

上海海洋大学

毕业设计（论文）

（2019 届本科）

题目：基于 AR 的工业数据交互系统设计与研究

学院：信息学院

专业：计算机科学与技术

班级：15 级计科 3 班

姓名：杨嘉豪

学号：1551336

指导教师：杨蒙召 朱晨彬

2019 年 4 月

目 录

1 绪论	1
1.1 工业智能互联网发展背景介绍	1
1.2 设计与研究的目的与意义	1
1.3 本文工作内容安排	2
2 系统需求分析与总体设计	3
2.1 需求分析	3
2.2 模块设计	3
2.3 系统流程	4
2.4 技术架构	5
2.4.1 硬件架构	5
2.4.2 软件架构	5
3 工业自动化数据采集装置设计	7
3.1 自动化采集装置模块的架构设计	7
3.2 LoRaWAN 采集协议分析	9
3.3 云端数据库与数据接口	11
3.3.1 云端数据库	11
3.3.2 接口功能说明	12
4 AR 显示与后台数据库	16
4.1 二维码原理分析	16
4.2 ARCore 工作原理分析	16
4.2.1 运动跟踪	16
4.2.2 环境理解	17
4.2.3 射线检测	17
4.3 MongoDB 数据库	18
4.4 Unity3D 引擎	18
4.3.1 搭建 AR 环境	18
4.3.2 二维码扫描	19
4.3.3 寻找平面	20
4.3.4 数据显示模型设计	21

4.3.5 用户交互	21
4.3.6 与后台数据库的数据传输	22
5 系统测试与扩展能力分析	24
5.1 实际效果测试	24
5.2 扩展能力分析	27
结论	28
致谢	29
参考文献	30

基于 AR 的工业数据交互系统设计与研究

摘要：基于传统工业生产制造环境中经常由于场地局限或者环境恶劣等影响因素而不能实现人工作业，可能导致工作效率低下且现场作业人员安全得不到保障的问题，需要实现数据自动化采集并可以远程查看。增强现实技术是实时计算摄像机姿态，将虚拟内容叠加到现实世界场景中，使现实世界信息和虚拟信息相结合的显示技术。增强现实技术可以丰富工业数据的显示形式，开辟新的工业作业方式。本文利用二维码技术和增强现实软件开发包 ARCore SDK，结合当前流行的物联网采集模块与非关系型数据库，设计一种基于 AR 的工业数据交互系统，并进行了简单的实现。本系统实现工业作业数据的远程查看与存储，增加工业数据监控的便利性，提高工业生产制造效率；并且系统扩展性强，可为未来探索工业领域与 AR 技术结合提供技术方案。

关键词：增强现实；工业；物联网；ARCore；二维码

Design and research of industry data interaction system based on AR

Abstract : Based on the influence of factors such as site limitation or harsh environment in the traditional industrial manufacturing environment, manual work may not be realized, which may result in low work efficiency and the safety of on-site workers. It is necessary to realize automatic data collection and remote viewing. Augmented reality technology is a display technology that combines real-world information with virtual information by superimposing camera poses in real time and superimposing virtual content into real-world scenes. Augmented reality technology can enrich the display format of industrial data and open up new ways of industrial operation. This paper uses the two-dimensional barcodes technology and augmented reality software development kit ARCore SDK, combined with the current popular IoT acquisition module and non-relational database, design an industrial data interaction system based on AR, and carried out a simple implementation. The system realizes remote viewing and storage of industrial operation data, increases the convenience of industrial data monitoring, improves industrial production and manufacturing efficiency, and has strong system expansion, and can provide technical solutions for exploring the industrial field and AR technology in the future.

Key words: Augmented Reality; Industry; Internet of things; ARCore; two-dimensional barcodes

1 绪论

1.1 工业智能互联网发展背景介绍

工业互联网是从工业发展的技术需求和互联网发展的大潮中产生的一种新型技术驱动方式，通俗易懂地解释就是生产力发展的必然趋势和结果。在近十年来，由于 Android、IOS 等手机智能操作系统的普及和推广，各大厂商的云计算服务能力、大数据平台增加以及关于人工智能的深度学习算法的研究突破，各个行业各个领域的发展格局与实行方式都被极大地改变了。互联网、软件、通讯等很多产业都被重新定义，当这些科技技术向外发展和融入到传统行业时，产生了一个比以往任何时期都要大的趋势，那就是世界经济的网络化、数字化和智能化。

过去的工业体系经过了 3 次变革，第一次工业革命（工业 1.0、机械制造时代）是使用水和蒸汽来代替劳动力，使机械化得到卓越的发展，同时工业生产效率也以几倍甚至几十倍的幅度向上提升。第二次工业革命（工业 2.0、电气化与自动化世代）是使用电力来代替水和蒸汽的驱动方式，大规模的分工合作模式也得到发展，对制造业的影响尤其重大，因为对于制造业动力永远都是最基础的核心要求。第三次革命（工业 3.0、电子信息化时代）是以电子工程与互联网技术为基础，利用自动化生产制造模式来更进一步提高生产制造效率，同时很大地推进了电子商务与科技与生活融合的发展过程。第四次工业革命（工业 4.0、智能化世代）这个概念最早是德国在汉诺威工业博览会应对数字化浪潮和制造业竞争推出的战略选择，目的是通过充分利用信息通信技术和信息-物理系统来引导制造业进行智能化产业转型。

从智能制造角度，工业互联网有 3 种概念：第一种是工业网络；第二种是工业智能化发展的关键网络基础设施、新应用模式和新兴产业；第三种是智能制造，智能制造不是一个从无到有的新概念，而是在现有互联网上发展、扩展和升级而逐步形成的，它在范围、功能和复杂度上将远远超过现在的互联网。工业互联网以互联网为基础，由实际的工业数据驱动，通过人工智能对数据的分析和建模，实现工业领域各产业各个层面的智能化升级，形成一种人、机、物互联的生态体系。

1.2 设计与研究的目的与意义

在工业领域中，经常需要根据各种工业现场设备的运行数据和参数来对工业生产制作过程进行调整，但由于工业环境的多样性与危险性，工业数据可能无法用传统方法由人工测量，需要借助自动化运作的采集仪器来进行工业数据采集。数据采集完后工业作业人员需要进行工业辅助信息的查看和监测，分析数据对工业生产制作过程的影响，进而调整详细的工业作业。

本文拟设计一个将工业数据进行 AR 形式显示的交互系统，打破工业现场环境的局限性，使用自动化采集装置采集工业现场数据并上传至云端，在手机端进行工业数据的 AR 显示，同时把工业数据存储到后台数据库，供工作数据查看或者其他应用扩展。采用本系统，工业人员可以不限地点，

不限时间进行工业数据的查看和监控，提升工业效率，为探索工业领域与 AR 技术结合提供技术方案。

1.3 本文工作内容安排

本文从如下几个方面，对工业现场数据采集、工业 AR 显示及后台数据存储进行了全面的研究。

本文的第 2 章主要围绕工业 AR 系统的系统设计讨论，重点就工业 AR 系统的基本技术架构给出了详细的设计需求分析。

工业自动化数据采集是对数据进行各类数字化量化并完成数据从采集装置向云平台进行传输的单向过程。本文的 3.1 节，详细分析了常见的自动化采集装置模块的结构框架，并在 3.2 节中，围绕本项目所重点使用的广域采集协议进行分析说明。在本文的 3.3 节中，重点围绕数据采集完成后的云存储环节进行说明，其中包括：云端数据库介绍以及数据访问接口的详细说明。

在第 4 章中，本文分析说明二维码技术原理和 ARCore 实现原理，并进一步对工业 AR 系统的软件实现部分进行了简单的实现，探索了工业数据与 AR 显示效果结合的方式，包括使用 Unity3D 引擎对工业 AR 应用的详细实现，还有后台数据库的设计等内容。

第 5 章就整个系统实际运行效果进行测试和对本系统的扩展能力上限进行分析。

2 系统需求分析与总体设计

2.1 需求分析

传统工业生产制造过程中所用的工业数据都需要工业作业者配合仪器进行人工测量，并且人工测量的标准和方式不统一，人工测量工业数据准确率达到很高标准且耗费人力物力，这种场景下就需要可以以一种数据确认率高的自动采集方式代替人工测量。工业数据监控管理人员需要一直在工业现场读取采集装置的工业数据，但可能由于现场工业生产制造环境恶劣，长期以往对工业数据监控管理人员的身体健康造成不良影响，若可以实现远程地监控管理工业数据，就不需把工业数据监控管理人员派驻在工业现场。工业数据通常是以表格文字的形式供监控管理人员查看，对于工业人员是抽象且易于疲劳的，需要以一种更加形象与有趣的方式来进行工业数据的监控查看管理。在工业生产制造过程中的工业数据需要进行记录存储，且工业生产制造过程中不同环节的工业数据类型可能都是不相同的，需要一个环节的操作需要引用其它环节的不同类型的数据进行参考的，工业生产制造过程中也需要时常更换设备，这就对工业数据监测系统运行的灵活性要求很高。

经过以上分析，本系统需要实现的内容包括：

- （1） 工业数据使用自动化采集代替人工测量；
- （2） 工业数据采集装置可灵活更换；
- （3） 远程查看监控工业数据；
- （4） 工业数据以 AR 形式显示；
- （5） 工业数据需要保存至本地服务器以供分析或扩展。

所以系统共分为三大部分进行设计与实现，第一个部分为工业数据的自动化采集，第二个部分为远程工业数据 AR 显示查看，第三个部分为对工业数据的本地化存储。

2.2 模块设计

系统总体分为工业数据采集、AR 显示和后台数据库三个模块，每个模块的功能不同，这其中包括数据采集、云端存储、二维码扫描、数据读取、寻找平面、用户交互、数据显示、数据存储等。

- （1） 工业采集模块分为：

- ① 数据采集：将自动化数据采集设备放置在有需要的工业环境中定时采集所需要使用的工业数据。
- ② 云端存储：将自动化数据采集采集到的工业数据通过网关或其它设备上传到云端数据库保存。

- （2） AR 显示模块分为：

- ① 二维码扫描：工业用户使用手机扫描含有要读取数据的采集装置硬件型号的二维码图

片，从而指定要查看数据的采集装置。

- ② 数据读取：根据扫描二维码得到的采集装置硬件型号访问云端数据库接口得到工业数据，并进行数据可读化的转化。
- ③ 寻找平面：工业用户移动手机找寻真实世界中的平面并在 AR 程序中显示平面模型，用于实现工业用户的交互操作。
- ④ 用户交互：使用手指单击屏幕中平面模型的某一位置，判断本次单击屏幕位置所对应的 AR 场景中的位置符不符合 AR 显示的要求。
- ⑤ 数据显示：在交互确认完成无误后的 AR 场景中特定位置和角度生成数据显示模型，供工业用户通过手机屏幕查看工业数据的 AR 显示。

(3) 后台数据库模块：

数据存储：存储 AR 显示工作过程中的工业数据，供后期检查工作状态或其它应用调用。

系统总体模块图如图 2-1 所示。

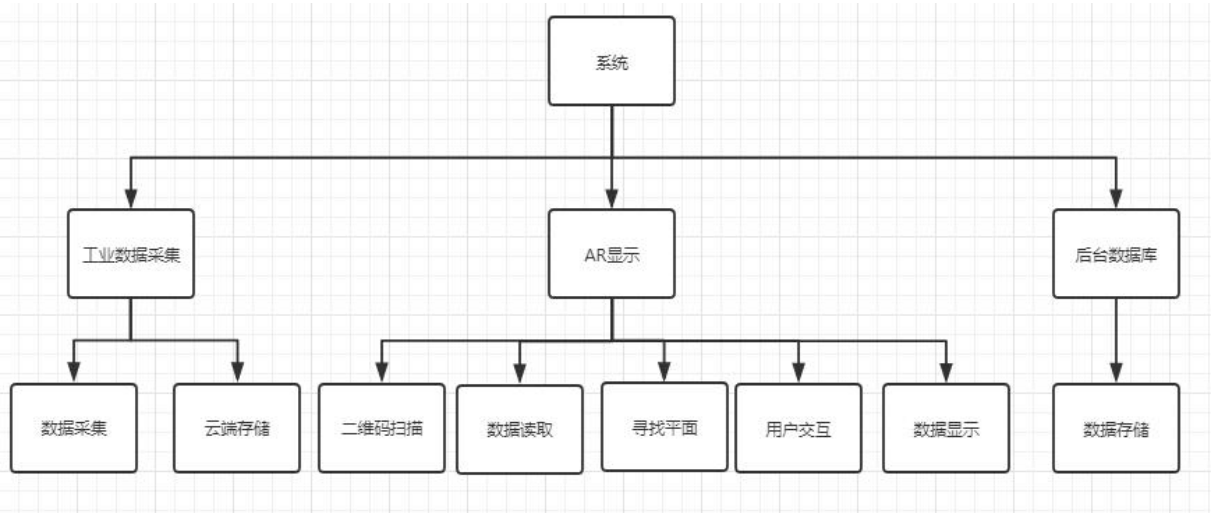


图 2-1 系统模块图

2.3 系统流程

首先需要将工业自动化数据采集装置放置在需要采集工业数据的工业环境中，将工业自动化数据采集装置供电运行后，工业自动化数据采集装置将工业数据采集并存储至云端数据库。远程工业用户打开 AR 程序手持手机进行二维码的扫描,这个二维码包含需要读取数据的采集装置的硬件型号信息，二维码扫描成功后需要扫描周围环境寻找一处平面进行 AR 平面模型网格可视化。当平面可视化完成后，用户需要手持手机对准平面在想要显示数据模型的现实世界对应在屏幕上的位置进行交互操作。交互成功之后程序会在对应的区域显示数据界面模型，并将此次得到的工业数据同步存储到后台数据库中存储。系统流程图如图 2-2 所示。

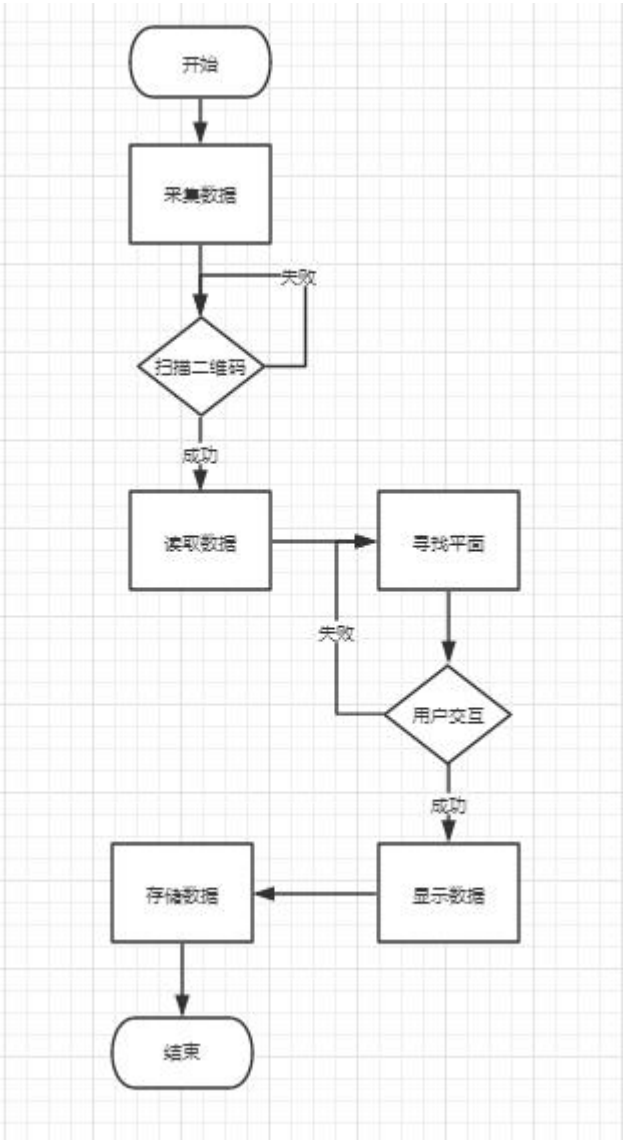


图 2-2 系统流程图

2.4 技术架构

2.4.1 硬件架构

数据自动化采集模块是属于硬件架构，需要实现工业数据无线采集，这一模块的设计与实现使用了一套基于 LoRa 协议的物联网数据采集方案，该方案更换设备方便，设备使用类型的灵活性高。

2.4.2 软件架构

AR 显示和后台数据库这两个模块是属于软件架构。

AR 显示模块的整体实现是在 Unity3D 平台下开发的 Android 程序，提供给支持 ARCore 的 Android 系统手机运行。AR 显示模块包含的技术有二维码技术和增强现实技术。二维码技术作为本系统 AR 显示模块的入口可以实现获取需要查看数据的灵活性，本系统采用的是 Google 推出的用于在 .net 平台上生成或识别二维码的开源项目 ZXing.Net 插件。增强现实技术可以实现虚拟内容与现实世界结合显示的强大显示效果，本系统采用的是 Google 推出的增强现实软件开发包 ARCore SDK，它支持在

Unity3D 游戏引擎中构建供 Android 系统运行的 AR 程序。

后台数据库模块采用的是 MongoDB 数据库。

3 工业自动化数据采集装置设计

现在所有场景用途的物联网架构都是一样的，都是由传感器、通讯网络与云端管理平台所组成的三层架构，先由传感器获取设备数据，再通过通讯网络上传到云端管理平台存储、计算，最后再用分析出来的数据作为物联网系统运行的参考决策。

3.1 自动化采集装置模块的架构设计

使用的自动化采集装置模块由网关设备和传感器设备两部分组成，分别为慧联无限（EasyLinkin）LoRun G300 网关和蓝思科技（lansitec）温湿度传感器，并不局限于这两个型号的设备，可使用任何功能相似的设备进行代替。

LoRun G300 是慧联无限（EasyLinkin）公司旗下的一款网关设备，它基于 LoRaWAN 协议通信，主要用于连接各类标准 LoRaWAN 物联网终端设备，为它们提供功耗极低且安全稳定的双向通信服务。G300 采用星型 LoRa 网络连接本地 LoRa 终端设备，工作频段为 470MHz/868MHz/915MHz/923MHz，再通过以太网或 4G 移动网络的方式连接到网络云服务器，从而实现设备联网的各种物联网应用。LoRun G300 网关实物图如图 3-1 所示。



图 3-1 LoRun G300 实物图

蓝思科技温湿度传感器基于 LoRa 调制方式，工作频段有 EU863-870、EU433、US902-928、CN470-510、AU915-928、AS920-923、AS923-925 和 CLAA，产品内置天线，传输距离可大于 2km。数据采集方面温度测量范围为-40~125℃（精度±0.3℃），湿度测量范围为 0%~99%（精度±3%）。而外壳采用防爆型 ABS 塑料材质，大小尺寸为 85*114*55mm，支持短路保护，且防水等级达到 IP65，可用于多种用途，如气象站、暖气装置、烘干、通风与空调系统、智能农业和建筑自动化。温湿度传感器由 19000mAh 容量的一次性锂电池供电，续航时间与数据报文上传时间间隔有关，按默认的 5 分钟间隔最多可以连续使用五年。蓝思科技温湿度传感器实物图如图 3-2 所示。



图 3-2 温湿度传感器实物图

打开电源开关及加入 LoRa 网络后，传感器会向应用服务器发送注册消息。如果应用服务器没有答覆，传感器在转换成默认模式（周期报告模式）前还会重新尝试发送 5 次注册消息。传感器的配置信息包含在它发送给应用服务器的注册消息中。

在接收到同意入网回应后或者发送 5 次注册尝试后应用服务器没有答覆的情况下，传感器会开始工作。温度信息和湿度信息会周期性地报告给应用服务器。工作循环是由应用服务器的命令配置的。温度信息和湿度信息的接收与报告周期默认是 5 分钟。传感器入网流程图如图 3-3 所示。

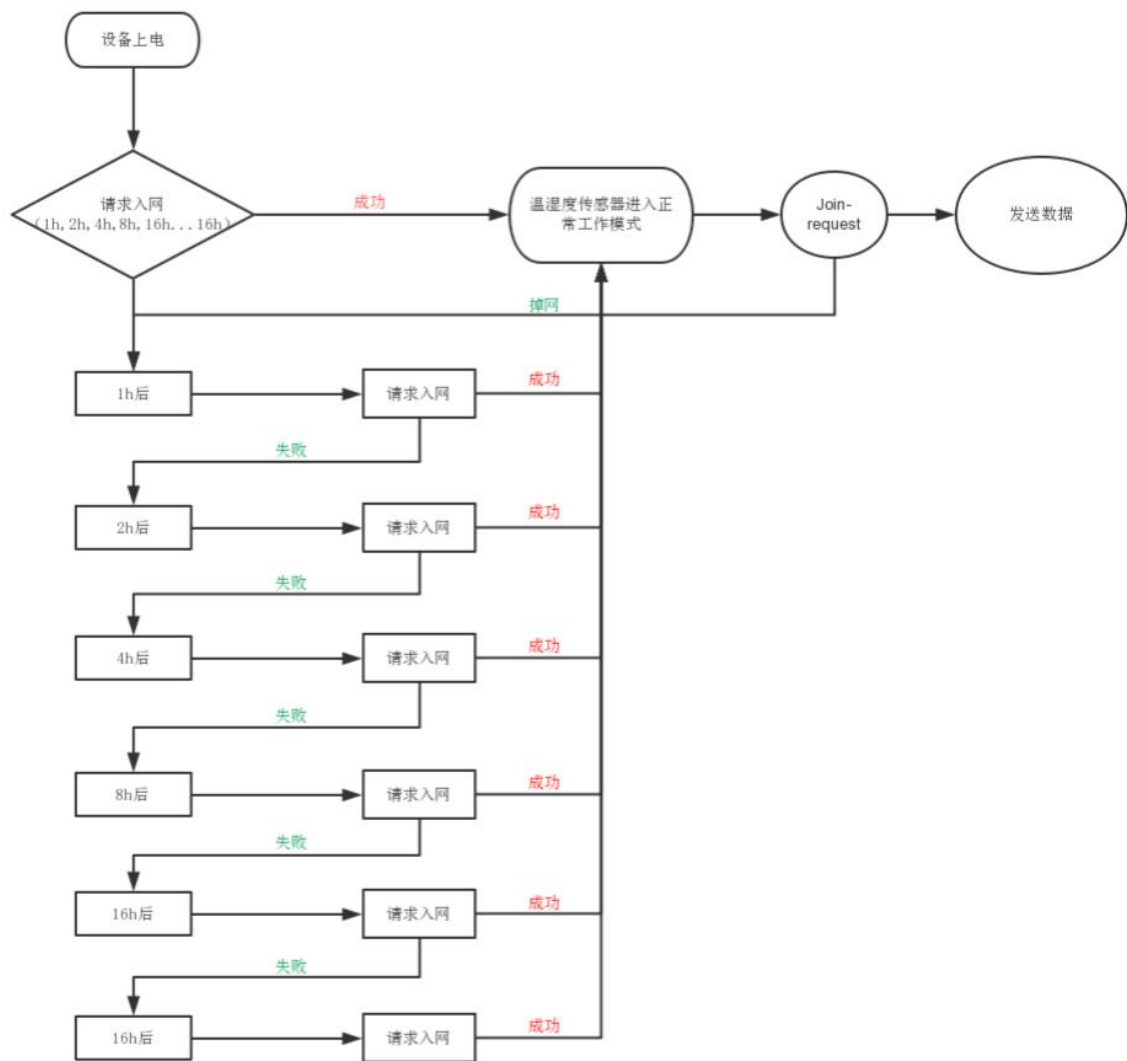


图 3-3 温湿度传感器入网流程图

3.2 LoRaWAN 采集协议分析

蓝思科技温湿度传感器与慧联无限 LoRun G300 网关的通信协议为 LoRaWAN 协议。

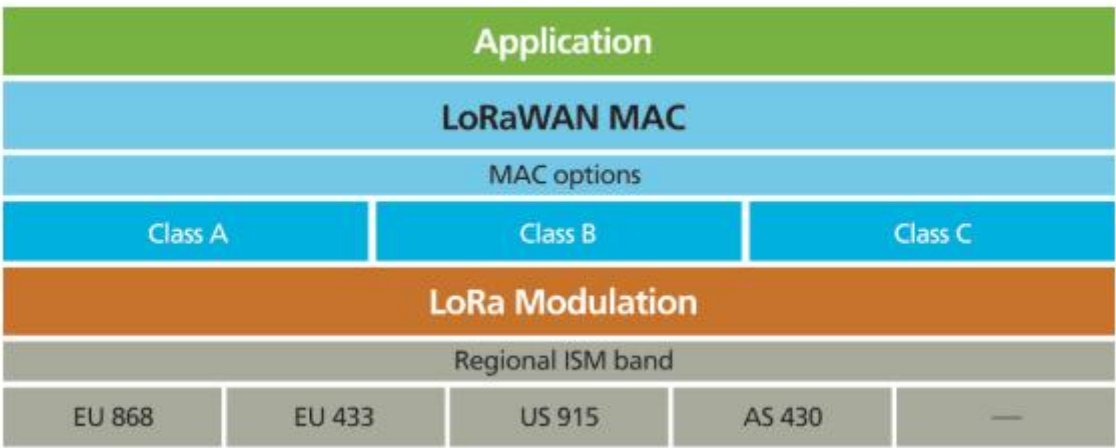


图 3-4 LoRaWAN 协议栈层次图

LoRaWAN 协议栈如图 3-4 所示，LoRaWAN 是 LoRa 的一个超集，LoRa 只定义了通信的物理层协议，而 LoRaWan 在 LoRa 的基础上增加了数据链路控制层（MAC）协议，因此采用 LoRaWAN 协议的终端设备可以将数据发送到任何已经与云平台连接的网关上。

LoRa（long range）是基于线性调频的扩频调制技术，它与 FSK 调制一样都是低功率运行，但相比 FSK 调制大大地增加了通信距离，改变了一直以来关于功耗与传输距离二选一的传统考虑方式，为用户提供一种很简单就能实现远距离、长续航、大容量的系统，从而应用到智能物联网方面。LoRa 网络主要由 LoRa 终端设备、网关设备、应用服务器和云存储几部分组成，应用数据可实现双向传输。

LoRaWAN（LoRa Wide-Area Networks）开放规范是基于 LoRa 技术的低功耗广域网（LPWAN）协议，它定义了通信协议和系统架构，规定使用 LoRa 技术的终端到网关的标准使用规范。

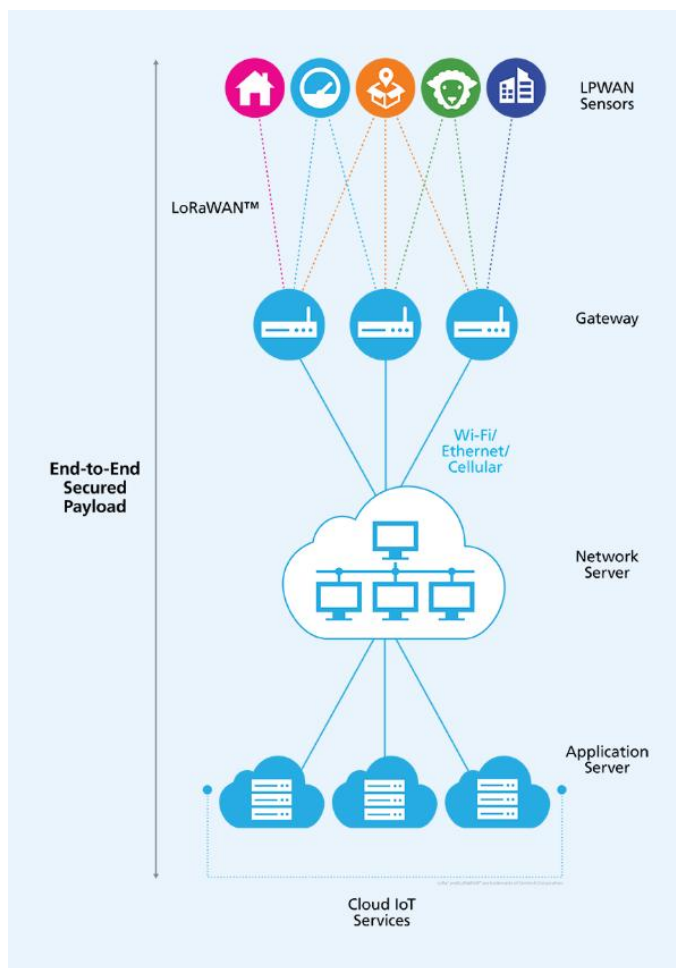


图 3-5 LoRaWAN 网络架构图

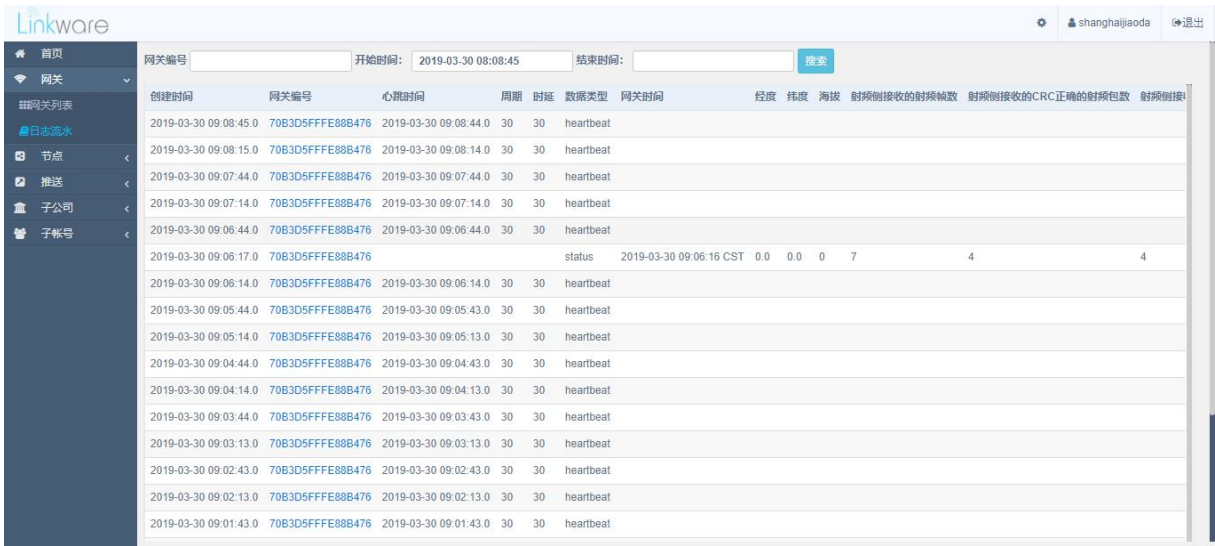
LoRaWAN 网络架构图如图 3-5 所示，最上边是各种 LoRa 终端设备，接着下边是 LoRaWAN 网关，这两者通过网关转换协议把传感器采集到的数据转换为标准的 TCP/IP 格式通过 Wifi/以太网/蜂窝网络发送到互联网服务器上，互联网服务器再经过分析计算，传输给应用服务器以供物联网应用使用。

3.3 云端数据库与数据接口

3.3.1 云端数据库

温湿度传感器采集到数据并通过 LoRaWAN 协议上传到 LoRa 网关后，网关再通过以太网网络连接上传到设备厂商的云端数据库，云端数据库是 EasyLinkin 慧联无限推出的 Linkware 网管云平台（<http://lpwan.net4iot.com/cs/>），在网管云平台上可以查看当前云平台账户下的所有网关信息和节点信息，还可以通过推送下行报文去对网关和传感器进行设置。

本系统中工业自动化数据采集装置的网关设备和传感器设备都需要在 Linkware 网管云平台上先行注册，云平台才会接收网关上发的数据，并提供数据接口。由于慧联无限并没有开放网关设备的注册权限，所以需要联系慧联无限公司的工作人员，让工作人员帮助我们注册网关设备。网关设备注册完成并上电后，可以在网管云平台方便查看网关设备的工作情况。网关日志流水查看界面如图 3-6 所示。



创建时间	网关编号	心跳时间	周期	时延	数据类型	网关时间	经度	纬度	海拔	射频侧接收的射频帧数	射频侧接收的CRC正确的射频帧数	射频帧大小
2019-03-30 09:08:45.0	70B3D5FFFE88B476	2019-03-30 09:08:44.0	30	30	heartbeat							
2019-03-30 09:08:15.0	70B3D5FFFE88B476	2019-03-30 09:08:14.0	30	30	heartbeat							
2019-03-30 09:07:44.0	70B3D5FFFE88B476	2019-03-30 09:07:44.0	30	30	heartbeat							
2019-03-30 09:07:14.0	70B3D5FFFE88B476	2019-03-30 09:07:14.0	30	30	heartbeat							
2019-03-30 09:06:44.0	70B3D5FFFE88B476	2019-03-30 09:06:44.0	30	30	heartbeat							
2019-03-30 09:06:17.0	70B3D5FFFE88B476				status	2019-03-30 09:06:16 CST	0.0	0.0	0	7	4	4
2019-03-30 09:06:14.0	70B3D5FFFE88B476	2019-03-30 09:06:14.0	30	30	heartbeat							
2019-03-30 09:05:44.0	70B3D5FFFE88B476	2019-03-30 09:05:43.0	30	30	heartbeat							
2019-03-30 09:05:14.0	70B3D5FFFE88B476	2019-03-30 09:05:13.0	30	30	heartbeat							
2019-03-30 09:04:44.0	70B3D5FFFE88B476	2019-03-30 09:04:43.0	30	30	heartbeat							
2019-03-30 09:04:14.0	70B3D5FFFE88B476	2019-03-30 09:04:13.0	30	30	heartbeat							
2019-03-30 09:03:44.0	70B3D5FFFE88B476	2019-03-30 09:03:43.0	30	30	heartbeat							
2019-03-30 09:03:13.0	70B3D5FFFE88B476	2019-03-30 09:03:13.0	30	30	heartbeat							
2019-03-30 09:02:43.0	70B3D5FFFE88B476	2019-03-30 09:02:43.0	30	30	heartbeat							
2019-03-30 09:02:13.0	70B3D5FFFE88B476	2019-03-30 09:02:13.0	30	30	heartbeat							
2019-03-30 09:01:43.0	70B3D5FFFE88B476	2019-03-30 09:01:43.0	30	30	heartbeat							

图 3-6 网关设备工作日志查看界面图

网关设备注册完成后，在网管云平台上可以将任意数量需要使用的传感器设备进行设备注册。传感器注册界面如图 3-7 所示。注册完成后打开传感器开关，当网关设备与传感器设备同时工作后，可以在网管云平台上查看到传感器设备的工作情况，接下来就可以访问网管云平台的数据接口得到需要的传感器工作数据。传感器日志流水查看界面如图 3-8 所示。

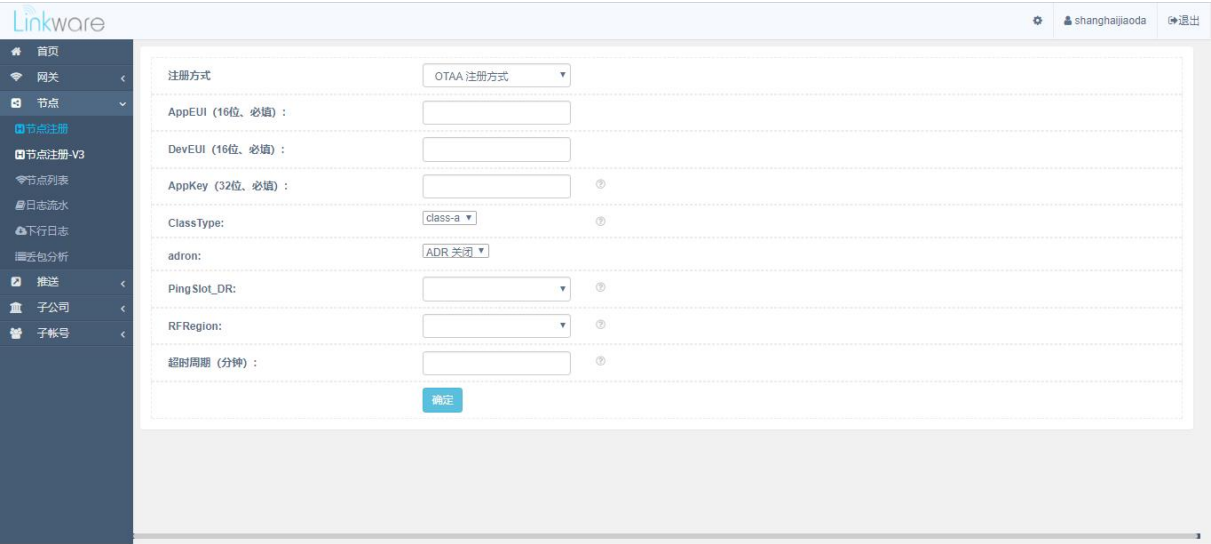


图 3-7 传感器设备注册界面图



图 3-8 传感器设备工作日志查看界面图

3.3.2 接口功能说明

网管云平台对第三方应用提供灵活的 Http 协议格式接口(也支持 Https 协议),返回数据采用 json 格式,可以快速和方便地进行集成。云平台接口调用示意图如图 3-9 所示。

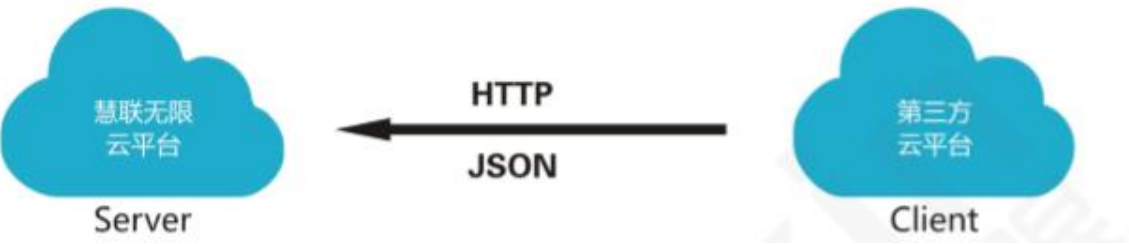


图 3-9 接口调用示意图

说明获取设备最新数据的接口,请求协议是 Http,请求类型是 Get 方式,请求地址是

http://api.easylinkin.com/api/v1/application/data，请求参数是 mac（必填，设备唯一编号）、token（必填，由网管云平台管理员设置并提供）、cid（必填，客户编号）。

返回响应为：

```
{
  appeui : string; //app EUI64
  mac : string; //设备唯一编号
  data : string; //需要取的数据
  last_update_time : string; //最后更新时间，格式：yyyyMMddHHmmss
  data_type; //数据类型
  status; //设备状态 0:在线 1:离线
}
```

实际调用接口时，JSON 格式如下所示：

```
{"appeui":"xxx","mac":"xxx","data":"xxx","last_update_time":"xxx","data_type":2,"status":"x"}
```

通过网络访问上面接口得到的有效数据为 data 字段，它是传感器采集到的温湿度数据，但它的长度为 10 个字节，也就是 data 字段是长度为 20 位的 string 类型 16 进制字符串，但解析数据需要将 16 进制字符串转化为 2 进制字符串，所以 data 字段中的数值说明都按照 2 进制格式说明。

温湿度数据由 8 个部分组成，数据字节结构如图 3-10 所示。

Bytes	1	1	1	2	1	1	1	2
Item	TYPE	VOL	RSSI	SNR	TEMP_INT	TEMP_FRA	HUMIDITY	CRC

图 3-10 温湿度数据字节结构图

接下来对每个部分的值进行说明，依次为

TYPE field

Bit	Name	Value	Description
7~4	TYPE	0x2	Message type, AS can use it to identify different uplink messages.
3~0	VER	0x0~0x1	Version of the message type, 0 means no SNR field. 1 means SNR filed included.

图 3-11 TYPE 部分数据说明图

TYPE 部分，如图 3-11 所示，data 的第 0~1 位拆分，将 data 拆分后还再需要拆分为两个部分，TYPE 为 2 进制第 7~4 位，表示该条信息的类型用于区分不同的上行数据，这里因为是周期性温湿度数据所以 TYPE 值应为 2；VER 为 2 进制第 3~0 位，表示该条信息的版本，0 代表没有 data 中没有 SNR 部分，1 代表 data 中包含 SNR 部分。

VOL field			
Bit	Name	Value	Description
7~0	VOL	0~100	Battery left (unit: percentage)

图 3-12 VOL 部分数据说明图

VOL 部分，如图 3-12 所示，data 的第 2~3 位拆分，将 data 拆分后不需要再拆分，VOL 为 2 进制第 7~0 位，表示传感器设备剩余的电池容量（单位为%）。VOL 部分数据说明如图 14 所示。

RSSI field			
Bit	Name	Value	Description
7~0	RSSI	0~160	The downlink Received Signal Strength Indication, detected and calculated by the sensor. (unit: -1dBm)

图 3-13 RSSI 部分数据说明图

RSSI 部分，如图 13 所示，data 的第 4~5 位拆分，将 data 拆分后不需要再拆分，RSSI 为 2 进制第 7~0 位，表示被传感器设备检测并计算得到的由网关下发后传感器接收到的信号强度（单位为 -1dBm），因为传感器向网关上发数据是要先收到网关下发的命令数据后才将当前数据上传给网关，所以这个可以看作是网关和传感器的通信质量，质量最好时为 0dBm，质量最差时为-160dBm。

SNR field			
Bit	Name	Value	Description
31~0	SNR	-3000~3000	The downlink Signal Noise Ratio, *0.01, i.e. -30.00~30.00

This filed is introduced since version 1, not included in version 0. If version field is 0, there will be no such field, the message length will be 8 bytes.

图 3-14 SNR 部分数据说明图

SNR 部分，如图 3-14 所示，data 的第 6~9 位拆分，将 data 拆分后不需要再拆分，表示下行信噪比。

TEMP_INT field			
Bit	Name	Value	Description
7~0	TEMP_INT	-127~127	Integer part of temperature. Bit7 indicates the sign, 1 means below zero. (unit: °C)

图 3-15 TEMP_INT 部分数据说明图

TEMP_INT 部分，如图 3-15 所示，data 的第 10~11 位拆分，将 data 拆分后不需要再拆分，TEMP_INT 为 2 进制第 7~0 位，但第 7 位为标志位 0 代表零上温度 1 代表零下温度，剩下 6~0 位为温度整数部分数值，表示传感器可以测得的温度整数部分最高为 127℃，最低温度为-127℃。

TEMP_FRA field

Bit	Name	Value	Description
7~0	TEMP_FRA	0~99	fractional part of temperature (unit: °C)

图 3-16 TEMP_FRA 部分数据说明图

TEMP_FRA 部分,如图 3-16 所示,data 的第 12~13 位拆分,将 data 拆分后不需要再拆分,TEMP_FRA 为 2 进制第 7~0 位,表示传感器可以测试的温度小数部分最高为 99,最低为 0。

HUMIDITY field

Bit	Name	Value	Description
7~0	HUMIDITY	0~100	Humidity (unit: %)

图 3-17 HUMIDITY 部分数据说明图

HUMIDITY 部分,如图 3-17 所示,data 的第 14~15 位拆分,将 data 拆分后不需要再拆分,HUMIDITY 为 2 进制第 7~0 位,表示传感器可以测得的湿度数值(单位为%),最高为 100%,最低为 0%。

CRC field

Bit	Name	Value	Description
15~0	CRC		Same CRC16 as calculated in register message. This can be used for server to check if any configuration mismatch.

图 3-18 CRC 部分数据说明图

CRC 部分,如图 3-18 所示,data 的第 16~19 位拆分,将 data 拆分后不需要再拆分,CRC 为 2 进制第 15~0 位,表示在注册信息中计算得到的同样的 CRC16,设备厂商用来检查服务器配置是否不匹配。

4 AR 显示与后台数据库

4.1 二维码原理分析

二维码是指在一维码的基础上扩展出的在二维方向上具有可读性信息的条码。黑色和白色的小正方形相间排列组成的矩形图案表示二进制数据，设备扫描二维码后可以解析得到里面所包含的信息。二维码存储容量大，且有一定的容错性，容错性表现在即使没有识别到全部的二维码、或者当二维码有污渍或损坏时，也不影响设备解析得到正确的二维码信息。二维码结构如图 4-1 所示。

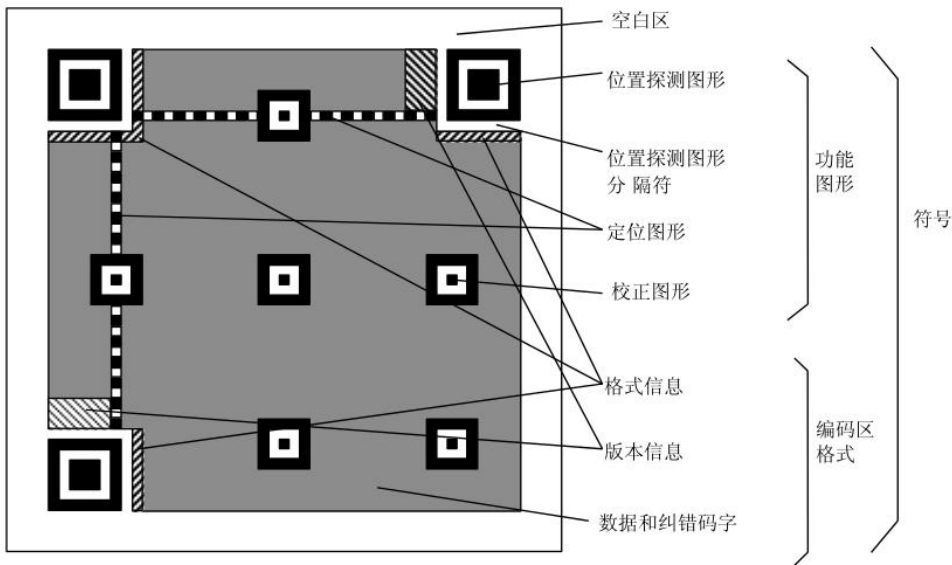


图 4-1 二维码结构示意图

4.2 ARCore 工作原理分析

ARCore 是 Google 的增强现实体验构建平台，使用手机摄像头进行现实世界的信息采集并将虚拟内容与现实世界相结合通过手机屏幕显示出来，它的工作原理主要由运动跟踪、环境理解、光估计这三个功能组成。

4.2.1 运动跟踪

跟踪和计算手机相对于现实世界的位置。跟踪用户的运动，并确定其在二维和三维空间中的位置是任何 AR 应用的基础。当手机移动时，ARCore 使用一个叫并行测距与映射（COM）的过程来跟踪计算手机相对于现实世界的位置。ARCore 计算手机位置变化的方式是通过对比纪录手机摄像头图像中每个像素间的视觉特征差异信息（特征点）并将这些视觉信息与手机的 IMU 惯性测量结果结合起来，得到一个手机随时间推移而相对于现实世界的姿态（位置与方向）。最后通过将 Unity3D 引擎渲染虚拟内容的虚拟摄像头的姿态与 ARCore 跟踪计算得到的手机摄像头的姿态对齐，就可以看到在 Unity3D 引擎中渲染的虚拟内容与现实世界结合显示在手机屏幕里，即使手持手机的位置和角度变化，显示出来的虚拟内容也会随着手机的变化而调整自身在三维空间中的位置和角度，就像是虚拟

内容真实存在于现实世界中一样。运动跟踪功能示例如图 4-2 所示。

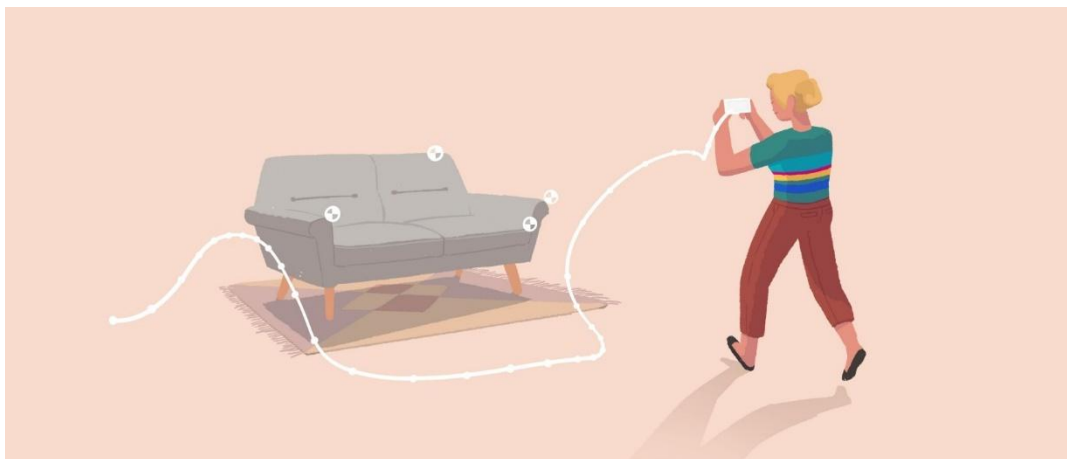


图 4-2 ARCore 运动跟踪示例图

4.2.2 环境理解

让手机可以检测各类表面的大小和位置，ARCore 会通过检测特征点和平面来不断改进它对现实世界环境的理解。在后来 ARCore SDK 的更新中，ARCore 开放支持检测环境中与图像数据库中参考图像特征点数据相符的物理图像和检测人类或者动物的脸部信息。环境理解功能示例如图 4-3 所示。

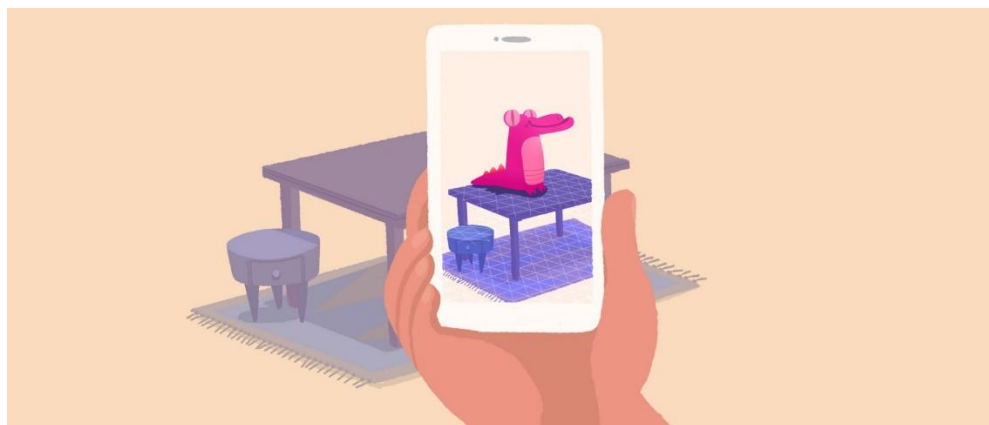


图 4-3 ARCore 环境理解示例图

4.2.3 射线检测

现实世界是三维的，而手机屏幕却是二维的，在二维的平面上确定想要在三维世界中显示模型的位置通常的做法就是做射线检测（Raycast），射线检测的基本思路是在三维世界中从一个点沿一个方向发射出一条无限长的射线，在射线的方向上，一旦与添加了碰撞器的模型发生碰撞，则产生一个碰撞检测到的对象，可以利用射线实现子弹击中目标的检测，也可以用射线来检测碰撞的位置，可以从屏幕中点击的点，利用摄像机的位置来构建一条射线，与场景中的平面进行碰撞检测，如果发生碰撞则返回碰撞的位置，这样就可以通过二维屏幕来实现在三维世界中的位置估算。射线检测功能示例如图 4-5 所示。

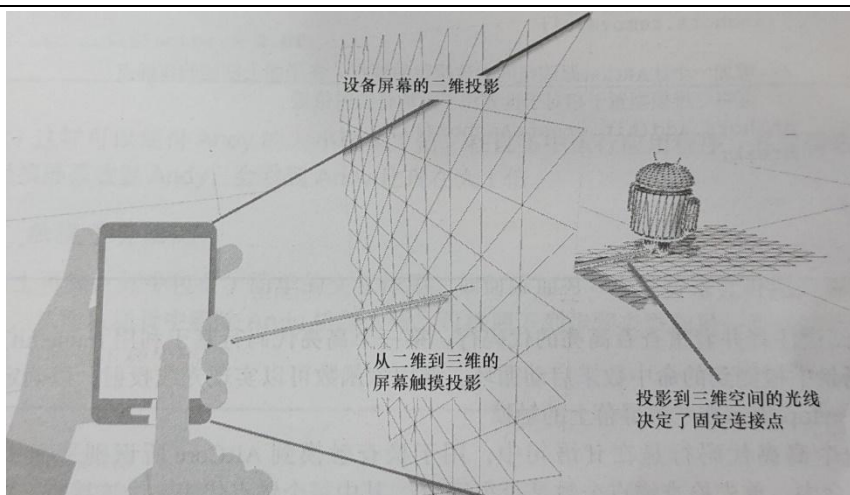


图 4-5 ARCore 射线检测示例图

4.3 MongoDB 数据库

MongoDB 是一个面向文档存储的分布式非关系型数据库，不需要预定义模式，架构灵活，维护成本低。相比于常见的 MySQL、SQL Server 和 SQLite 等关系型数据库，非关系型数据库中每个表的字段名称和属性不需要确定，一个数据表中可以每条数据的字段名称和属性都可以不同，每条数据中包含的字段数量也不固定，这种数据存储不需要固定的模式，非常利于超大规模数据的存储，可以很容易地进行横向扩展。

工业数据包括很多种类，所用于采集数据的传感器数量也很多，且每个传感器采集的数据类型都可能不一样，怎么将这么多种类不同的数据集中到一起来存储且后期要方便管理，成为了需要解决的问题。而 MongoDB 等非关系型数据库就是为这种用途而诞生的，我们可以把数量众多的传感器采集到的数据都集中到一个 MongoDB 数据库中的一个集合（表）里，即使这些传感器采集到的数据可能都不相同，例如在一个集合（表）里可以同时存储二氧化碳测量传感器、温湿度采集传感器和酸碱度测量传感器等用于采集不同数据的传感器数据，非常方便地进行管理和分析。

在本次系统实现中，关注点则只有温湿度传感器设备上传到云平台的温湿度数据，所以只需要一个集合（表）“sensor”用来存储键值对结构的数据，因为 MongoDB 是非关系型数据库的特性，集合不需提前创建，也不用指定集合中每个字段的属性，只需在进行数据插入时指定每个字段的属性和值就可以。

4.4 Unity3D 引擎

在 Unity3D 中主要完成对 ARCore SDK 在 Unity3D 引擎中的实现，搭建 AR 场景，生成 3D 模型，完成手机与现实世界的交互体验。

4.3.1 搭建 AR 环境

AR 最基础的部分是将手机摄像头与 Unity3D 相机组件关联，相当于人的眼睛是手机屏幕而手机

屏幕就是 Unity3D 相机组件，将手机位置的改变体现到 Unity3D 世界中。这部分的实现是将 Tracked Pose Driver 脚本和 ARCore Background Renderer 脚本挂载在相机组件上，ARCore Background Renderer 脚本的作用是调用手机的摄像头工作，将摄像头的图像信息转化为 Texture（纹理）类对象供 Unity3D 相机组件在 Unity 世界中渲染背景，使得手机屏幕看到的画面就是手机摄像头拍摄到的画面。Tracked Pose Driver 脚本是 Unity3D 引擎自带的将追踪设备得到的姿态（位置和方向）信息应用于游戏对象的 transform 组件的脚本，在这里追踪设备是指手机，游戏对象是指 Unity3D 相机组件，这样任何时刻相机组件都会与手机的姿态同步，意义在于当手机有位置移动后，交互操作在 Unity 世界中生成模型时，Unity 会理解后一个时刻手机屏幕位置所对应的 Unity 世界坐标与前一个时刻手机屏幕所对应的 Unity 世界坐标是不一样的，是相对于手机姿态变化的差值。Unity 场景中的 AR 相机对象如图 4-6 所示。

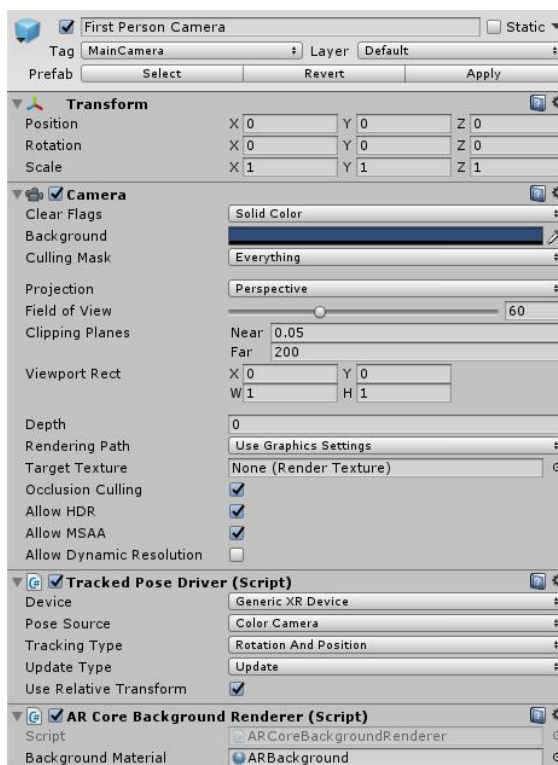


图 4-6 ARCore 相机组件

4.3.2 二维码扫描

使用 ZXing.net 插件，将手机每一帧显示的摄像头画面进行二维码解析，得到二维码代表的硬件型号信息。主要使用 ZXing.net 插件中 BroadReader 类的 Decode(Color32[] rawRGB ,int width,int height) 方法来对转换为 Color32 类型的摄像头图像信息进行解码，并用 ZXing.net 插件中的 Result 类来存储解析二维码得到的硬件型号信息。关键代码如下：

```
Color32[] data = texture.GetPixels32();
```



```
Result result = barcodeReader.Decode(data, texture.width,  
texture.height);
```

4.3.3 寻找平面

获取环境中的特征点进行网格化，生成平面模型。新建一个名为 PlaneGenerator 的游戏对象将 DetectedPlaneGenerator 脚本挂载在游戏对象上，并且指定脚本中定义的 public GameObject DetectedPlanePrefab 为预制件 DetectedPlaneVisualizer。DectedPlaneGenerator 游戏对象如图 4-7 所示。

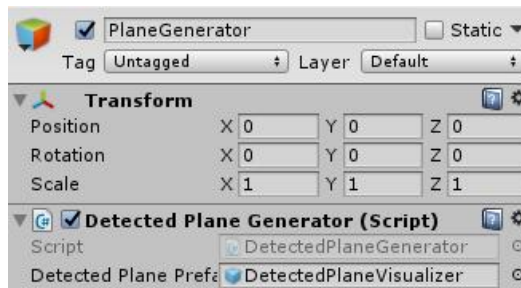


图 4-7 DetectedPlaneGenerator 游戏对象

DetectedPlaneVisualizer 预制件是在 Unity3D 中根据从现实世界中检测到的平面利用 PlaneGrid 材质来生成网格化平面的 3D 对象，平面是由预制件中挂载的 DetectedPlaneVisualizer 脚本渲染的。当手机摄像头采集到足够检测现实世界中平面的信息后在 Unity 世界中生成平面模型，而手机摄像头相当于 Unity 场景中的相机组件，所以在 Unity 世界中渲染的模型会显示在手机屏幕中现实世界对应的位置上。DectedPlaneVisualizer 预制件如图 4-8 所示。

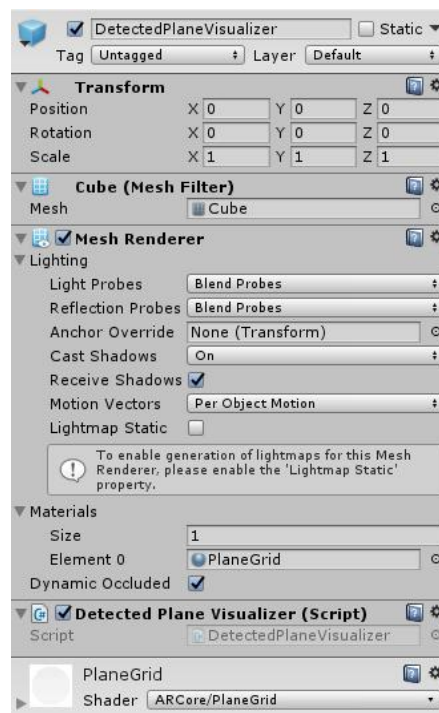


图 4-8 DetectedPlaneVisualizer 预制件

4.3.4 数据显示模型设计

当 Canvas 的渲染模式设置为 World Space 时，Canvas 被视为 3D 场景中与其他普通游戏对象性质相同的类似于一张平面（Plane）的游戏对象，即可以在场景中和其他 3D 对象一样显示出来。用于显示数据的模型界面如图 4-9 所示。



图 4-9 数据显示模型界面图

4.3.5 用户交互

在手机屏幕上点击 AR 场景中的平面模型后进行屏幕触摸点与平面模型的射线检测，判断触摸点与平面模型的碰撞检测结果，将碰撞点的位置和方向用来生成数据展示界面。关键代码如下：

```
// 获取屏幕触摸点
Touch touch;

if (Input.touchCount < 1 || (touch = Input.GetTouch(0)).phase !=
TouchPhase.Began) {
    return;
}

TrackableHit hit;

// 设置射线检测碰撞器类型为平面
TrackableHitFlags raycastFilter =
TrackableHitFlags.PlaneWithinPolygon;

// 进行射线检测
if (Frame.Raycast(touch.position.x, touch.position.y, raycastFilter,
out hit)) {
```

```
// 检查射线检测碰撞点在平面背面时
if ((hit.Trackable is DetectedPlane) &&
    Vector3.Dot(FirstPersonCamera.transform.position -
hit.Pose.position,
    hit.Pose.rotation * Vector3.up) < 0) {
    return;
}

// 正确地点击了平面的正面
else {
    SensorUIVisualizer visualizer = null;
    // 实例化模型
    visualizer = Instantiate(SensorUIVisualizerPrefab,
hit.Pose.position, hit.Pose.rotation);
    // 在碰撞检测点创建锚点
    Anchor anchor = hit.Trackable.CreateAnchor(hit.Pose);
    // 将模型绑定为锚点的子对象
    visualizer.transform.parent = anchor.transform;
}
}
```

4.3.6 与后台数据库的数据传输

因为 Unity 中的脚本文件是用 C# 编程语言编写的，而 MongoDB 官方有提供 C# 使用的 MongoDB .NET Driver，所以可以直接在 Unity 中连接 MongoDB 数据库。Unity 连接 MongoDB 需要用到 3 个文件，分别是 MongoDB.Bson.dll、MongoDB.Driver.dll 和 MongoDB.Driver.Core.dll。将这 3 个文件导入到 Unity 中后可以利用 MongoDB .NET Driver 的 API 接口来开发，进行工业数据存储。关键代码如下：

```
// 建立与后台 MongoDB 数据库的连接
MongoClient client = new MongoClient("mongodb://" + hostAddress + ":27017"
);

// 查找要使用的数据库
IMongoDatabase database = client.GetDatabase(databaseName);
```

```
// 得到数据库中要使用的集合（表）  
  
IMongoCollection<BsonDocument> collectoin =  
database.GetCollection<BsonDocument>(collectionName);  
  
// 进行数据存储  
  
collectoin.InsertOne(bsonDocument);
```

5 系统测试与扩展能力分析

5.1 实际效果测试

测试手机为华为 P20，系统版本为 EMUI 9.0.0（系统底层为 Android 9.0），屏幕宽高分辨率为 1080x2240，内存为 6.0 GB RAM，存储空间为 64.0 GB ROM，CPU 为麒麟 970，后置摄像头像素为 2000 万+1200 万像素双镜头，使用手机截图的方式得到工业 AR 程序运行测试效果。

在华为 P20 上打开工业 AR 程序后，按照 UI 的指示先将手机对准含有设备信息的二维码扫描读取 mac 型号为“004A77021106023E”的传感器采集到的温湿度数据。二维码扫描界面如图 5-1 所示。



图 5-1 二维码扫描界面截图

扫描成功后得到提示并出现需要用户寻找平面的信息，当通过移动手机成功在现实世界环境中寻找到平面后，平面会以网格化形式显示出来。网格化平面显示效果如图 5-2 所示。



图 5-2 平面可视化模型生成截图

用户需要用手指触摸屏幕中的平面显示区域进行 3D 工业数据显示模型的生成，查看 3D 界面模型的工业数据，并从不同角度观察 AR 显示与现实世界的结合效果。以正面角度观察 3D 工业数据显示模型的效果如图 5-3（1）所示、左侧 45 度角度如图 5-3（2）所示、右侧 45 度角度如图 5-3（3）所示、背面角度如图 5-3（4）所示。



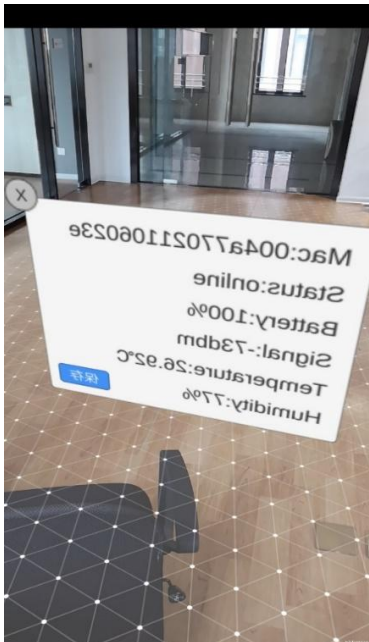
（1）正面查看数据的手机截图



（2）左侧 45° 查看数据的手机截图



（3）右侧 45° 查看数据的手机截图



（4）背面查看数据的手机截图

图 5-3 数据显示模型与现实世界结合的效果

在安装有本系统 MongoDB 数据库的 Windows 10 电脑上打开 cmd 命令行窗口，查看 MongoDB 数据库中关于上述测试所存储的工业工作数据。MongoDB 中测量数据的存储结果如图 5-4 所示。

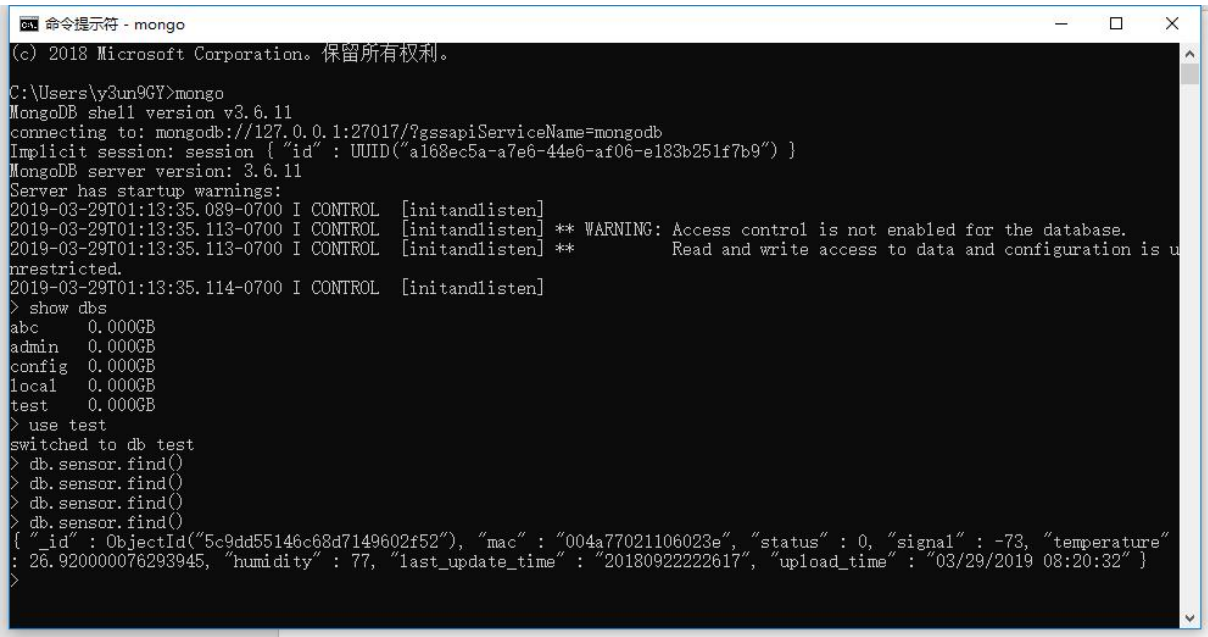


图 5-4 工业数据在后台数据库中的存储情况

5.2 扩展能力分析

扩展用于工业复杂结构设备的维护修复，例如汽车维修，发动机引擎、底盘、传动结构都是由很多小零部件组成，结构非常复杂。如果将汽车的各个硬件参数提前录入到数据库中，实现零部件的物体识别，将零部件的虚拟模型在对应位置显示，将维护修改的组装拆卸步骤排序显示，在维修人员进行维修的过程中使用增强现实形式同步指导，提高维修效率和维修过程中对零部件拆装卸的准确性。

扩展用于工业现场故障的远程诊断，加入视频通信技术，使得远程的资深工程师不需到达现场也可以通过现场工程师传输故障模块的画面进行故障模块识别，显示模块的辅助信息，帮助诊断故障信息，并远程下达指令，现场工程师根据系统传达的指令按照增强现实显示的形式对故障进行修复，提升故障诊断与修复的效率。

扩展用于工业技术培训，工厂的人员流动比较大，新员工需要在短时间内进入工作状态，这时技术培训就特别重要。但是由于传统教学理解抽象、教学耗费时间久和教学人员水平的参差不齐，导致培训完成之后，工厂新员工的动手能力仍然得不到很高的提升。利用本系统进行扩展，在教学中让每一个新员工在增强现实的指引下，对虚拟的工业模块进行操作，提升培训人员对培训内容的更真实的感受，让新员工接受系统化的标准步骤指导，更有效地提高他们的操作能力、调动他们的学习积极性，也不再受到培训场地的限制和教学人员水平的限制，从而使他们更快地熟悉工作环境，提升工作效率。

结论

本文设计了一个基于 AR 的工业数据交互系统，主要内容包括工业数据自动化采集装置、工业数据在手机端的 AR 显示和利用后台数据库进行数据交互。工业数据自动化采集装置包含网关、终端传感器设备与云端数据库，这一部分是和工业物联网有关的内容，可以使用但不仅限于本文中的装置。工业数据的 AR 显示则是本系统中很重要的一个部分，因为它是连接工业自动化采集装置和后台数据库的连接点，这部分的设计和实现使用了多种技术，包括二维码技术、增强现实技术，显示效果和交互操作是 AR 的两个重要内容。后台数据库更多是当作存储数据的作用，数据库的性能和扩展灵活性是本系统数据库选择的重要标准。通过系统中这三个部分的设计、研究和简单实现，尝试解决传统工业面临的技术瓶颈，提高工业生产制造效率，为工业领域与 AR 技术结合提供了技术方案探索。

本系统 AR 的显示效果还有很大的提升空间，目前只是侧重于最简单地使用文字形式显示数据，后续可以继续对 AR 显示效果进行美化，用更形象的方式显示工业数据，提升用户使用时的沉浸感。

本系统扩展性很强，具体可应用用途很多，就增强现实技术的发展来看，它的工业应用范围绝对比论文中讨论的内容多得多，论文中只实现了部分用途的简单实现，通过对工业 AR 系统扩展，可以实现很多工业场景的应用，但因为现在 ARCore 的识别能力有限，所以系统的 AR 具体显示环境的局限性比较大。

致谢

首先非常感谢上海交通大学的钱良博士提供的系统总体设计方向和测试开发设备，通过设计与实现本文的研究内容我接触到了很多当今流行和正处在起步发展阶段的技术，其次还要感谢我的指导老师杨蒙召老师对我论文写作的指导，在完成毕业设计论文过程中两位老师给了我很多帮助与指导。

经过这段时间刻苦的学习和研究，我完成了我大学生涯的最后一次“作业”。因为用的开发工具是以前没有接触过的游戏开发引擎，系统具体实现方法还是和做软件开发有很多的不同，在这期间遇到了很多概念上的误区与技术上的难题，但在两位老师和众多同学的帮助下都得到了解决。

一个阶段的结束同时也是另一个阶段的开始，虽然我的大学生活即将结束，但我在新的道路上还有很长的路要走。四年的计算机专业学习激发了我对于编程的兴趣和热情，增强了我的自学能力和解决问题的能力；四年的校园生活也培养了我与同学老师之间的情谊。

在此再次感谢在这四年里一直支持和帮助我的同学与老师，谢谢他们！

参考文献

- [1] 蒋武洲,周海东,刘福等. 物联网 LoRa 技术的应用与发展[J]. 电视技术, 2018, 42(8): 100-103
- [2] 李伟杰,戴亚文. 基于 LoRa 的工业温度无线采集系统设计. 自动化与仪表, 2019, 34(1): 37-40、49
- [3] 余晓晖. 工业互联网与产品智能化变革. 衡阳通讯, 2018, (10):18-20
- [4] 杨荣杰. 基于信息链接技术的配电自动化设备现场技术支持系统的应用. 电气技术, 2018, 37(14): 48-50
- [5] 魏源通. 基于 LoRa 的工业检测自组网系统设计. 计算机测量与控制, 2019, 27(2): 225-228
- [6] 曹伟智. 工业设计领域下 VR 和 AR 技术的新融合. 美术大观, 2018, (11): 90-91
- [7] 倪乐波, 戚鹏, 遇丽娜, 王婧. Unity3d 产品虚拟展示技术的研究与应用. 数字技术与应用 [A], 2010, 9(35): 54-55
- [8] 周玲, 陈强. 城市地下管网增强现实系统研究. 计算机工程与应用, 2019
- [9] 陆泽萍, 谢伙生, 陈鑫凯, 林晶. 基于 Unity3D 的实验室移动 AR 自动导览系统设计与实现. 福建电脑, 2018 (12): 113-116
- [10] 凌骁珺, 王煜. 二维码技术的分析与研究. 电脑知识与技术, 2018 (21): 47-49