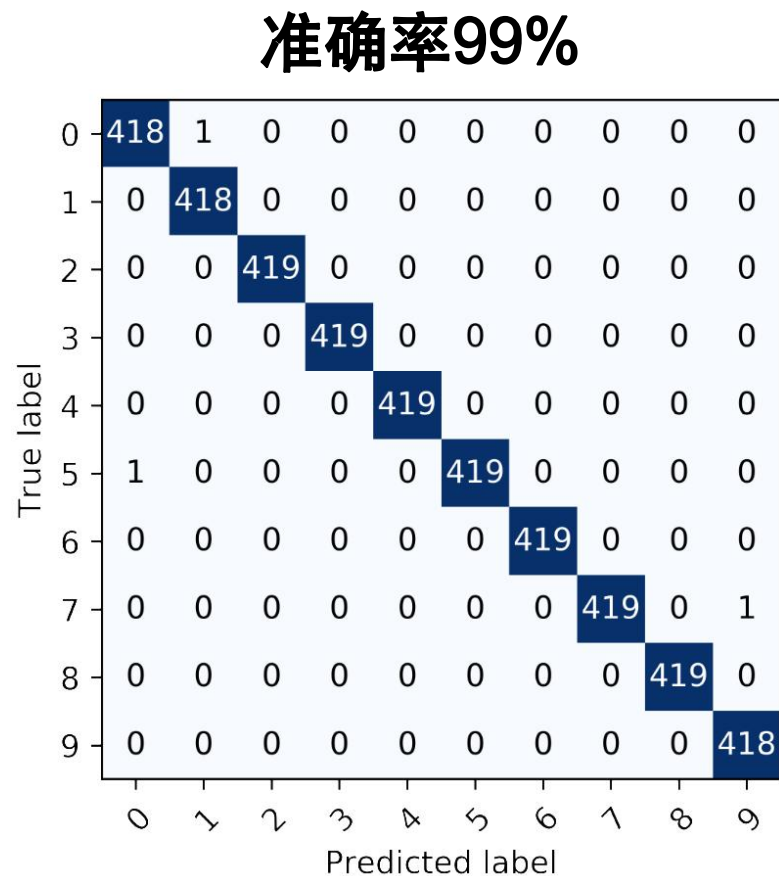


图2-12 测试集上混淆矩阵

## 2.3 数字识别

---

### ● 可视化数字

#### ➤ 目标

- 可以提升用户体验，用户可以看到自己数字轨迹；
- 可视化效果进一步提升时，可以直接对可视化数字进行识别

#### ➤ 问题

- 将加速度二次积分得到位移，不是太准确

#### ➤ 解决方案

- 对三维轨迹使用PCA进行处理，之后再投影到二维平面上

## 2.3 数字识别

### ● 可视化数字

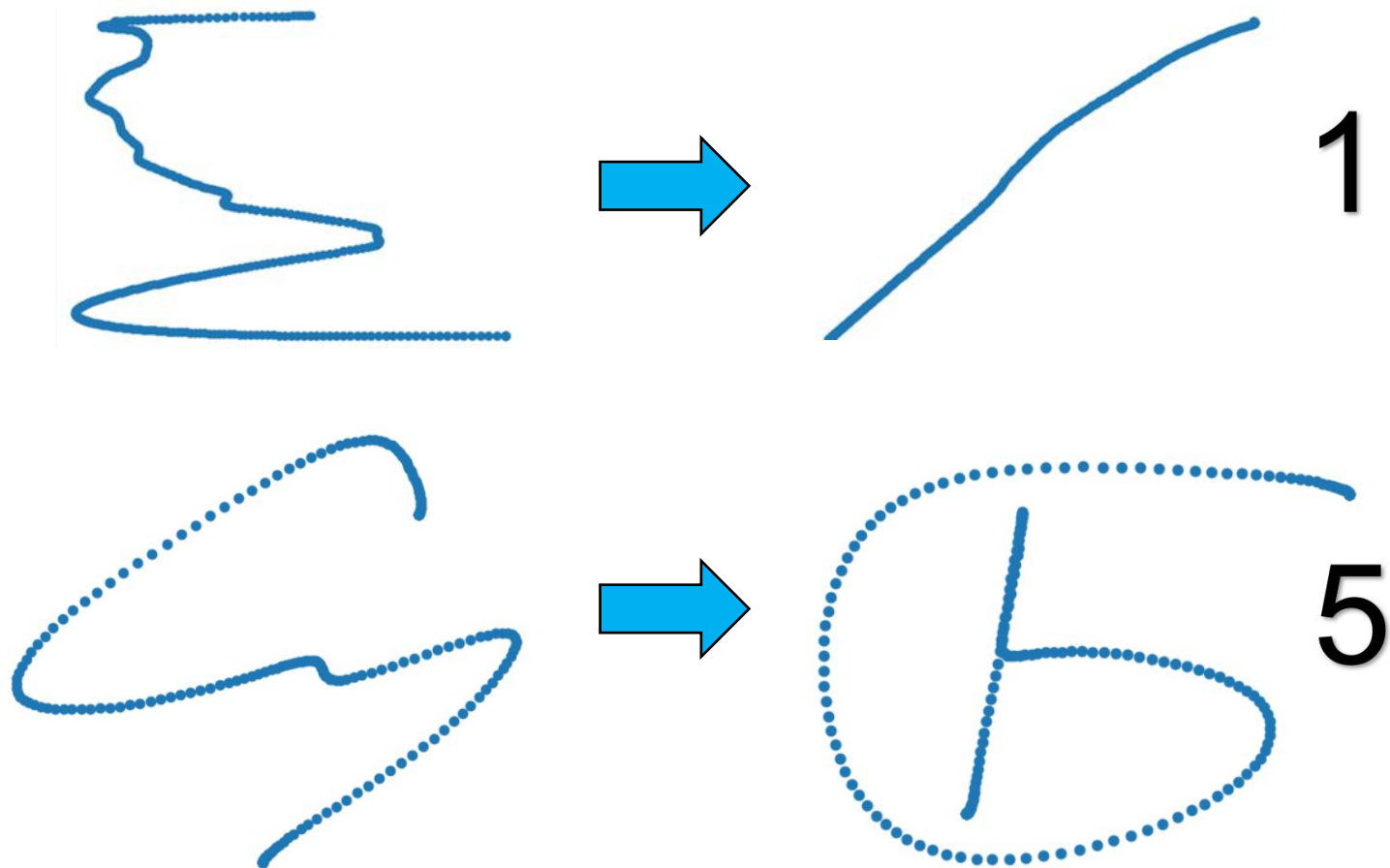


图2-13 数字可视化(左为原图, 右为PCA处理后)

# 系统构成

## 3 PART 3

3.1 应用架构

3.2 软件开发

## 3.1 应用架构

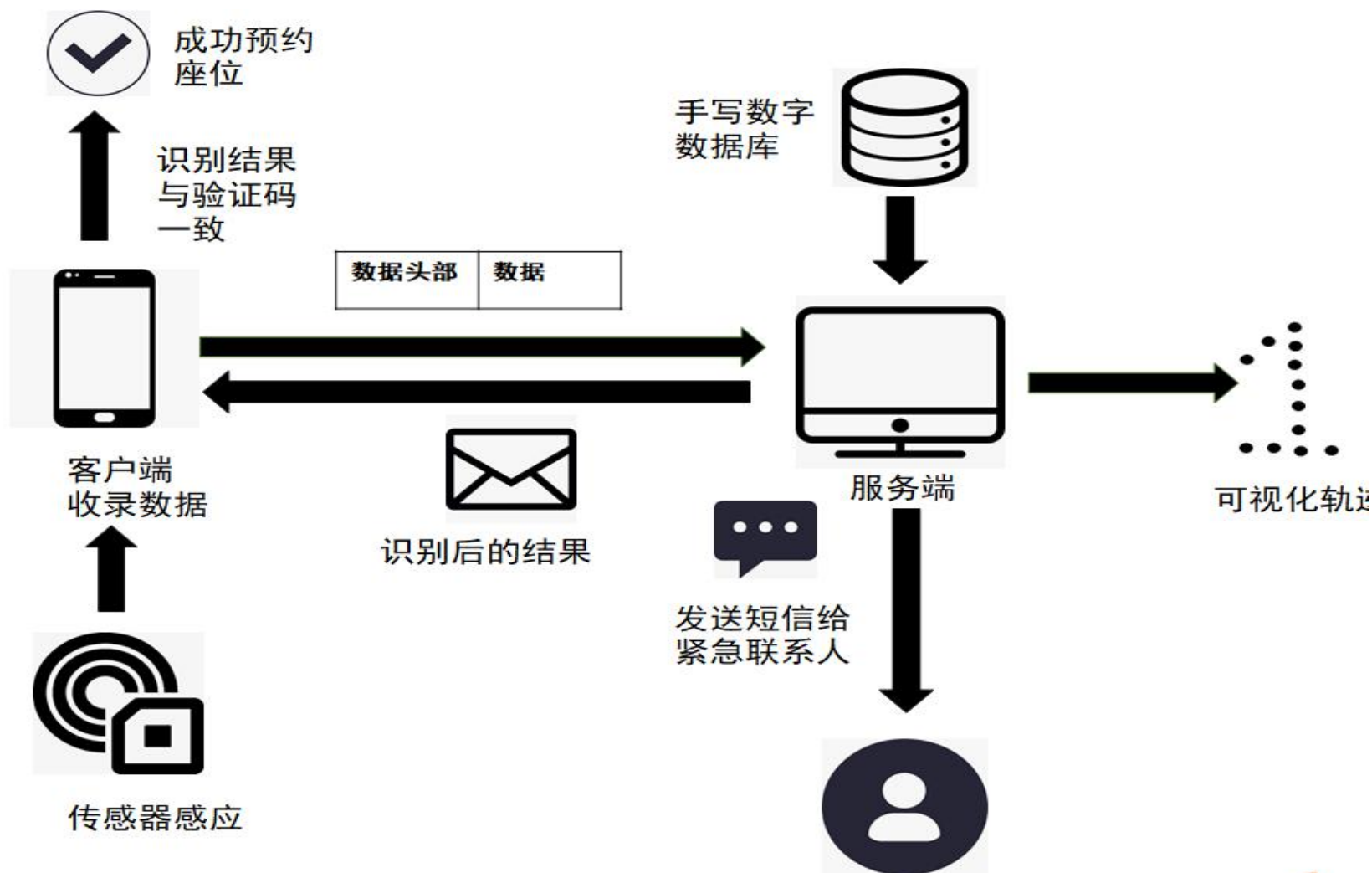


图3-1 应用架构图



## 3.2 软件开发

### A. 座位预定场景

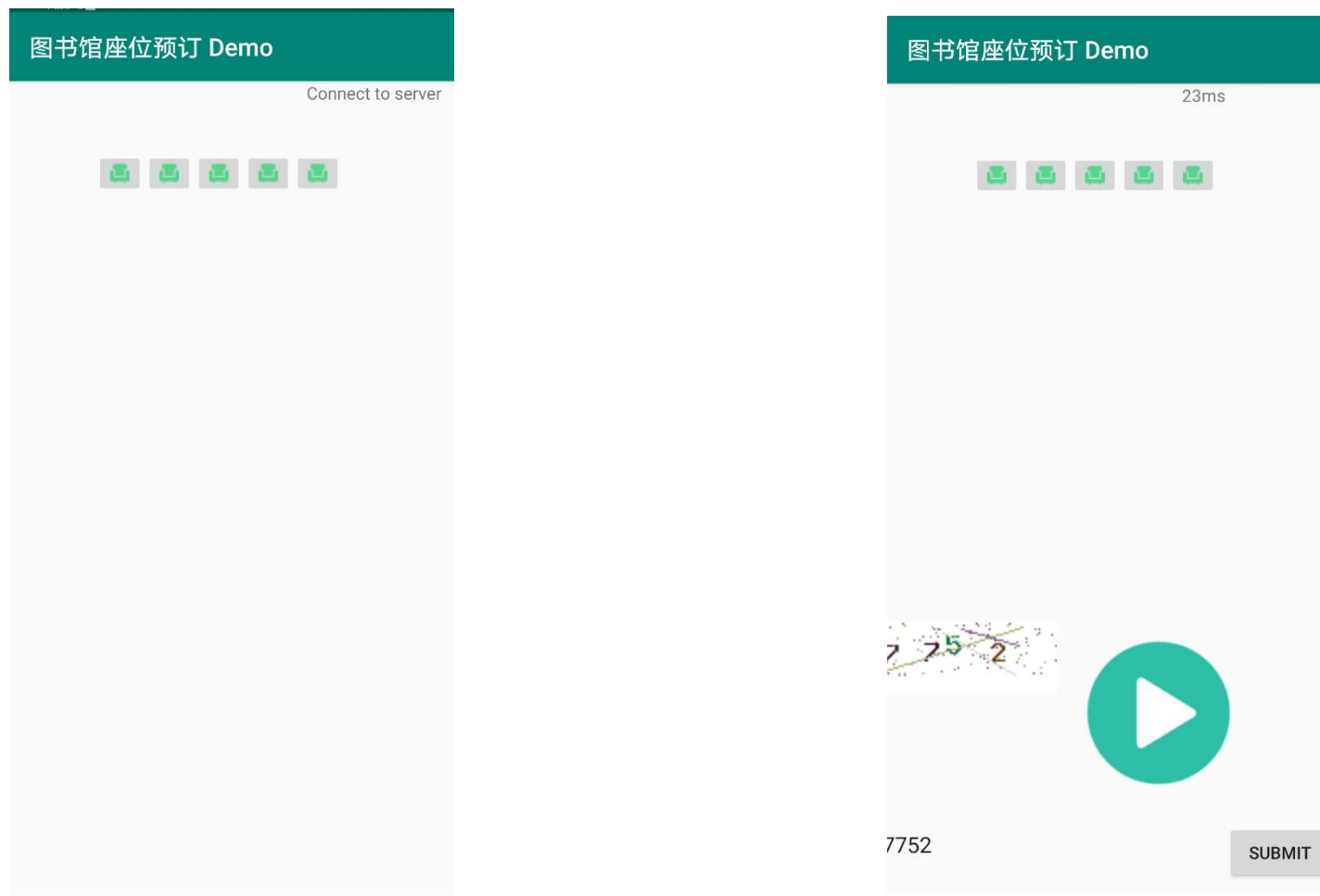


图3-2 客户端运行时界面

## 3.2 软件开发

### B. 紧急求助场景

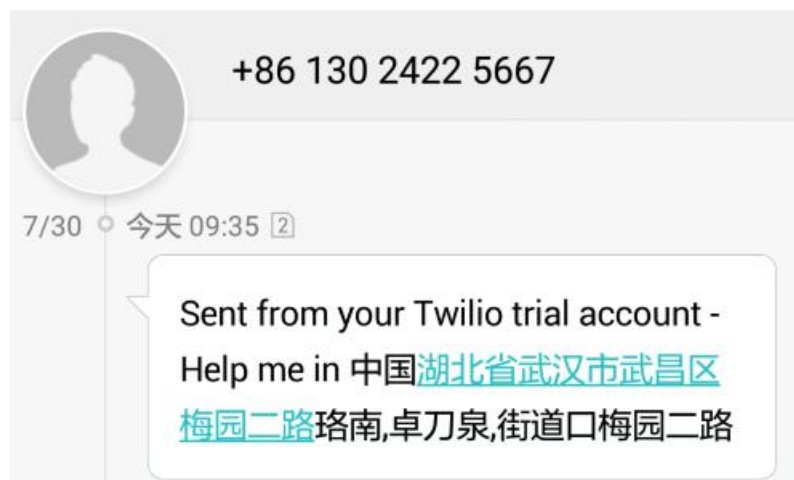
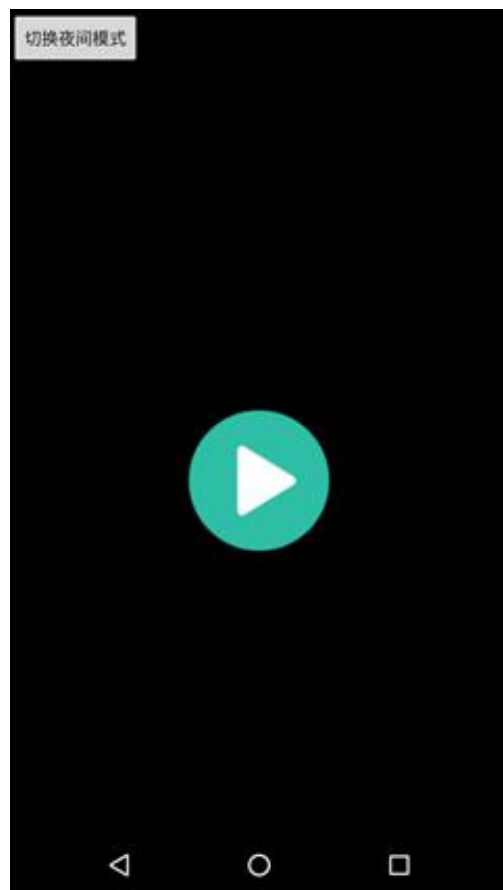
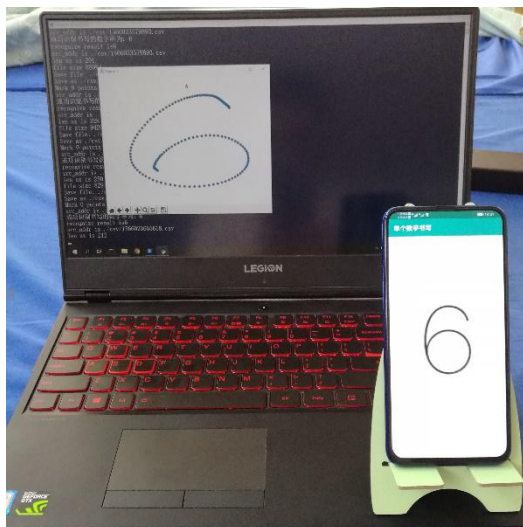


图3-3 客户端运行时界面

## 3.2 软件开发



服务器端的功能为接受客户端传来的数据并进行处理，再调用识别数字的函数进行数字的识别，最后返回给客户端。

并在必要时向用户的紧急联系人发送求助短信。

短信中包含时间、精确的地点、经纬度等信息，我们在App中获取到了经纬度

	HWTNY
时间	2019-08-16 23:43:22
电话号码	+8616602755946
地点	中国上海市闵行区东川路2380江川路东川路
经纬度	(31.013552,121.405106)
紧急联系人电话	+8613110454286

图3-4 服务器模块

# 4

PART 4

## 总结

4.1 应用测试

4.2 创新点

4.3 未来工作

## 4.1 应用测试[]总结，不完全是测试

---

### ● 不同人空中书写数字准确率

#### ➤ 可能存在问题

- 数据集由组员收集，应用时其它用户书写准确率可能不高

#### ➤ 测试结果

- 经不同用户使用测试，总的准确率为

## 4.2 创新点

---

### ● 空中手写数字，验证码书写、VR设备？

#### ➤ 可能存在问题

- 数据集由组员收集，应用时其它用户书写准确率可能不高

#### ➤ 测试结果

- 经不同用户使用测试，总的准确率为

## 4.3 未来工作

---

### ● 空中手写数字，验证码书写、VR设备？

#### ➤ 可能存在问题

- 数据集由组员收集，应用时其它用户书写准确率可能不高

#### ➤ 测试结果

- 经不同用户使用测试，总的准确率为

## 4.1 提升数字分割准确率

---

### ● 优化数字分割算法，提高分割准确率

#### ➤ 问题

- 分割结果影响识别准确率，目前分割后识别准确率约85%

#### ➤ 解决方案

- 参照最新的VAD方法，调整系统结构，使用端对端模型，进一步提高准确率



## 4.2 减轻服务器负担

---

### ● 减轻服务器负担

#### ➤ 问题

- 每个数字文件大小约20KB，服务器难以承受流量压力

#### ➤ 解决方案

- 识别实现在客户端，使用 TensorFlow Lite进行实现

## 4.3 迁移学习

---

### ● 提高不同人、不同书写姿势的准确率

#### ➤ 问题

- 数据库中写的姿势、风格有限，新用户识别率并不高

#### ➤ 解决方案

- 数据库中收录更多数字笔画
- 采用迁移学习的思路，新用户使用时，书写10个数字若干次，更新模型参数

## 4.4 丰富应用场景

---

### ● 可在多种设备和场景中扩展

#### ➤ 设备扩展

- 智能手表
- 智能手环
- 其它含有加速度和重力传感器设备

#### ➤ 场景扩展

- 用于输入密码，同时进行身份认证
- 在使用VR设备时输入数字

**Thanks**