



山东科技大学  
SHANDONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

## 本科毕业设计（论文）

题目：远程数字化身

学 院： 电气信息学院

专业班级： 计算机科学与技术 18 级 4 班

姓 名： 侯普领

学 号： 201803204408

指导教师： 任丙忠

完成日期： 2022 年 5 月 1 日

**BACHELOR'S DEGREE THESIS OF  
SHANDONG UNIVERSITY  
OF SCIENCE AND TECHNOLOGY**

**Remote Digital Avatar**

College: Shandong University of Science and Technology

Subject: Computer Science and Technology

Name: Puling Hou

Directed by: Bingzhong Ren

**QINGDAO CHINA**

学术声明:

## 郑重声明

本人呈交的毕业论文，是在导师的指导下，独立进行研究工作所取得的成果，所有数据、图片资料真实可靠。尽我所知，除文中已经注明引用的内容外，本设计（论文）的研究成果不包含他人享有著作权的内容。对本设计（论文）所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确的方式标明。本设计（论文）的知识产权归属于培养单位。

本人签名: \_\_\_\_\_ 日期: \_\_\_\_\_

## 摘要

人类的身体有诸多局限性，而且我并不认为在人类的科技和道德达到一个较高水准的情况下对身体直接进行接入网络的生化改造是个好主意，在这之前还是应该掌握身体的全部权限；为了在保持身体独立性的前提下最大程度的实现身体能力的扩展，另外可选的技术还有很多，例如外骨骼、可穿戴设备、非联网植入式义体等，但还有另一种方式，可以让我们“脱离”自己的躯体，但是不会对我们的身体进行任何改造，而且没有被控制的风险，同时更重要的是能给我们提供另一种形式的生命体验，这就是——数字化身。

网络如同机器的意识，失去了机器将失去与现实世界交互的能力。反之，拥有了机器，网络将获得与现实交互的能力。而人类也可以借由网络获得超越空间限制的，与现实交互的能力。本课题将研究一种以现有的技术水平可以进行大规模商用铺设的远程数字化身系统，将对其“用户——服务器——基站——机器人”模式进行分析介绍。

疫情的出现导致的区域封锁进一步体现了人类肉体的局限性，所以人类目前已经可以并且已经到了进行这种应用初级形态的普及的最好时期；

**关键词：**远程数字化身；远程操作机器人；元宇宙应用；

## ABSTRACT

Our bodies have many limitations, and I don't think it's a good idea to biochemically modify the body directly to the Internet when human technology and morality have reached a high level. Before that, we should master all the body Permissions; in order to maximize the expansion of physical capabilities on the premise of maintaining physical independence, there are many other optional technologies, such as exoskeletons, wearable devices, and non-networked implantable prostheses. But there is another way that allows us to "break out" of our bodies, but without any modification to our bodies, without the risk of being controlled, and more importantly by giving us another form of Life experience, that's what - digital avatars.

The network is like the consciousness of the machine. Without the network, the machine will lose the ability to interact with the real world. Conversely, with machines, the network will gain the ability to interact with reality, and humans can also gain the ability to interact with reality beyond the limits of space through the network. This topic will study a remote digital avatar system that can be deployed on a large scale with the existing technical level, and will analyze and introduce its "user-server-base station-robot" mode, and based on this, propose a A prototype of a set of general protocols.

The regional blockade caused by the emergence of the epidemic further reflects the limitations of the human body, so human beings can and have reached the best time to popularize this primary form of application;

**KEY WORDS:** remote digital avatar; remotely operated robot;metaverse application

# 目 录

第1章 绪论.....	3
1.1 问题的提出及研究意义 .....	3
1.2 相关技术概论 .....	4
1.2.1 图像压缩及视频编码技术 .....	4
1.2.2 超分辨率重建技术 .....	5
1.2.3 SLAM 技术 .....	6
1.2.4 元宇宙 .....	7
1.3 国内外研究现状 .....	8
1.4 论文叙述结构 .....	8
第二章 RDA 系统概述及分析 .....	10
2.1 RDA 系统工作模式概况 .....	10
2.2 RDA 服务类型 .....	12
2.2.1 商用型 RDA .....	13
2.2.2 专业型 RDA .....	16
第三章 RDA 系统设计分析 .....	18
3.1 RDA 云服务器设计分析 .....	19
3.2 RDA 基站设计分析 .....	20
基站内机器人的连接管理 .....	21
基站结构设计要求 .....	22
3.3 RDA 机器人设计分析 .....	22
第四章 RDA 系统搭建 .....	24
4.1 WS 通用服务篇 .....	24
4.1.3 WS Service 部署模式 .....	25
4.1.4 WS 服务搭建 .....	25
4.2 云服务器篇 .....	27
4.2.2 API 接口设计 .....	27
4.1.3 表结构设计 .....	28
4.1.4 业务服务器搭建 .....	32
4.2 基站篇 .....	35
4.2.1 基站服务器 .....	35
4.2.1 基站结构概念设计 .....	36
4.3 机器人篇 .....	38
4.3.1 机器人客户端 .....	38
4.3.2 机器人概念形态设计及要点 .....	44
4.4 用户客户端篇 .....	49
4.4.1 用户操作 UI .....	50
4.4.2 基于 UDP 的 P2P 视频传输接收模块 .....	50
4.4.4 图像实时超分辨率重建模块 .....	51

4.4.4 基于 TCP 的 P2P 指令传输发送端.....	55
总结 .....	57
参考文献 .....	58
致谢 .....	60

# 第1章 绪论

## 1.1 问题的提出及研究意义

华住会旗下开始推出一种叫“智慧酒店”的酒店类型，我第一次去汉庭智慧酒店的时候就跟酒店里的一台机器人一起坐电梯，是它叫的电梯。它是一个圆筒形的机器人，还闪烁着蓝光，我们在电梯里，我就在它的后方看着它。它刚才通过某种无线技术联络电梯来接它，等到达目的地后电梯再通知它该下车了，它便继续慢悠悠的离开电梯往前移动。然后我突然意识到刚才我是把它当作一个独立的个体看待的，所以它应该有独立的思维、独立的行为，所以我刚才描述为“它联络电梯”，不过事实上，它叫电梯的方式跟人类完全不同，主要的一点区别在于——它不是通过自己的感官独立的判断位置的；它与电梯其实是一个整体，应该说，整个酒店才是一个机器人。

酒店里的一切通过网络连接，运行在酒店局域网中的系统就是酒店的灵魂，这没什么难理解的，很多酒店都有管理系统，智慧酒店的特殊在于，它拥有“从网络空间伸向现实世界的触手”。

这其实就是物联网系统，物联网和传统互联网的区别就在于那条从网络空间伸向现实世界的触手——机器。

目前人类已经可以让声音和影像通过网络即刻到达全球的任何位置，比较普遍的如视频聊天、直播；普及度还不是那么广的如 VR 在线会议，目前做的比较好的比如 Facebook 旗下的 Horizon Workroom，借助今年 Facebook 刚发布的 VR 眼镜 Oculus quest2，可以做到多人以虚拟形象在一个虚拟会议室空间里在线开会，并且可以将连接的电脑桌面投射到这个空间里的虚拟电脑上去，就像在这个空间里使用电脑一样。从 BB 机到智能手机，从音频通话到实时视频通话再到多人虚拟实境，从马车到高铁，我们所做的努力都是为了一个问题——打破距离和物理规则的限制。目前还有最重要的一块版图没有完成——在现实世界里跨越距离进行交互。

为了拼凑上这最后一块版图，我设计了一个系统，即——通过大规模的铺设“网络伸向现实的触手”——机器人，让早已可以通过网络穿梭在世界各地的人

们真正获得与现实世界交互的能力，就像，打破一个次元壁，拥有一个在线下的数字化身（RDA， Remote Digital Avatar）。

你可以随时随地自由接入一台位于遥远地方的机器人，雇佣以获得其使用权，在大街上穿梭，在当地的超市里购物，探望亲人朋友；或者你是个科研人员，你“雇佣了”一台位于南太平洋斐济群岛基站的一台深潜探测机器人，操控其对海底微生物进行采样；或者你是未来某所中学的地理课老师，接下来你将进行有关月球环形山地貌的课程，于是你接入了一台位于月面基站的机器人，并与全班同学共享 VR 眼镜的视频数据，一起操控这台机器人在月球表面探索；再或者你是一个极客兼旅行爱好者，你自己拥有几台机器人，你也可以将其接入系统，当你在世界各地旅行的时候，其他人可以接入你的机器人，在他们感兴趣的地方逛逛，你也可以赚取一笔租赁费用。

为实现这个光景，我们需要的不仅仅是要实现一个远程遥控系统那么简单，我们需要一种新的应用形态，一种可以让每个人都可以自由通过网络与远方的现实交互，并且可以深度参与进这个社区的系统形态。

一个开放的系统，允许用户自由构建、全身心参与的系统，与近两年大热的一个概念所描绘的未来图景非常相似——Metaverse—元宇宙。在下面的相关技术概论章节中会详细介绍元宇宙和其他一些相关技术概念。元宇宙将人类更加紧密的连接了起来，模糊了线上线下的概念，而真正打破线上线下的次元壁，这个系统便尤为重要和关键。

## 1.2 相关技术概论

RDA 系统是一套复杂的系统，需结合大量前沿技术，这一章节我们将对涉及到的相关技术做一个简要的论述。

### 1.2.1 图像压缩及视频编码技术

由于当代信息技术的快速发展，数据可以有多种表示方式。数据文件包括了文本、图像、音频和视频，这些都需要占用大量的硬盘空间。视频文件可以理解

为由一系列图像按帧组成的单独个体。在接受到前端化身机器人捕获到的视频后，传输过程会带来更高的带宽要求，所以需要将高质量的图像低损失和无延迟地传输到用户设备，视频压缩能够高效的减少存储空间并加速传输过程。

视频压缩技术用于减少和消除视频数据中的冗余。压缩后的视频文件更小，因此视频可以通过网络快速传输。压缩效率取决于给定分辨率和帧速率的比特率。对于较低的比特率，压缩将更有效。在图像质量降低的情况下，视频压缩可能有损。对于有损压缩，可以开发有效的压缩技术来提供高质量的视频。即使压缩后的视频与原始视频不同，肉眼也无法清楚地看到这些差异。

目前主流的视频压缩传输方式有单帧图片压缩传输与视频编码，视频编码需要考虑前后帧之间的关系，例如 H264 编码，但是在实时视频传输时，可以不考虑 P 帧，只考虑当前帧和前一帧；常用的视频编码技术有 H264、H265、VP8、VP9 等；单帧图片压缩传输就是将接收到的每一帧独立压缩传输，不需要考虑前后帧，常用的有 JPEG，这可以说是所有图像压缩技术的基石；mpeg，基于 JPEG 的动态图像压缩技术，等；对视频进行编码传输与图片压缩传输相比，对硬件的要求要高一些，码率越高对硬件的要求越高；

现存的所有的压缩标准都是从共同的起源演变而来的，所以它们之间共享着许多基本原则，比如对视频序列图像进行编码时，通常将图像分割为大小  $16 \times 16$  像素的规则的二分模块。对于每一个模块，都可以应用不同的方法来对图像信息进行编码，本文采用了 JPEG 的方式对块进行编码，这种方式也被称为内部模式<sup>[2]</sup>。

### 1.2.2 超分辨率重建技术

图像分辨率用于评估图像中的细节丰富程度，所谓超分辨率重建就是将给定的细节丰富程度较低的图片还原成细节丰富程度较高的图片；为了实现这个过程，我们首先需要知道自然界中的高分辨率图像经过我们观测传导到我们的眼睛里怎么变成低分辨率图像的，这就是降质退化模型；

目前主流的超分辨率重建技术有三种：

1. 基于插值的超分辨率重建
2. 基于重构的超分辨率重建
3. 基于学习的超分辨率重建

由于实景交互型 RDA 在使用过程中需要实时高清视频传输, 而传输低分辨率图像, 可以大幅降低数据包大小, 同时降低机器人设备的性能要求, 到客户端在对收到的低分辨率图像进行重建; 本次系统将使用基于学习的超分辨率重建技术。

### 1.2.3 SLAM 技术

当机器人在一个新的环境中, 需要快速对周边地图进行构建, 从而利用地图进行自我定位和路线导航, SLAM(Simultaneous localization and mapping)技术直译为同步定位与建图, 可以良好的完成机器人对于环境重建这一任务。

SLAM 是一个计算问题, 能够构建或更新未知环境的地图, 同时跟踪代理在其中的位置。SLAM 技术广泛的应用于自动驾驶、无人机和移动机器人等。在本文使用场景中一个 SLAM 过程可以理解为机器人在一个未知环境中从一个未知坐标开始进行移动, 通过自己位置的估计在移动过程中进行定位, 从而逐步完善当前未知环境的地图构建。移动机器人在运行时使用自身携带的各种传感器在未知环境中识别用于 SLAM 系统准确定位的特征标志, 识别到标志后根据机器人与特征标志间的相对位置和里程计读数估计机器人和特征标志的全局坐标。

本课题机器人拟采用 ORB-SLAM3 完成了地图的创建并同时对机器人位置进行定位, 其框架如图 1 所示。ORB-SLAM3 包含了跟踪、局部建图、闭环这三个线程<sup>[1]</sup>。

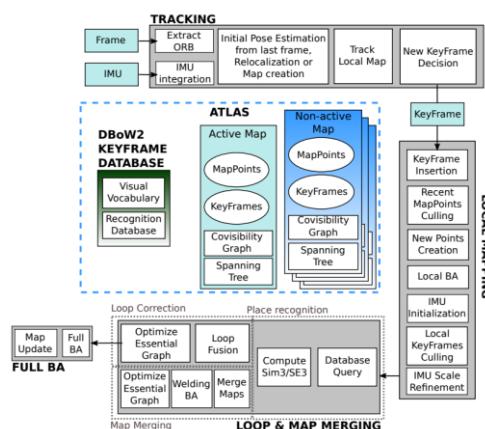


图 1<sup>[1]</sup>

跟踪线程的输入为前端获取到的视频帧集，可以是单目、双目或 RGBD 的格式，还有一个数据流为 IMU，即惯性测量单元。在获取到输入数据之后，第一步需要提取 ORB 特征点和对 IMU 进行积分来计算不同时刻下的速度。跟踪线程分成了三个阶段，第一个阶段通过最后一帧对姿态进行粗略的初始化和估计，并对地图重新载入或者创建，第二阶段为跟踪局部地图点，通过局部区域的辅助，此时优化出的位姿相对第一阶段来说更加精细，最后一个阶段决定是否当前帧为关键帧。

在获取关键帧之后，进入第二线程局部建图部分。通过一系列操作，关键帧插入、剔除地图点、新增地图点、IMU 初始化、关键帧剔除以及 IMU 尺度精细化来完成地图的局部创建。

在局部建图线程中获取到的关键帧作为第三部分闭环以及地图合并线程的输入，通过计算 Sim3 完成对候选关键帧的检测，然后对本质图进行优化并对环进行融合，最后得到全局的地图。

#### 1.2.4 元宇宙<sup>[11]</sup>

这是最近两年特别火的一个概念，有人说这是未来，有人说这就是骗局；不过元宇宙不是一项具体的技术，它是一种技术形态，象征着人与人更加紧密的连接，与打破肉体限制和自由创造的欲望；

引用陈序老师的一句话来解释元宇宙——“元宇宙是一种让人以独立的数字身份参与和生活的可能的数字世界”。所以元宇宙与现实世界的地位是平等的，并且有两种可能的发展方向，一个是与现实世界密切交融的“狭义元宇宙”；一种是完全脱离现实世界存在的“广义元宇宙”，可能就像某篇科幻小说中描述的，一根老旧的线缆连接到一件破旧的屋子里，里面有一台蒙着一层尘土的终端，屏幕上显示着——“不要拔掉电源！这里面生活着一百亿人”，而转头一看，一直老鼠正在啃噬那根孱弱的电缆线。

所以我们还是聚焦于狭义的元宇宙；狭义的元宇宙与现实世界密不可分，它是现实世界的一个投影，人们在其中工作生活娱乐，随时在线上线下切换；例如将来互联网公司可能选择开到元宇宙里，而你在元宇宙中的数字身份被公司录用了，以后每天你戴上 VR 眼镜在元宇宙中去公司上班，尽管公司的实体在很远的

地方，或者根本没有线下实体；这种场景说起来似乎也并不算太新颖，但要实现真正的“线上=线下”，在法律上和观念上都做到等同的话，需要非常多的规则需要制定；比如承载这个世界的服务器是不是就不应该由一个中心化的公司所把控，毕竟你的数字身份是一个独立的身份，在虚拟世界中跟你之于你在现实世界中的身份是等价的，而你在现实世界中的存在与否并不会受某个集体的把控对吧。

在这种线上线下界限模糊的技术形态下，仍有最后一环没有补齐，那就是尽管你在元宇宙中漫步在巴黎的数字孪生城市中，你也不可能真的立即和线下真实的巴黎中的某个行人产生物理上的交互，线上线下还缺少最后一个媒介——物理层面即时交互的媒介。

### 1.3 国内外研究现状

有一个与远程数字化身相似的概念，叫做 Tele Existence<sup>[3]</sup>，由 Susumu Tachi 在 1980 年首次提出，其在论文中描述了将虚拟现实技术应用到工业远程机器人控制的应用场景。并通过国家大型项目开发的 TELEExistence Surrogate Anthropomorphic Robot (TELESAR) 和 TELESAR V 等异构机器人系统的构建证明了其可行性。分别是“危险环境的机器人”和“CREST Haptic Telexistence Project”，以及“Humanoid Robotics Project”下开发的 HRP 超级座舱双足机器人系统。TELESAR II 和 IV 等相互遥存系统已经开发出来，能够通过将异我机器人与反射投影技术 (RPT) 相结合，在本地空间中产生身临其境的感觉，验证了 Teleexistence 技术的可行性。

而本课题则致力于设计一种适用于大规模商用铺设和普通用户使用的系统架构，不需要复杂和价格高昂的终端设备和学习过程，借助现有的技术完全可以实现。

### 1.4 论文叙述结构

本文将按照以下结构进行叙述：

第一章：绪论部分，讨论问题研究的背景及研究意义，并对涉及到的相关技术做一个概述。第二章和第三章，讲解RDA系统的工作模式概况，并通过从应用场景的分析，讲解RDA系统的设计要点。第四章，根据上面讨论出的系统结构和

设计要点实现一个简单的示例系统，以及基站和机器人的概念设计图。

最后，对所作内容进行总结，列出参考文献，并致谢。

## 第二章 RDA 系统概述及分析

### 2.1 RDA 系统工作模式概况

RDA系统本质上是一种开放式的机器人租赁服务系统，由四部分组成，分别是——云服务器、机器人基站（后面简称基站）、机器人、用户客户端。它的工作模式如下图：

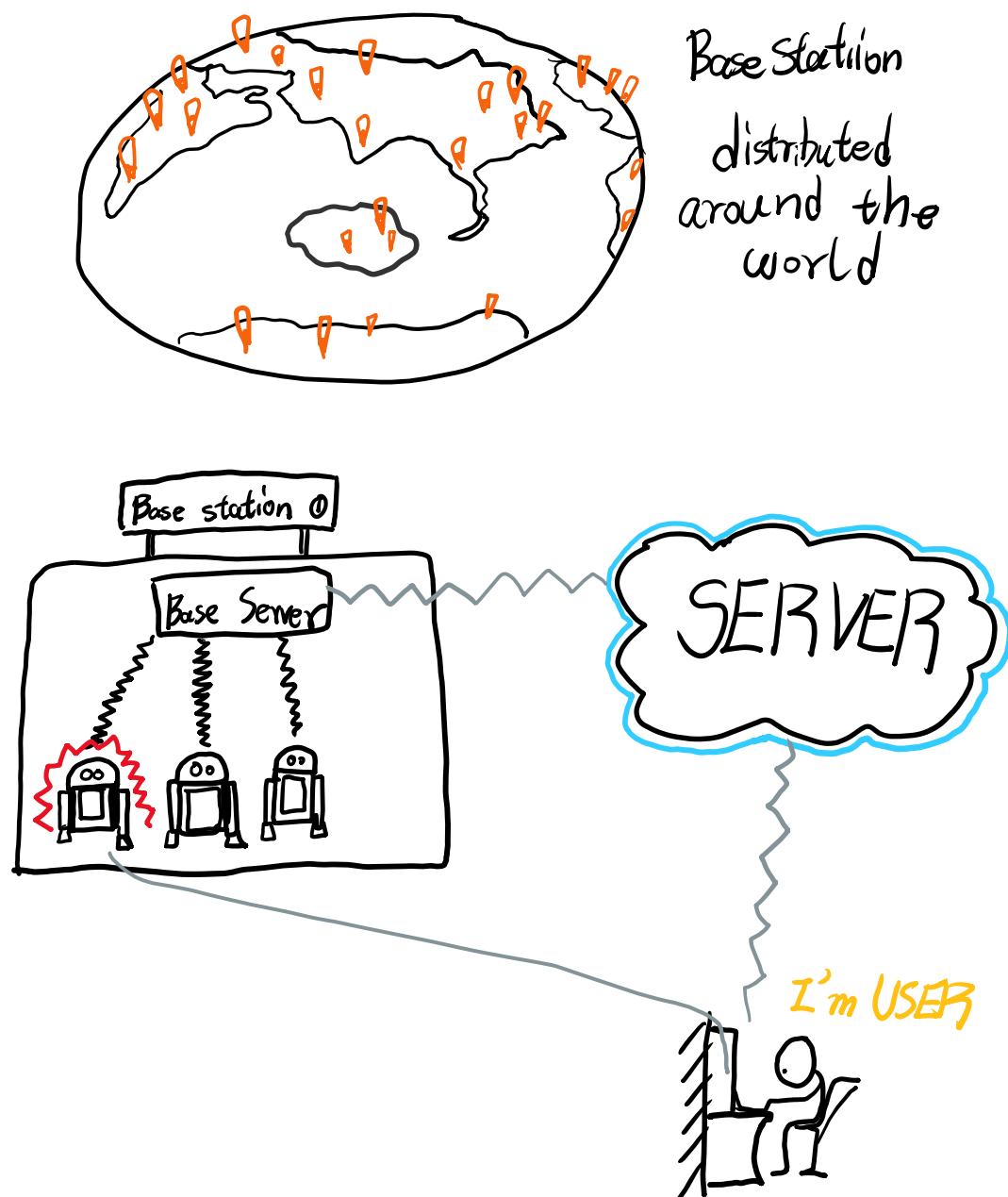


图2

如上图所示，RDA基站在全球范围分布，基站用于存储维护机器人，并在云服务器上保持登陆状态；云服务器则记载着基站信息及其所有的机器人信息，我们来以用户张三为例说明系统的运作过程：

张三目前人在哈尔滨，今天他想去重庆逛逛顺便看一下自己的老朋友，他选择通过在线RDA服务来完成这一计划。

首先，他通过在自己的终端上运行RDA客户端与云服务器连接，可以查看指定区域内的基站分布；张三确定地域位置为重庆，电脑将虚拟地球缩放到重庆的范围；重庆市内区域显示有五个基站——洪崖洞基站、解放碑基站、重庆大剧院基站、环球金融中心基站和双子塔基站，其中大剧院基站和金融中心基站都是陆空基站，张三选择接入重庆大剧院基站，他想顺便看场歌剧。

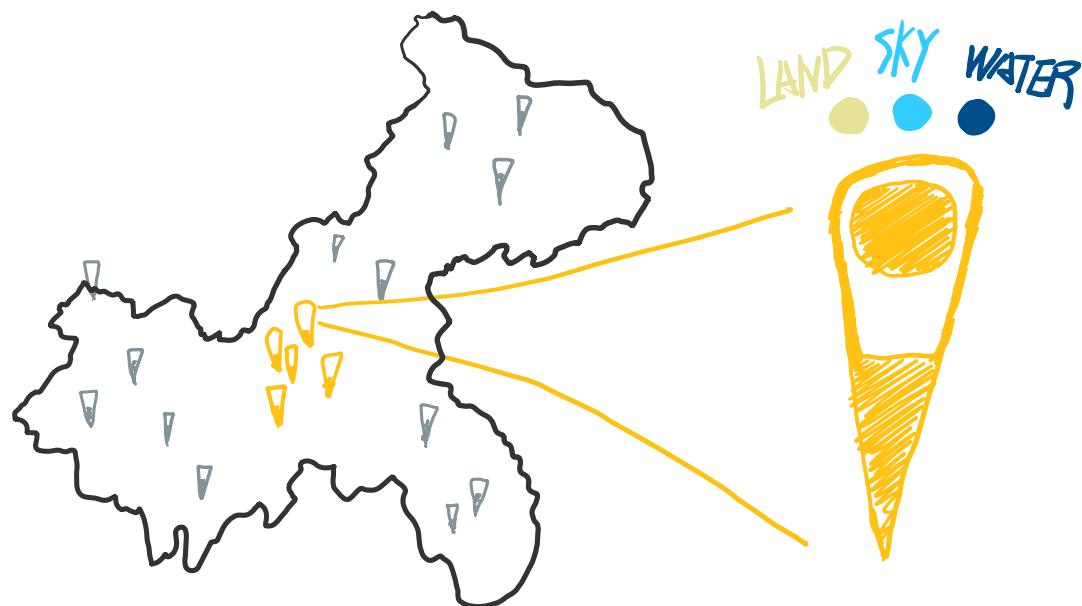


图3

页面转入重庆大剧院基站详情页，该页面将会显示基站内的所有可用机器人的信息，包括型号、状态等；目前显示重庆大剧院基站总共有200个机器人，其中有150台陆行机器人，50台空行机器人，目前可用数量分别为25和5台，看来需求量还是很大的；张三选择陆行机器人接入服务，选择服务时间为一天，进行缴费；这时云服务器开始给大剧院基站发送“获取机器人”请求，请求信息包括——用户ID、机器人类型、租用时间；基站接收到该请求后将根据要求的机器人类型选择一台处于待命状态的机器人，型号是R2L2-L06s，并回传给服务器该机器人的信息，主要包括机器人ID、机器人IPV6地址并分配连接用的token，同时给

了机器人一份；服务器将此信息包转发给用户客户端，用户客户端将通过IPV6地址和分配的token直接连接该机器人，开启音视频和指令传输通道，此时张三便可以驾驶这台机器人进行自由活动了。

张三在网上买了大剧院的票，将票的二维码通过机器人的小屏幕展示给剧院检票人员；R2L2-06s型机器人配有超广角摄像头，经过客户端的实时图像超分辨率重建，张三通过自己的VR眼镜可以获得很好的身临其境的体验；在这之后，机器人张三去逛了商场，买了些东西，通过机器人小屏幕展示付款码，并将购买的物品放进机器人的储物空间中。

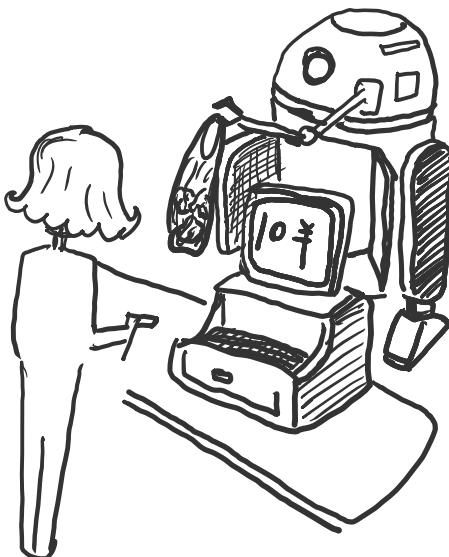


图4

在一天的最后“张三”去看望了他的朋友，并送给了他朋友一些上午买到的东西。

服务时间快要结束了，张三驾驶着机器人返回了基站，基站检测到机器人储物空间有物品，于是张三的客户端弹出一个询问框——“服务结束，请确认您新增的物品，确认邮寄？”点击确认后，系统将自动把张三在服务过程中增加的物品邮寄给张三，就像真的在那边购物过了一样。

## 2.2 RDA 服务类型

在上文的说明以及示例场景中，我们可以知道，远程数字化身就是借助分布在不同地方的具有各种交互功能的机器人来做为“你”在当地的化身，通过各地

的基站来统一管理和连接各地区的机器人，再由一个位于云端的中央系统将世界各地的基站和用户连接起来；所以实现 RDA 的关键在于广泛的基站分布和可以满足各种用户需求的基站和机器人种类。

因为需要远程数字化身服务的用户类型是多种多样的，像普通用户只是需要通用性的服务，即能满足在当地的自由移动和交互；而对于选择这项服务的科研人员，他们有着比较专业的用途，例如地质勘测、水文勘测、标本采样、环境分析等，需要具有特殊功能的机器人，同时其工作地点也就是基站的地理位置也会要求比较特殊。

所以为满足不同用户的需求，RDA 系统在大类上分为两大类——商用型、专用型。

### 2.2.1 商用型 RDA

其中，商业型 RDA 的使用场景聚焦于“使人通过网络在另一地与当地的人事物自由交互”，比如应用到旅游业；线上旅游这个词一点都不陌生，它从刚推出到现在已经有几年了，所谓线上旅游目前就是通过提前对场景——比如某一个



图 5

博物馆，对其各个场景进行 360° 拍摄，生成球形照片，然后再在某些著名藏品上附上链接，点开后是详细介绍，或许还有这个展品的三维图；正在进行这

种工作的最著名的就是 Google 了，Google 在早年间首先建立了一个地球模型，使得人们可以鸟瞰很多区域，然后慢慢的往里面补充街景，人们将街景小人放到一些有街景的位置就可以进入这片区域的街景；Google 团队在全球范围搜集街景，有很多奇怪的拍摄工具，比如街景车、街景三轮、街景背包、街景火车、街景船，为了应对某些特殊的环境，还聘请了一些专业人士，比如街景骆驼、街景袋鼠，如图 5。

生成的街景基本是个静态场景，通过转动鼠标来左右上下旋转，点击设置好的箭头切换场景；街景的清晰度还算可以，但是体验方式只是单向的，而且非常不灵活，而且可以看到全世界街景制作的成本是非常高的，并且没有实时性，看到的图片很多都是几年前拍摄的。

现在看 Google earth 里很多区域还拥有了三维模型，特别是美国，比如如图 6 中的纽约：

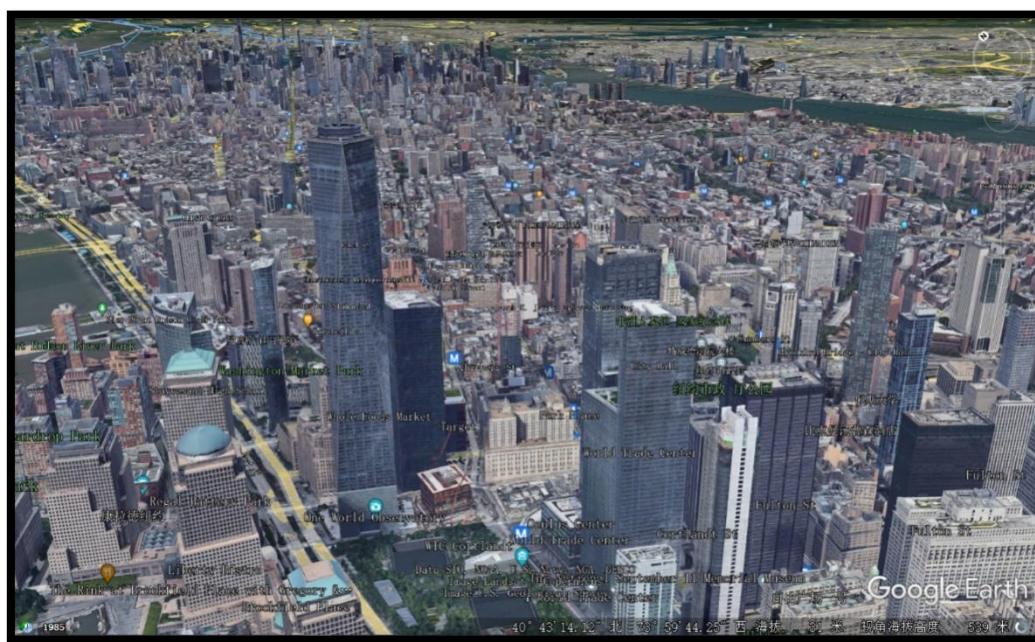


图 6

这种城市模型借助“倾斜摄影”建模，可以使用包括无人机、高空飞机，甚至卫星提供的数据进行全自动建模；这种方式比街景要灵活很多，而且区域覆盖大，不过明显清晰度不是很够，而且完全达不到“身临其境”的感觉，其体验方式也是单向的，也没有实时性。

自 2020 年疫情爆发以来，全球旅游业受到了沉重打击，人们都被封锁在家中不能自由出行；但大危机往往是大变革来临前的黑暗，根据麦肯锡对全球旅游

消费复苏的预测，中国旅游行业最快可能于 2022 年恢复至 2019 年的水平，并建议企业做好以下三件事：

1. 坚定的数字化转型
2. 敏捷的疫情应对
3. 果断地新机把握

其中针对旅游消费的数字化转型，在数字化营销方面，各大短视频和直播平台网红带货的模式逐渐兴起，例如那个被打造成理塘旅游 IP 的一夜爆火的丁真，以及去年春节时马蜂窝举办的直播旅游，这种模式更多的只注重“经济”，即你在直播间里买了什么东西，借旅游为名开凿消费主义陷阱，毕竟直播旅游的话，你将被迫和数百万个直播间看直播的人以及主播一同“旅游”，而且没有什么交互性，只是听主播讲解卖货罢了；大前研一所说的低欲望社会不是一种好的社会，但是欲望过大的浮躁的社会也不能说是一种好的社会。

所以，RDA 需要解决上述痛点，需要具有以下特性：

1. 身临其境的实时场景传输
2. 良好易用的交互性
3. 一对一的服务体验

除了旅游场景外，生活应用方面还有很多其他被“模糊距离”所影响的场景，所以整个系统应该是非常开放的，例如家用机器人也可以支持接入系统，将一台家庭电脑安装上基站系统，并标记为【私人】标签，这样该基站和其下属的机器人将不会出现在公共列表中，只有你可以在你的账户空间看到并选择接入这台机器人，远程去家里处理一些事务，或者离家很远的子女们可以借助父母家中的具有 RDA 功能的机器人来跟父母进行交互，进行一些协助照顾等。

对于商用基站的建设，可以由专门的区域承包商负责公共区域的基站建设，同时一些有能力拥有自己的符合标准的 RDA 机器人和基站的组织或个人也可以将自己的基站接入系统，例如博物馆可以设立一个馆内专用基站，里面设置了定制的 RDA 机器人。

### 2.2.2 专业型 RDA

对于专业型机器人的应用场景细分，目前大概分为科研、教育、医疗三大场景。

对于专业型基站的建设，因为其要求一般比较特殊，一般由相关研究机构承建，或者选择定制服务。下面我们将借助对专业型 RDA 使用场景的探索逐步解析专业型 RDA 系统的功能要点：

#### 1. 科研领域

地球科学研究领域及生物学领域，在自然环境恶劣的地方建立科学考察站总是非常大的挑战，维持人类在极端环境下的生存需要大量的人力物力；比如南极科考站，单是建站选址就是一项复杂的工作；目前各个互联网企业都开始流行“居家办公”的方式，科学家们显然目前并不能享受随处工作的便利；如果将科学考察站和 RDA 系统结合，将搭载各种专业设备的特型机器人基站铺设到世界各地的科考点之后，科研人员将可以根据不同的任务类型选择接入不同功能的机器人展开远程实地考察，研究数据实时传输，或接入所属机构的实验室基站进行远程实验操作，与当地工作人员协同实验和讨论，这对科研人员将是一种非常灵活的工作模式，而且接入系统的基站将可以设置可见权限，拥有权限的用户都可以接入基站，这将极大促进人类科研数据资源共享，为全球科学家合作研究资源公用提供了一个极佳的媒介。

#### 2. 教育型

教育型 RDA 通常可以借用科研型 RDA，不过需要调整其操作等级和信息查看等级，因为通常科研型 RDA 的专业设备较为复杂，需要经过训练的有资质的科研人员才能熟练使用，贸然操作专业设备会导致设备损坏或基站损坏的风险，所以对于教育用途的 RDA 接入需要机器人的自动等级相应的提高；且有些科考地点环境较为复杂危险，例如月球基站，所以有必要还可以设置行动边界，这些相关接入权限由基站拥有者设置，一般都是各研究机构，将对不同等级的接入者，例如高校、中学、小学、个人接入者，做不同的权限控制。

#### 3. 医疗型

医疗型基站将主要面向紧急医疗服务，应成为城市设施的一部分，将装配搭载有应对处理各种可能的紧急医疗场景的医疗设备的机器人，基站本身也有容纳

一定数量伤员的医疗空间，和更多医疗器械，而接入医疗型基站的操纵者将不存在区域局限，可能来自全世界任何地方，但是需要具备行业资历认证才能接入（此类人称为“医疗 RDA 认证者”，简称为医疗认证者），并需承担部分必要责任；医疗型 RDA 基站在城市区域内均匀分布，当发生诸如急性心血管疾病、车祸导致的各种人体损伤等紧急医疗状况时，可以通过客户端 App 即时通知所有在线的医疗 RDA 认证者，迅速接入医疗机器人前往现场进行救援，可以移动的也可以直接将病人送往基站在基站内由医疗认证者进行远程救助。

RDA 模式本质上是实现了将人和空间位置解耦，因为很多问题需要人来解决，但问题本身也不断解决着来解决问题的人。例如一些偏远地区的医疗需求；在一些偏远的乡村、山区等地，由于人口较少以及当地通常不是很富裕的财政状况都不足以支持修建一座拥有现金医疗设施的现代化医院，但是这些地方的卫生条件往往又是最堪忧的，就算有那么一座还可以的医院，其本身各种居住条件城市设施也无法和大城市相比，无法吸引优秀人才长期驻留，像支援医生只能是一时的做法，而人力资源恰好是最关键的一环。

同类问题也出现在一些人迹罕至但同时又是比较著名的风景区的地方，比如巴丹吉林沙漠、新疆魔鬼城、喜马拉雅山珠峰大本营等地，这些地方去的人不多，且地广人稀自然环境恶劣，非常容易受伤迷失，但总是有人去，又不可能兴建大型医院，没有人会愿意长期驻留在这种地方等待几年可能都不出现一次的病患的。以上场景展示了一种矛盾性，通过引入 RDA 系统则可以很好的解决这种矛盾性；在偏远地区设立医疗 RDA 基站，而基站可以长久的存在，每年只需要很少的维护，不需要人员驻扎，但却可连接全世界的医疗人才资源，这就是通过“网络伸向现实的触手”将人类更加紧密的联系起来。

### 第三章 RDA 系统设计分析

商业型 RDA 是使用最广泛的 RDA 类型，具有一个 RDA 系统的典型特征及功能；而专用型 RDA 在通用 RDA 系统的基础上根据使用领域的不同在功能上做了相应拓展，较为复杂；因此本课题将以商用型 RDA 作为实现范例。

首先，RDA 系统网络通信模式如图：

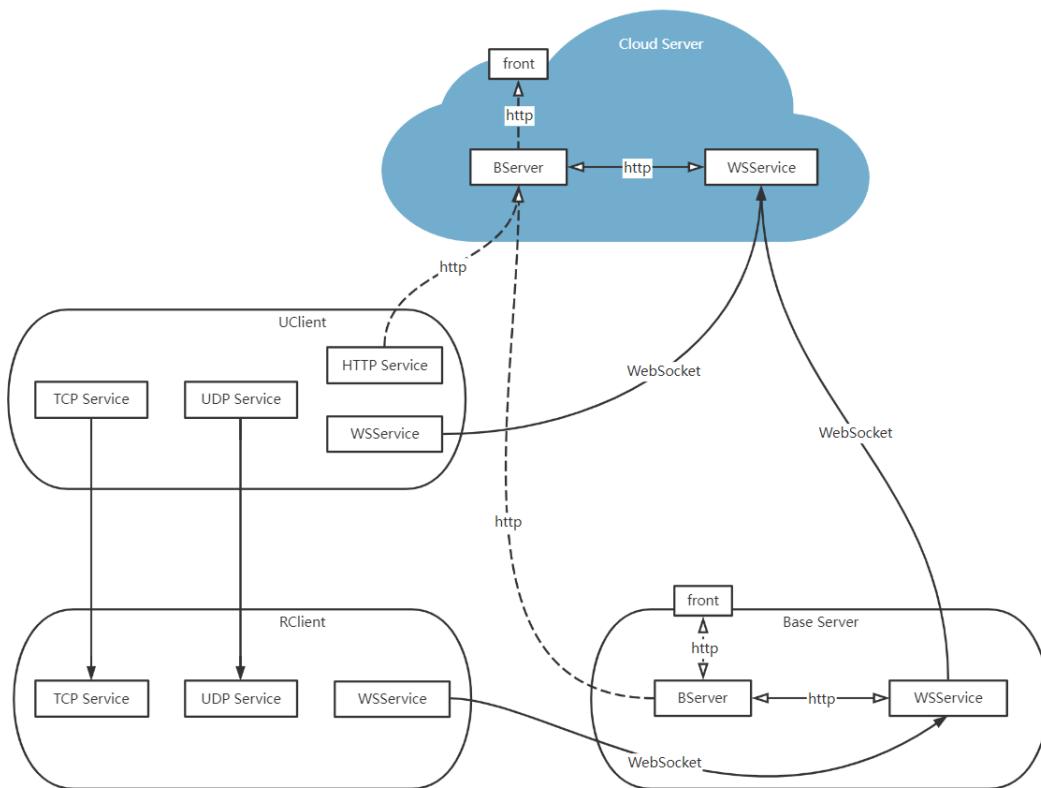


图 7 RDA 系统网络通信模式图

其中 UClient 代表用户客户端程序，RClient 代表机器人客户端程序，Base Server 为基站服务器程序，Cloud Server 为 RDA 中央云服务器。服务器程序设计采用前后端分离架构开发，前端和后端主要通过 http 协议相互通信；front 代表前端程序。用户和基站都需要登陆云服务器，并需要开启消息推送服务，允许服务器主动向客户端和基站服务器进行消息推动，云服务器可以即时将用户提交的获取机器人的请求发送给基站。用户客户端通过 UDP 协议与机器人客户端建立实时音视频通讯连接，命令传输则通过 TCP 协议连接传输；用户客户端和机器人之间的通讯不经过云服务器中转，进行高速 P2P 直连。

### 3.1 RDA 云服务器设计分析

#### 功能：

在 RDA 服务器中，用户、基站、机器人被作为三个平等的角色，有自己的账户，都需要注册，只是类别不同；除此之外，RDA 提供了一个非常自由的体系，为了维护基站和机器人的所属关系，引入了一个新的角色——Owner，即 RDA 设施所有者：

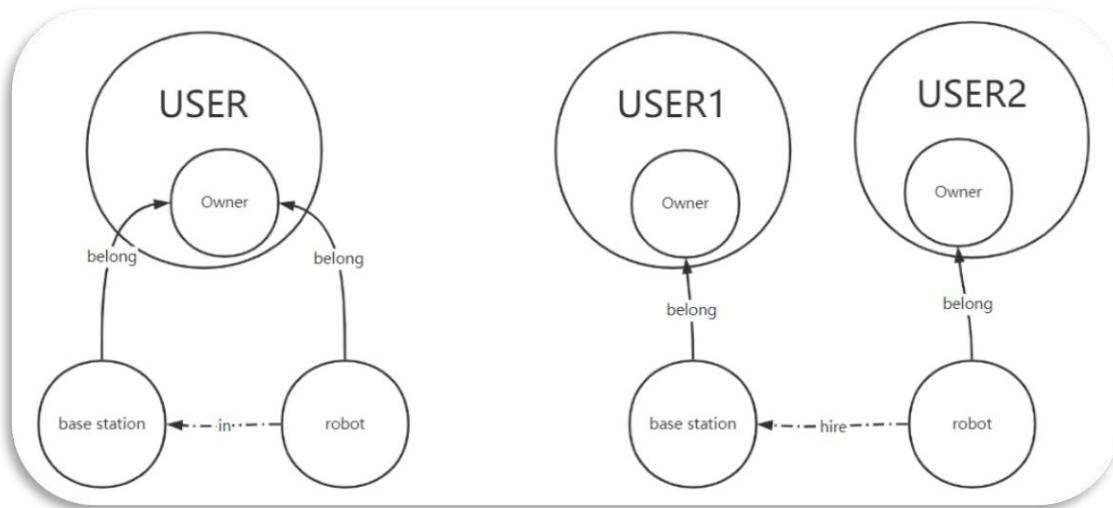


图 8 角色图

如上图，Owner 角色是跟 USER 相关联的，由 USER 衍生出，即必须先注册成为一个 RDA 用户，然后再申请 RDA 的设施所有者身份。

机器人与基站之间也不是强耦合的，如上图，有两种归属模式；一种是左边那种，机器人和基站都属于同一个 Owner；一种是右边第二种，机器人和基站分属于不同的 Owner，而机器人和基站之间是雇佣关系。不是必须同时拥有机器人和基站才能加入 RDA 系统，拥有其中一个角色，可以与另一方产生雇佣关系。

另外 Owner 的身份在这种模式下也可以灵活的转让，从一个用户账户下转移到另一个用户账户下，其拥有的角色也可以灵活的变化。这种方式将系统耦合性降低了，大大增加了自由度，可以产生出更多的市场运作形态，更灵活的适配不同的市场需求。

综上所述，RDA 云服务器将作为连接整个 RDA 世界的桥梁，需要至少具有以下功能：

1. 维护用户账户数据；包括基础账户信息、财务信息及所有者身份信息等数据；
2. 维护基站账户数据；包括基站基础信息（基站 ID、名称、地理位置等）、基站内机器人信息（型号、类型、状态等）、基站所有者等数据；
3. 维护机器人账户数据；包括机器人基础信息（机器人 ID、型号、状态）、机器人所归属基站、机器人所有者等数据。
4. 维护 Owner 链，记录历史状态。
5. 维护订单信息，记录跟踪每条历史订单。
6. 维护日志，主要是服务器和基站的日志备份。
7. 向外暴露 API 操作接口，诸如获取指定区域基站列表、获取指定基站机器人信息、登录注销、获取机器人等。

### **消息推送：**

用户、基站和云服务器之间主要通过 http 协议连接，但 http 协议是一种无状态协议，通信只能由客户端发起，如果使用 http 协议实现主动消息推动，就只能采用轮询的方式：每隔一段时间就由客户端发送一个消息询问有没有新消息；所以轮询非常浪费资源，是一种效率非常低的方式。

这次我们用到的 WebSocket 协议是基于 TCP 协议的一种面向连接的协议；建立连接时进行两次握手，握手基于 http 协议进行，因此可以通过大部分防火墙浏览器控制策略，不易被屏蔽。

我们将 WebSocket 单独设计为一个 WebSocket Service(简称 WSService)，独立于 BServer；BServer 就是 Business Server 的简称，即业务服务器，负责处理主要业务逻辑；BServer 与 WSService 之间通过 http 协议通信，成为一种松耦合结构，且方便开发时 WSService 的复用，同时增强了系统鲁棒性。

## **3.2 RDA 基站设计分析**

RDA 基站是终端机器人与云服务器连接的通道，并直接对机器人进行管理和

维护，例如充能、机体检修、物流管理（自动将用户在服务期间驾驶机器人获得的放置在机器人储物空间内物品取出并联系快递员进行邮寄）等后勤保障工作；

基站本身也可以看作一个大型机器人，基站的工作人员可以远程接入基站操作基站的一些内部机械，在基站遇到超出自动处理能力范围的情况时可以让人类接入基站为其提供协助；例如机器人检修、伤员救治的辅助等。

### 基站内机器人的连接管理

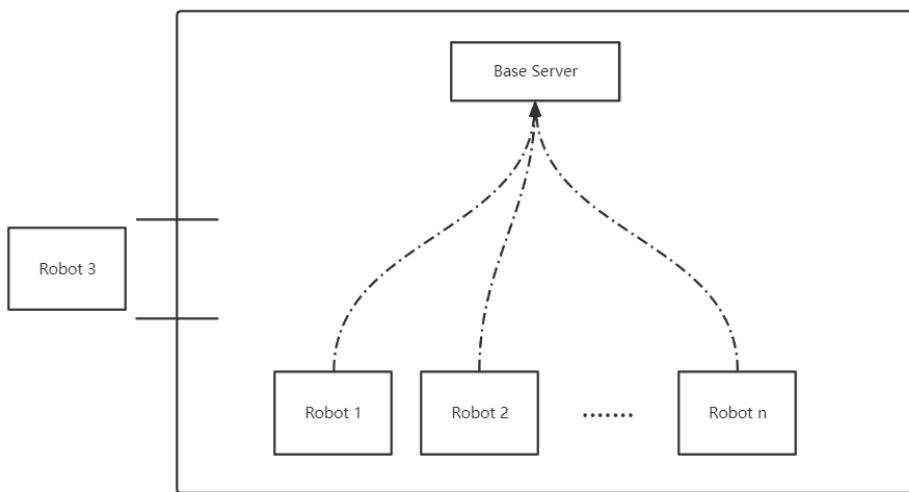


图 9

机器人进站后在连接站内 WiFi，重新登陆基站服务器，与基站服务器建立 WebSocket 连接，订阅基站的消息服务。基站服务器在机器人登陆后将更改本地机器人状态表，并向云服务器提交一个机器人状态更新消息。

在收到云服务器发送的“请求机器人”消息时，会根据消息中指明的机器人类型在本地机器人状态表中查找一个符合要求的机器人，并向其发送接入请求，收到机器人的确认回复及附带的数据包将其转发给云服务器。

机器人离站时将退出登录，断开站内 WiFi，基站将更新本地机器人状态表，并向云服务器提交一个机器人状态更新消息。

基站内机器人数可能很多，保持大量 WebSocket 连接对基站服务器也是一种比较大的负担，因此需要一种可以承载大量 WebSocket 连接的服务器部署方式。

## 基站结构设计要求

基站内应该具有以下两种角色：

1. RDA 机器人
2. RDA 维护系统
  - a) 充电
  - b) 检修
  - c) 清洗
  - d) 快递管理

基站需要满足占地空间小、可移动性强、可扩展性强、具有自我维护能力的要求。

## 3.3 RDA 机器人设计分析

机器人是 RDA 系统的末端，是网络伸向现实的触手；

### 机器人基础设计要求

每种 RDA 机器人的设计都需要考虑如何在降低成本的同时满足基本行动能力要求；

- 1) 对于通用型陆行机器人，需要特别具备快速稳定爬楼梯的能力；
- 2) 对于通用水行机器人，需要同时拥有潜水和水面漂浮两种模式；
- 3) 对于通用空行机器人，需要考虑其续航力问题，以及降低用户操作难度，同时需要考虑意外防误伤设计；

### 交互方式

机器人与人的交互是一个比较重要的问题；传统交互方式就是使用者需要完全控制机器人每一步应该如何做；另一种交互方式是协同式交互，即人只在关键时刻或机器人难以抉择的时刻介入，告诉机器人怎么做，或直接操控机器人完成。

RDA 系统因为要面向大众群体，所以需要降低操作难度，并提高系统自动化水平以避免用户操作产生的错误导致的意外状况；同时为了增强服务体验，用户

应该可以以尽可能少的命令达到想要的效果；综合考虑，RDA 系统应采用半协同式交互方式。

### **标识性设计要求：**

RDA 服务的特殊性有一点在于，机器人在代替操作者行动时，其他在现场的人看不到操作者，如果不对这一点进行规范的话，很容易被人利用进行违法活动，例如抢劫、偷拍、偷窃等；因此，机器人需要具有一种在外观上即可辨识的标识，就像汽车牌照一样；另外，一些敏感设备，例如摄像头，也需要其具有“能让人识别它在进行什么工作”的设计；例如摄像头不能隐藏，需要放置在明显的位置，且在打开录像时，机器人的会亮起指示灯，或者改变指示灯颜色。

## 第四章 RDA 系统搭建

### 4.1 WS 通用服务篇

WebSocket 协议属于应用层协议，基于 TCP 协议建立一条可靠的双向通讯链路进行消息的传递；WebSocket 建立握手的时序图如下：

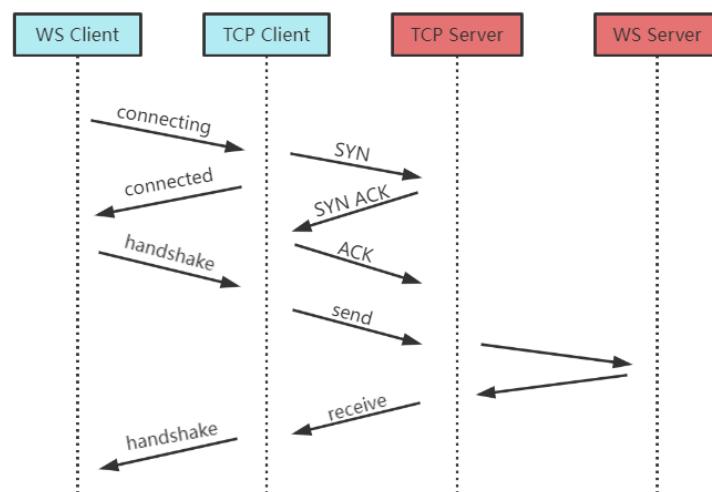


图 10

WS 服务有两个组件——Client 客户端组件和 Server 服务器组件，如图 11；只允许从 Client 端主动向 Server 端建立连接；因为云服务器和基站服务器中均需使用 WS 服务，所以将其设计为一种微服务的形式作为一项公共服务，可单独灵活部署。

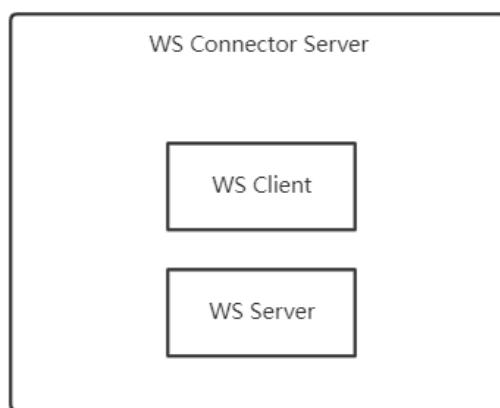


图 11

其中云服务器不需要使用 Client 端，只需要被动等待连接；基站服务器同时需要使用 WS Client 和 WS Server；不同场景下只需要开启对应服务即可。

RDA 机器人也需要 WebSocket 服务，不过机器人的 WS 服务将通过 C++编写，作为 ROS 系统的组件使用；而云服务器和基站服务器的 WS 服务使用 Spring Boot 和 Java-WebSocket 库开发。

本课题的 WS 连接服务器为搭建 WS 服务器的通用流程，以 Java 版为例。

#### 4.1.3 WS Service 部署模式

WS 连接服务器可以单机部署，也可以多机部署，支持 nginx 负载均衡；

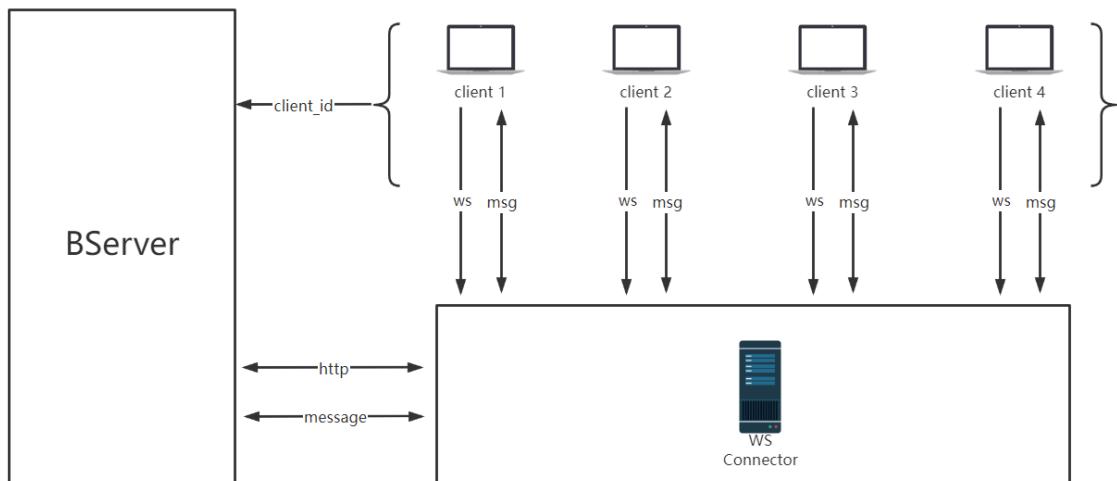


图 12 单机部署架构图

在 WS 连接量大的情况下，对单台服务器负载也是比较高的，这时可以采用可扩容的多机部署模式，多机部署架构图如图 13。

#### 4.1.4 WS 服务搭建

在实际使用场景中，会产生大量的 ws 连接，为缓解服务器压力，将采用分布式 WebSocket 服务。

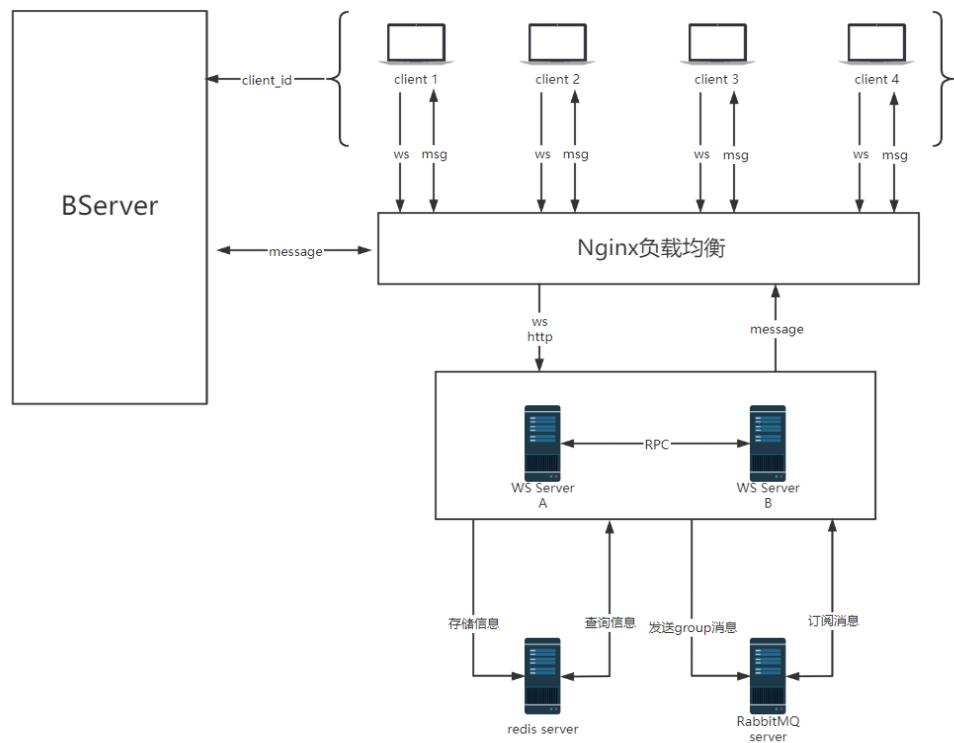


图 13 多机部署架构图

UML 时序图如图 14:

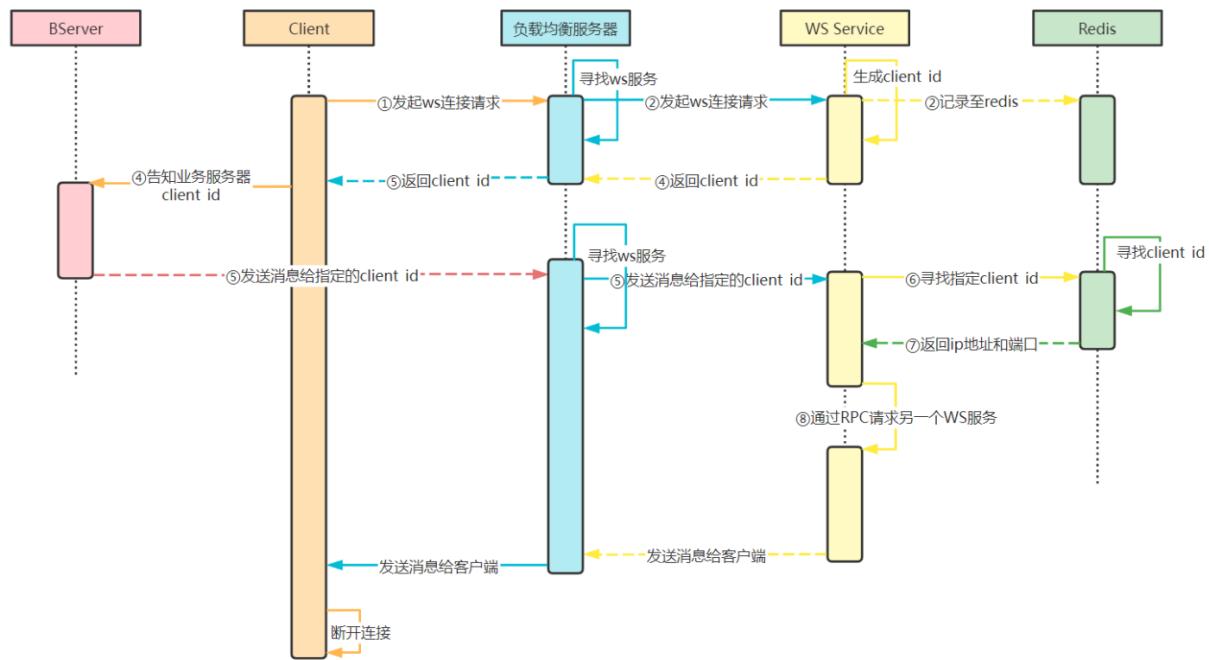


图 14

客户端建立 ws 连接时，连接请求经负载均衡服务器转发至 ws 服务器；ws 服

务器将生成一个客户端 id，并将其记录至 redis 数据库，同时将生成的客户端 id 返回给客户端，客户端将此 id 告知 BServer（业务服务器）；当 BServer 想要发送消息给客户端时，将使用 http 通过之前记录的相应客户端 id 来告知 ws 服务器要发送的客户端；ws 服务器接收到 BServer 的请求后通过此 id 查询 redis 解密获得对应的服务器 socket（IP 及端口）；随即通过 RPC 协议连接另一个 ws 服务发送消息；该 ws 服务将根据收到的客户端 id 将消息转发至相应客户端；

## 4.2 云服务器篇

此次将采用基于 SpringBoot 前后端分离架构开发云服务器；

### 4.2.2 API 接口设计

主要面向前端操作和与机器人及基站交互的 API，下面只列出关键 API 并只对某些 API 做详细标注。

#### 通用 API

##### ➤ /general / login

API 描述：面向用户、基站的通用登录接口；

参数列表：验证包【uid | bid && password】

##### ➤ /basestation / getBaseList

API 描述：用户客户端用于获取条件下的基站列表

参数列表：

(optional) **area** default: all

(optional) **baseID**

(optional) **baseName**

(optional) **baseType**

返回值：

基站列表，单元信息包括：

【基站名称 基站类型 基站各种机器人数量】

##### ➤ /robot / getRobotList

**API 描述：** 用户客户端用户获取条件下的机器人列表  
**参数列表：**

**baseID / robotID**

**paramType** 用于标注查询条件使用基站 id 还是机器人 id

**0: baseID**

**1: robotID**

**返回值：**

机器人详细信息列表，单元信息包括：

【机器人型号 机器人类型 机器人状态】

➤ [/general / getRobot](#)

**API 描述：** 用户客户端获取指定机器人连接信息

**参数列表：**

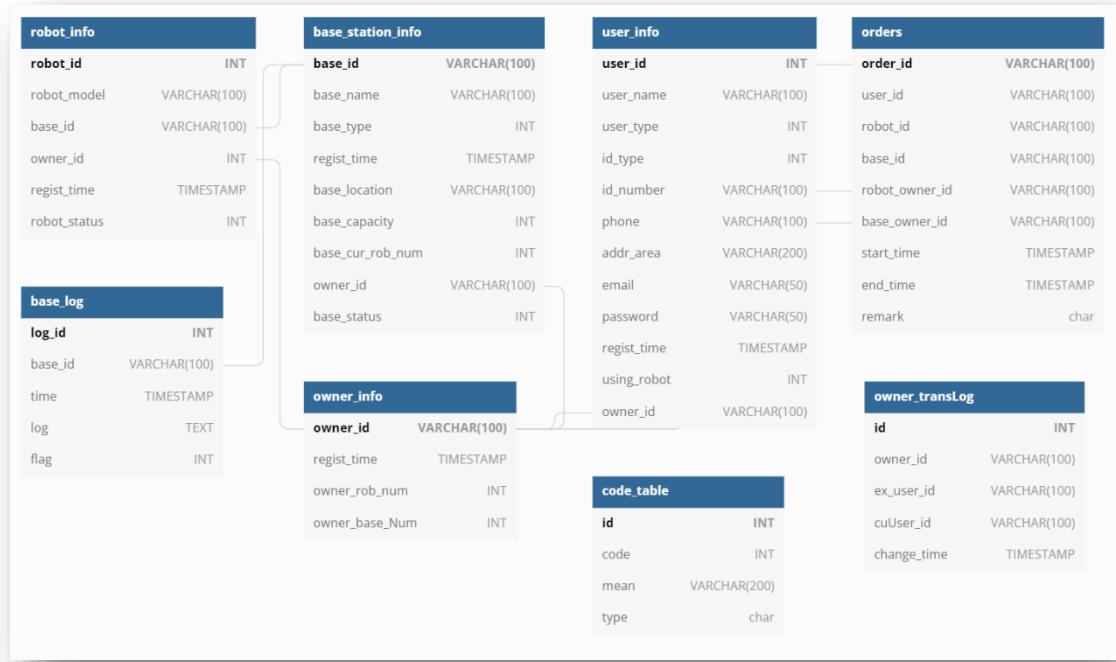
**robotID**

**返回值：**

机器人连接信息：【robot\_ipv6\_addr token】

### 4.1.3 表结构设计

云服务器部分有八张表：

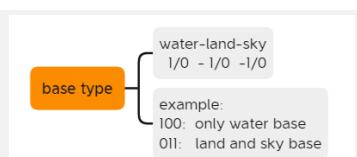


#### robot\_info——机器人信息表

字段名	含义	其他
-----	----	----

<b>ROBOT_ID</b>	机器人在注册时分配到一个 <code>robot_id</code> , 世界范围内唯一标识一个 RDA 机器人;
<b>ROBOT_MODEL</b>	记录机器人型号信息
<b>BASE_ID</b>	记录机器人所属的基站 id
<b>OWNER_ID</b>	记录机器人的所有者 id
<b>REGISTER_TIME</b>	记录机器人注册时间
<b>ROBOT_STATUS</b>	<p>机器人状态信息, 当基站传来机器人状态更新消息时更新此字段, 一共有五种状态:</p> <ul style="list-style-type: none"> <li>待上线: 机器人已注册, 但未接入系统 状态码: 10001</li> <li>工作中: 机器人正外出执行任务 状态码: 10002</li> <li>整备中: 机器人正在充电, 或正在检修 状态码: 10003</li> <li>待命中: 机器人已准备就绪, 可以接入 状态码: 10004</li> <li>待转让: 机器人正在等待转让 状态码: 10005</li> <li>故障: 机器人故障中 状态码: 10006</li> </ul>

## base\_station\_info——基站信息表

字段名	含义	其他						
<b>BASE_ID</b>	基站注册时分配到一个 <code>base_id</code> , 世界范围内唯一标识一个 RDA 基站;							
<b>BASE_NAME</b>	基站的名字, 创建基站时由基站拥有者确定;							
<b>BASE_TYPE</b>	基站类型, 对应陆地、海洋、空中三种场景, 由二进制数字的三个位标识	 <p>base type</p> <table border="0"> <tr> <td>water-land-sky</td> <td>1/0 - 1/0 - 1/0</td> </tr> <tr> <td>example:</td> <td>100: only water base</td> </tr> <tr> <td></td> <td>011: land and sky base</td> </tr> </table>	water-land-sky	1/0 - 1/0 - 1/0	example:	100: only water base		011: land and sky base
water-land-sky	1/0 - 1/0 - 1/0							
example:	100: only water base							
	011: land and sky base							
<b>REGIST_TIME</b>	基站注册时间;							
<b>BASE_LOCATION</b>	基站所在地址, 需在基站注册时说明;							
<b>BASE_CAPACITY</b>	基站可容纳的机器人数, 需在基站注册时说明;							
<b>BASE_CUR_ROB_NUMBER</b>	基站现存机器人总数量	不可由用户更改, 为动态迭代字段, 默认值为 0, 在注册机器人链接到此 <code>base_id</code> 时自动增加;						
<b>OWNER_ID</b>	基站所属的所有者 ID	在注册时自动记录, 因为注册基站必须由 Owner 角色进行注册;						
<b>BASE_STATUS</b>	基站的状态	在线: 基站正常在线;						

	离线: 基站未登录, 未与云服务器建立连接, 处于离线状态 状态码: 10010
	暂停中: 基站出于某种原因暂时暂停使用 状态码: 10020
	待转让: 基站被标记为等待转让给下一位 Owner; 状态码: 10030
	已废弃: 基站已被标记为废弃 状态码: 10040

## user\_info——用户信息表

字段名	含义	其他
USER_ID	用户注册时分配到一个 user_id, 全球范围内唯一标记一位用户	
USER_NAME	用户名由用户注册时填写	在将来与元宇宙结合的应用场景中, 用户名也可以直接与相应元宇宙中的数字身份关联;
USER_TYPE	用户类型, 目前有三种	<b>Personal:</b> 个人账户 代码: 10100 <b>Group:</b> 集体账户 代码: 10200 <b>Enterprise:</b> 企业账户, 注册时需要相关认证 代码: 10300
ID_TYPE	注册时添加的身份认证方式种类, 例如护照、身份证等	护照代码: 10301 身份证代码: 10302
ID_NUMBER	对应 id_type 的证件号码	
PHONE	绑定的手机号	
ADDR_AREA	用户所在地	
EMAIL	邮箱	
PASSWORD	密码	
REGIST_TIME	用户注册时间	
USING_ROBOT	该字段记录用户有没有正在使用机器人, 默认值为 0, 不可人为更改	
OWNER_ID	当用户申请成为一个 Owner 时, 分配到的 owner_id 将被记录在用户信息表里	

## owner\_info——所有者角色信息表

字段名	含义	其他
OWNER_ID	用户注册成为 Owner 时自动分配一个 owner_id	
REGIST_TIME	Owner 角色注册时间;	记录时间戳
OWNER_ROB_NUM	Owner 下属的机器人数量, 用于必要时相互印证的数据;	用户不可操作此字段, 当注册机器人时关联此 owner_id, 该字段将自动增

<b>OWNER_BASE_NUM</b>	Owner 下属的基站数量，用于必要时相互印证的数据；	加用户不可操作此字段，当注册基站时关联此 owner_id，该字段将自动增加
-----------------------	-----------------------------	--

### owner\_translog——所有者变更记录表

字段名	含义	其他
<b>ID</b>	一条记录的 id	
<b>OWNER_ID</b>	涉及的 owner_id	所有者发生变更时，owner_id 本身不变
<b>EX_USER_ID</b>	所有者身份之前所属用户 id	
<b>CU_USER_ID</b>	所有者身份现在所属用户 id	
<b>CHANGE_TIME</b>	变更发生的时间	记录时间戳

### orders——订单记录表

字段名	含义	其他
<b>ORDER_ID</b>	每条订单的 id 号	
<b>USER_ID</b>	该条订单的用户 id	
<b>ROBOT_ID</b>	该条订单用户雇佣的机器人 id	
<b>BASE_ID</b>	该条订单用户雇佣机器人所属基站 id	
<b>ROBOT_OWNER_ID</b>	该条订单用户雇佣机器人的所有者 id	
<b>BASE_OWNER_ID</b>	该条订单用户雇佣机器人所属基站的所有者 id	
<b>START_TIME</b>	订单开始时间	
<b>END_TIME</b>	订单结束时间	
<b>REMARK</b>	标记字段，可为空	

### base\_log——基站日志备份表

字段名	含义	其他
<b>LOG_ID</b>	每条日志的 id	
<b>BASE_ID</b>	提交该条日志的基站 id	
<b>TIME</b>	提交该条日志的时间	记录时间戳
<b>LOG</b>	该条日志的内容	
<b>FLAG</b>	可为该条日志打上标记，便于以后需要时快速查找；	

基站每隔一段时间就会向服务器提交一次日志，内容主要包括基站运行情况、机器人运行情况、故障率、各机器人状态等；

### code\_table——系统代码含义对照表

字段名	含义	其他
ID	该条代码的 id	
CODE	该条代码号	例如: 10001
MEAN	该条代码对应的含义	例如: 待上线, 机器人已注册但未接入系统
TYPE	该条代码的类型	

该表记录了所有系统代码对应的含义;

#### 4.1.4 业务服务器搭建

下面只对两个典型模块做展示和简要说明。

获取基站列表 UML 时序图:

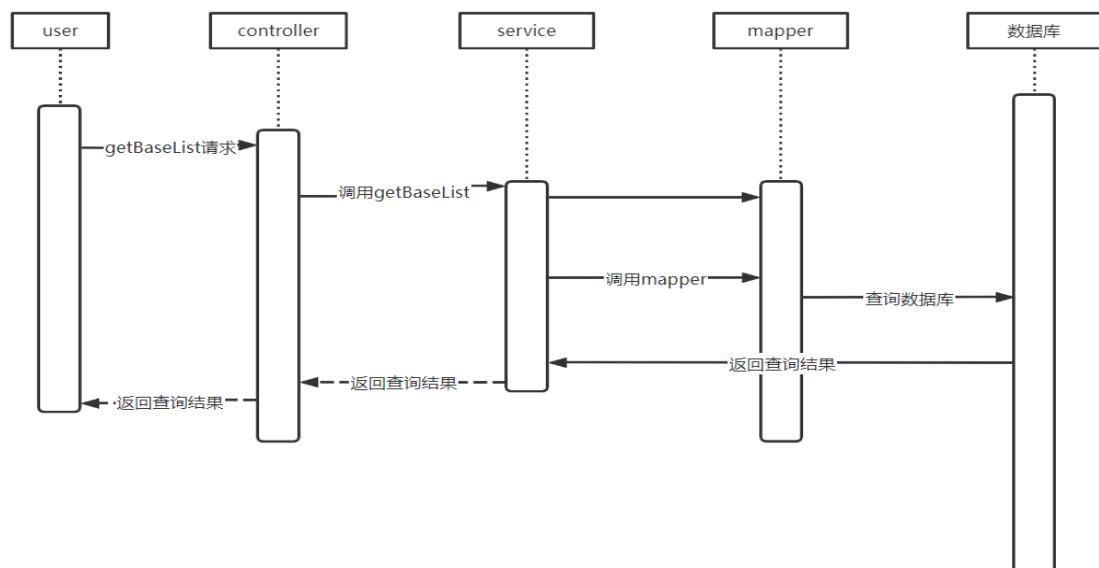


图 15

获取机器人 UML 时序图:

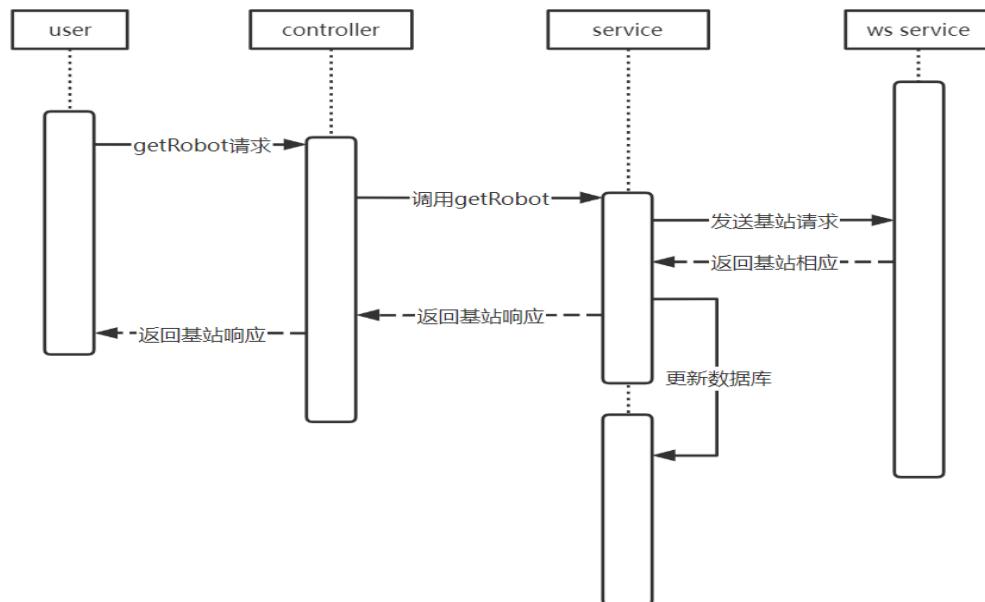
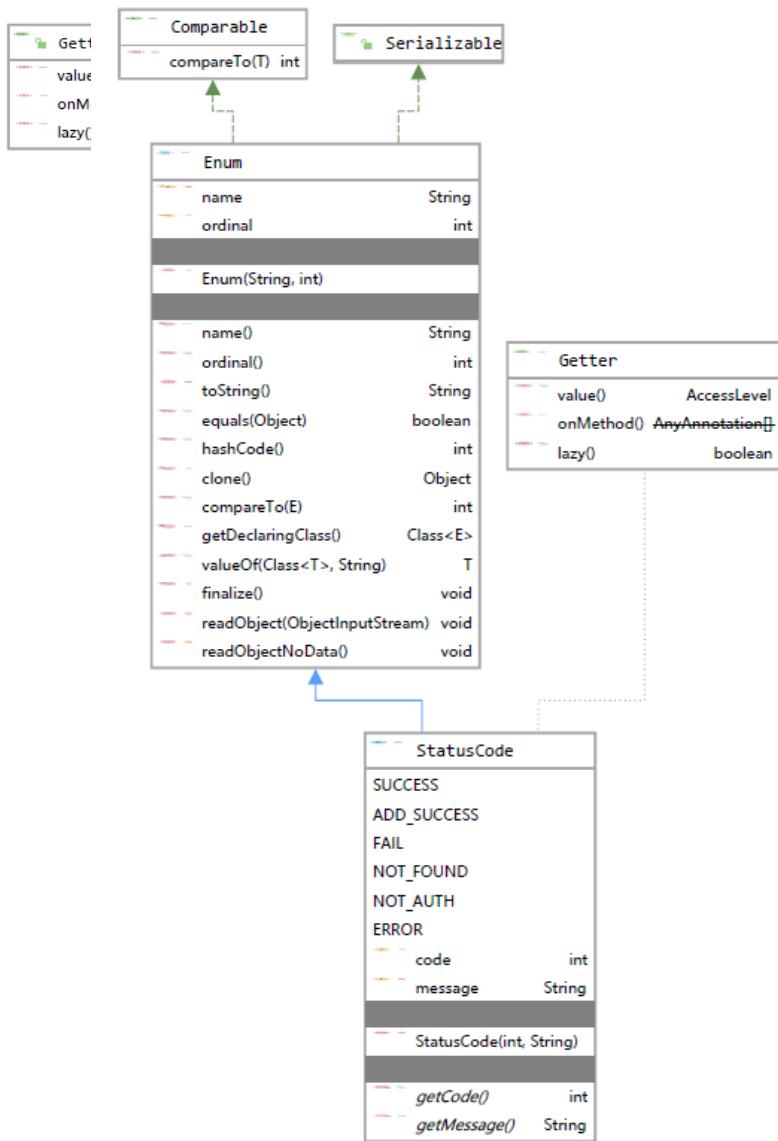


图 16

## 自定义 ResponseResult 响应消息主体



## 使用 CORS 解决跨域问题

跨域问题是浏览器的同源策略引起的；同源策略是浏览器一种非常关键的安全机制，但因为我们使用的是前后端分离的架构，因此前端通讯是分离的，需要通过一些应用层协议例如 http 进行交互；分离了之后就会产生跨域问题，简而言之就是发出请求的一方与接受请求的一方由于端口、协议、域名三个中的任意一个不同而被浏览器的同源策略拦截导致前端接收不到后端

回传的请求，所以后端程序是收到前端发来的请求并且回复了的，只是浏览器不允许脚本程序读取数据。

本质上是 http 头中缺少了一些字段，例如 `Access-Control-Allow-Origin` 字段缺失；通常使用 CORS 即 Cross-origin resource sharing——跨域资源共享协议来解决问题，更改 http header 字段值。

只需要在服务器中注入一个 `CorsConfig` 的 Bean，设置 header 中的以下内容：

- |  |          |
|--|----------|
| ■ <code>Access-Control-Allowed-Origin</code> | ———— “*” |
| ■ <code>Access-Control-Allowed-Header</code> | ———— “*” |
| ■ <code>Access-Control-Allowed-Method</code> | ———— “*” |

- Access-Control-Allowed-Credentials —— True

## 4.2 基站篇

### 4.2.1 基站服务器

#### (3) 表结构设计

基站服务器涉及到的表结构比较简单，如下图 17：

robot_info		base_info	
robot_id	VARCHAR(100)	base_id	VARCHAR(100)
robot_model	VARCHAR(100)	owner_id	VARCHAR(100)
robot_status	INT	base_type	INT
robot_v6_addr	VARCHAR(100)	base_capacity	INT
robot_log	text	base_location	VARCHAR(100)

图 17

只涉及到两个表，机器人表和基站表；

robot\_info:

字段名	含义	其他
ROBOT_ID		
ROBOT_MODEL		
ROBOT_STATUS		
ROBOT_V6_ADDR	记录机器人的临时 ip 地址，及时更新	
ROBOT_LOG	备份每个机器人的日志信息	

base\_info:

字段名	含义	其他
<b>BASE_ID</b>	基站自己的 id	
<b>OWNER_ID</b>	基站记录所有者 id	
<b>BASE_TYPE</b>	基站类型	
<b>BASE_CAPACITY</b>	基站容量	
<b>BASE_LOCATION</b>	基站定位地址	

## (4) 业务服务器搭建

### 4.2.1 基站结构概念设计

如第一二章所属，基站所对应的机器人类型有三种——陆行、水行、空行，装载维护每种机器人需要不同的对应的设施；而基站本身也是一个并不小的工程，为了将其应用场景变得更加灵活，同时可以更加灵活的更新迭代，对基站的设计可以按装载的三种机器人类型而进行模块化设计：

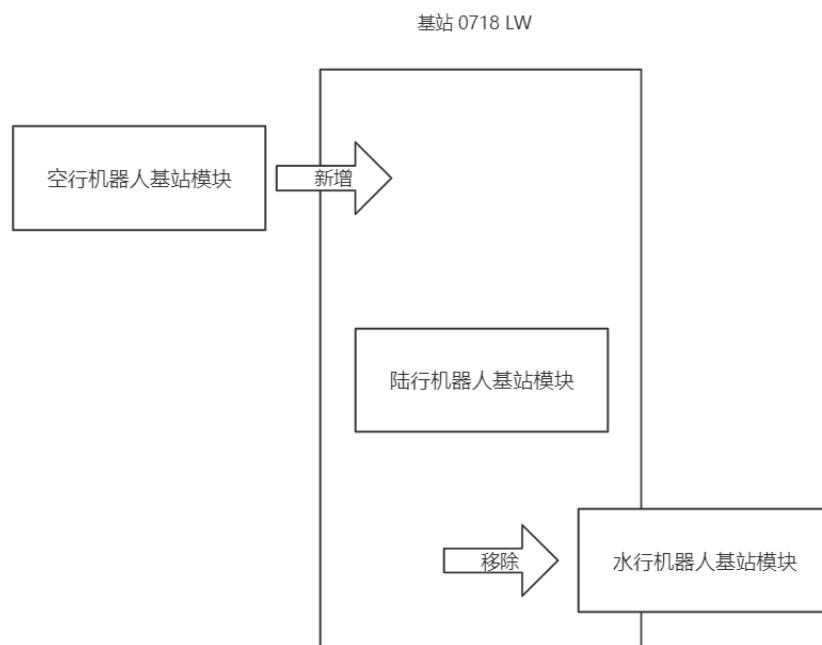


图 18

这种模式下，基站的从属关系就可以细分到每个基站模块：

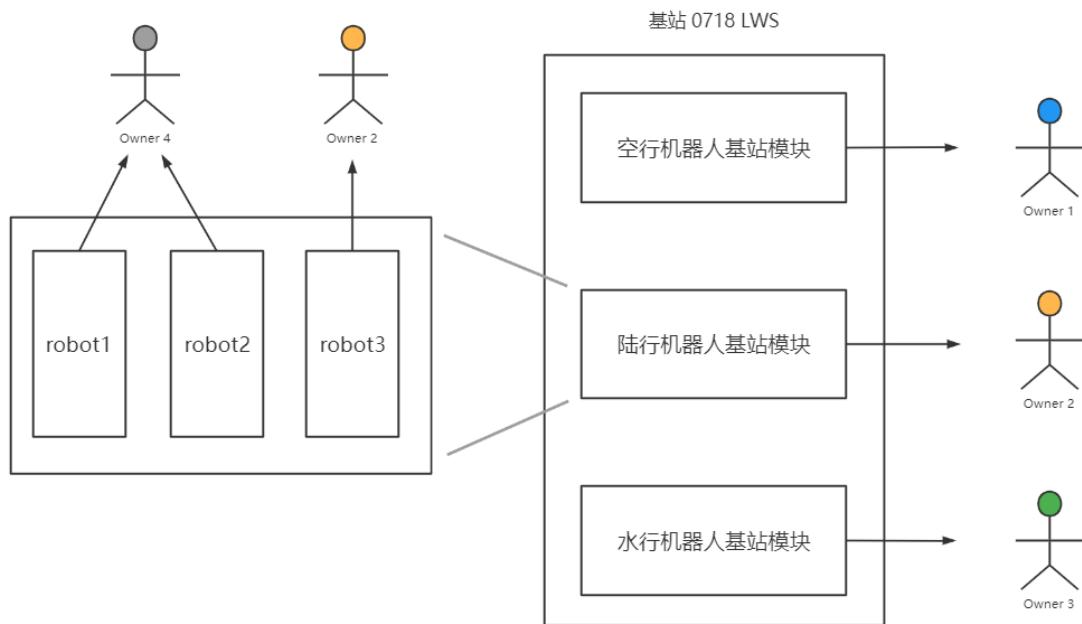


图 19

如图 20，基站 0718 LWS 是一个水陆空三栖基站，同时拥有三种类型的基站模块；其中空行基站模块是 Owner1 提供的，陆行基站模块是 Owner2 提供的，水行基站模块是 Owner3 提供的；陆行基站模块内的 robot1 和 robot2 所属于 Owner4，robot3 是 Owner2 所有。

### 基站概念设计图：

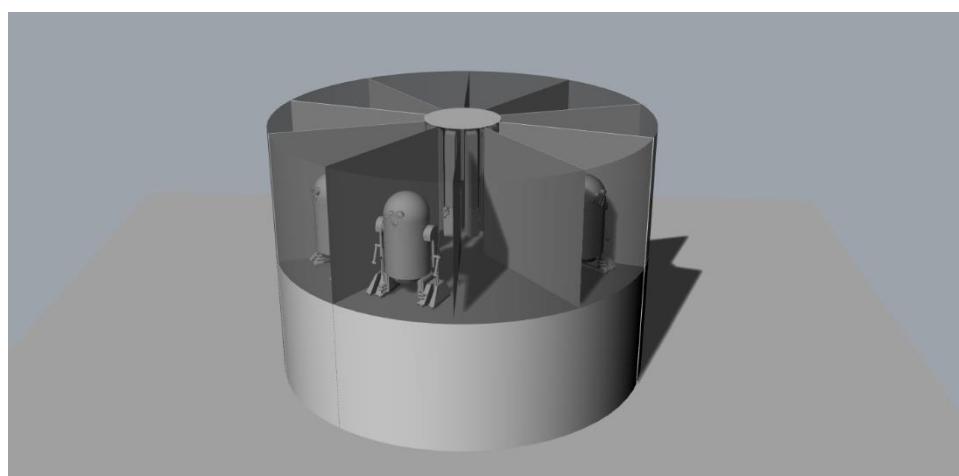


图 20 两层陆地基站

图 20 为一个两层陆地基站概念图，每层为一个模块单元，可单独拆卸；每个单元划分为几个格，每个格存储一个机器人和负责机器人日常检修的机械臂，

如图 21, 及其他设备。

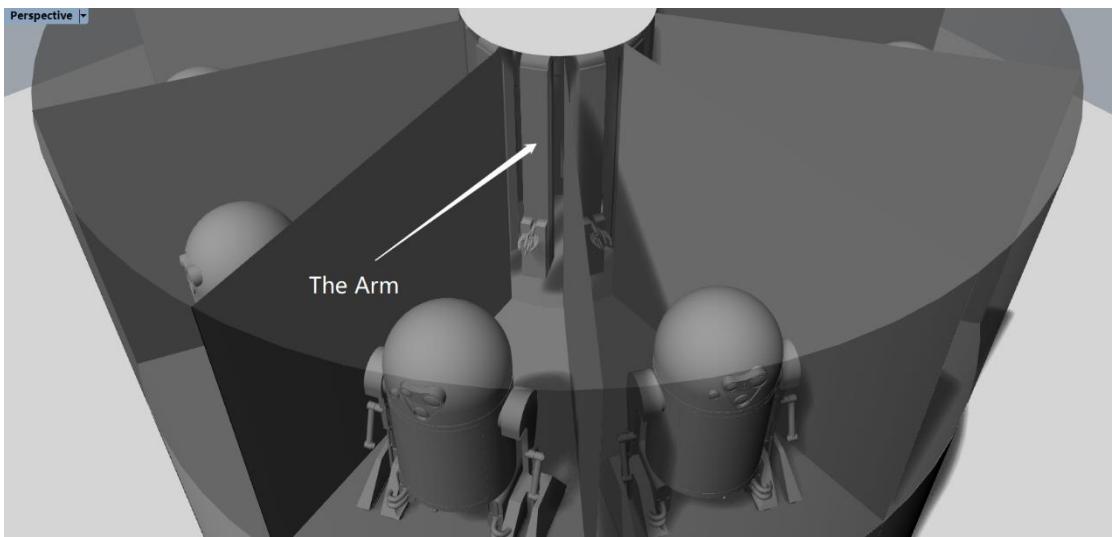


图 21 单元格机械臂位置示意图

## 4.3 机器人篇

### 4.3.1 机器人客户端

机器人客户端通过 ros 系统作为通信框架; ros 系统即机器人系统, 其实并不是一个系统, 本身是依附于 Linux 存在的, 是一种通信框架, 为代码提供了一个管理架构, 这种依附于其他系统存在的操作系统被称为“元操作系统”。

机器人本身是一台拥有诸多传感器的电脑, 多传感器通讯就是机器人的核心平台; 多设备通信的话, 一个很重要的影响因素就是数据同步问题, 需要一个统一的时钟系统进行管理, 否则数据也不能进行融合。

而 ros 系统为设备间通讯提供了一个统一的调度中枢即 ROSCORE, 另外为不同语言的代码提供了一个公共框架, 允许不同的语言开发的程序协同工作。

#### (1) 系统框架

ROS 系统将机器人的不同功能定义为不同的节点(node, 就是可执行程序),

节点之间通过消息传递进行通讯，ros 创建了一个网络可以连接到所有进程，为机器人的功能节点提供了一个较为松散的耦合方式。

以 R2L2-L1 型陆行机器人为例进行设计，其简要硬件架构图如图 22：

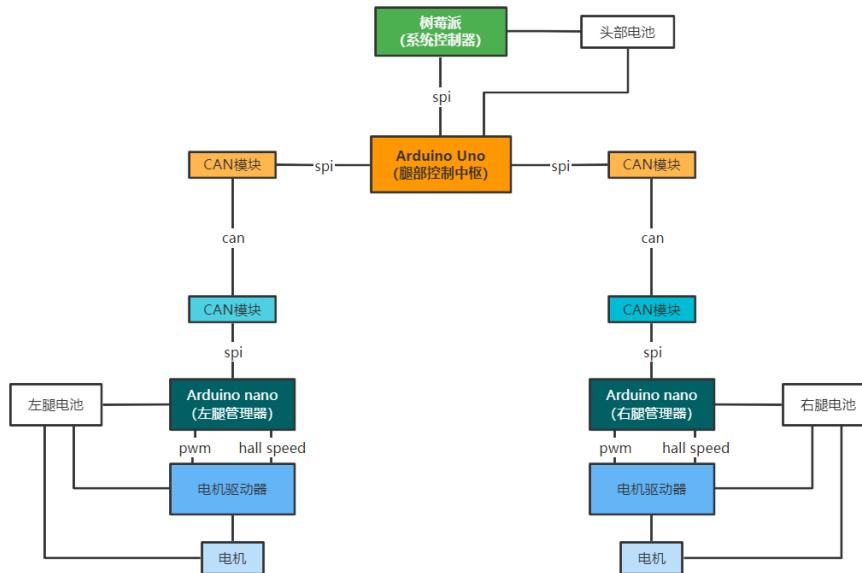


图 22

整个机器人硬件上也是采用了一种松耦合架构，左右两条腿为两个独立功能单元，具有自己独立的管理器和电源；管理器主要负责根据运动中枢回传的指令控制驱动器进行调速，并提取反馈速度信息；同时监控电池状态，生成日志返回给系统控制器。

基于这种架构设计 ros 系统节点计算图：

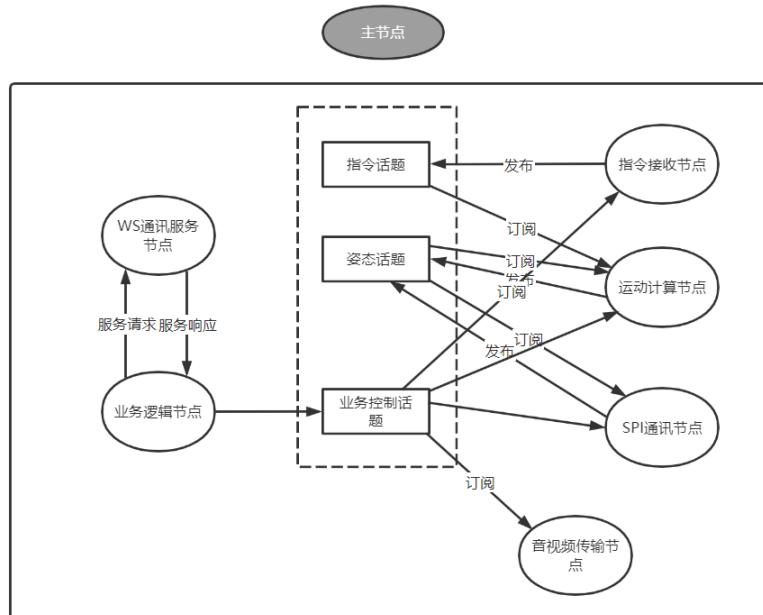


图 23

## (2) 基于 UDP 的 P2P 实时视频传输模块

使用 OpenCV 库获取实时图像，并进行 JPEG 压缩，基于获取的 ipv6 临时地址和端口构建 socket 套接字，包装成 UDP 数据报文发送给用户客户端；

功能节点 UML 类图：

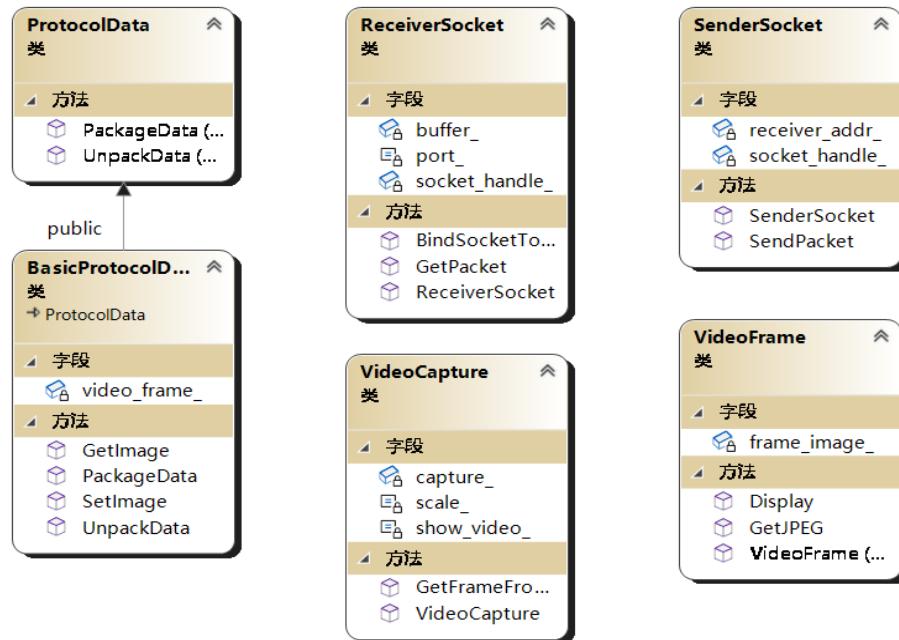


图 24

测试：

```

pi@raspberrypi:~/codehub/rClient/build $ ./rClient
socket created ...
socket id : 3
remote ip : 2409:8a3c:8205:db30:189a:58ff:4cc0:5380
remote port : 8888
data size : 12464 send: 1
data size : 12452 send: 1
data size : 12444 send: 1
data size : 12457 send: 1
data size : 12437 send: 1
data size : 12447 send: 1
data size : 12446 send: 1
data size : 12469 send: 1
data size : 12476 send: 1
data size : 12430 send: 1
data size : 12446 send: 1
data size : 12460 send: 1

```

获取到的临时ipv6  
地址

图 25

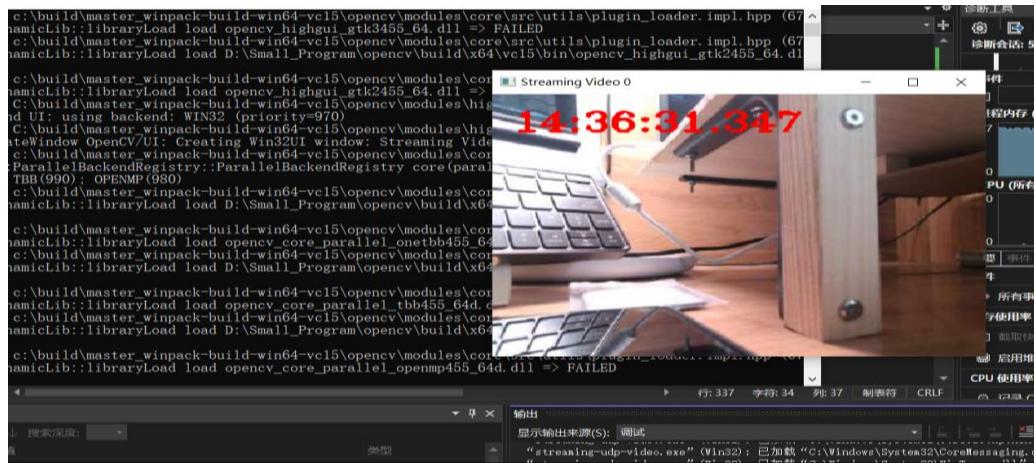


图 26

测试时延可以维持在 3ms 内。

### (3) 基于 TCP 的 P2P 指令传输模块

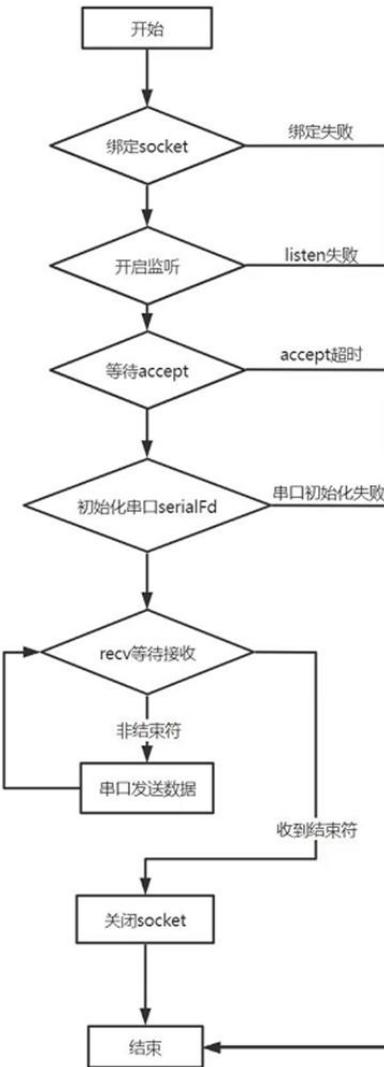


图 27

用户控制机器人的方式有多种，例如键盘控制、手柄控制、通过 VR 眼镜的手势识别控制等；关键因素就在于使用户的操作简单易理解且精确。此次演示系统中采用一套很简单的命令格式，如下图 28：

偏转角+速度等级	$(180 - 2 * \text{sitar}) / 180 = \text{左/右轮速度值占比}$
sitar范围: $[-90^\circ, 90^\circ]$	0101 1010-1111-1 最后一位标志位, 0左, 1右

图 28 简单控制指令格式

命令通过偏转角和速度等级控制机器人移动方向；偏转角定义为单位向量从 x 轴正方向开始逆时针转动与 x 轴正方向的夹角，范围是  $0-90^\circ$ ；通过第一位标志位控制方向，为 0 的时候为向左转，为 1 的时候向右转。以二进制形式表示的

指令体积非常小，易于解析，机器人收到一条指令直接通过位运算即可解析。计算流程图如图 29：

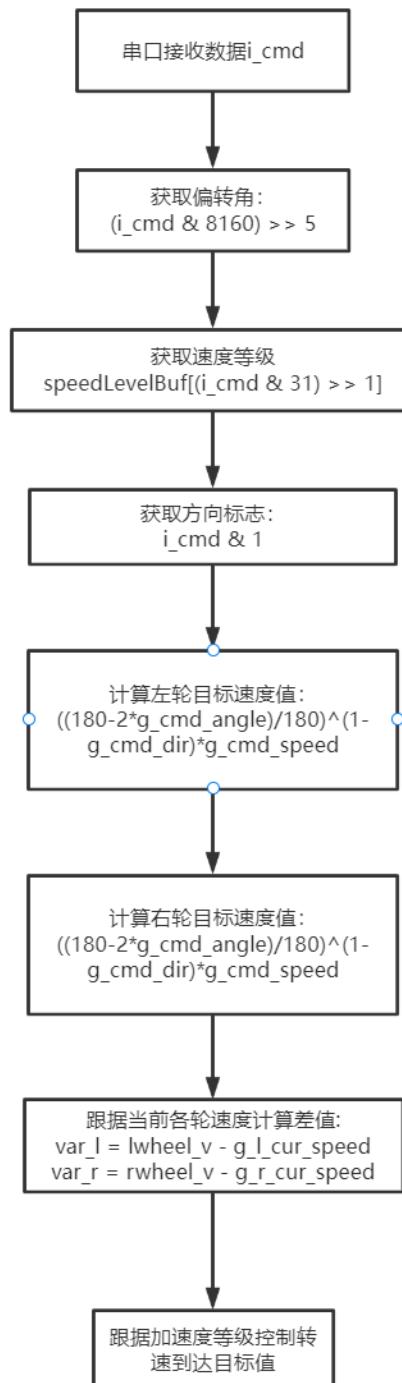


图 29 指令计算流程图

### 4.3.2 机器人概念形态设计及要点

#### (1) 陆行机器人

如图 30, 它的是以星球大战里的宇航技工机器人 R2D2 为原型设计, 是我正在进行中的另一个项目——R2L2, 在此项目中仅作为展示用例, 不作为商业用途, 也不是最终版, 在 RDA 系统中型号假定为 R2 L2-L1:

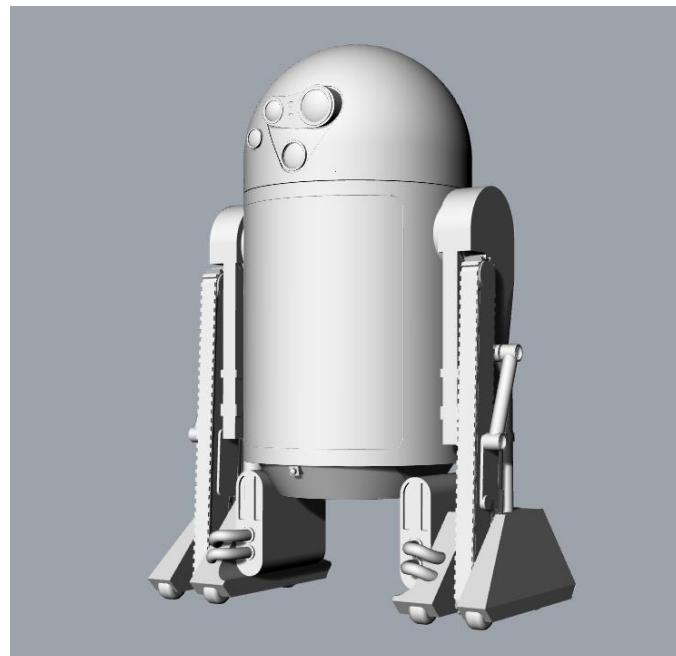


图 30 R2L2-L1

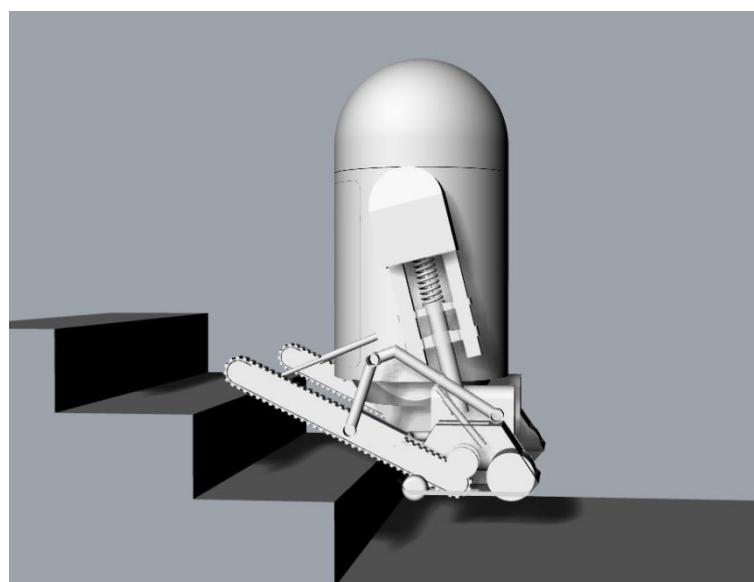


图 31 R2L2-L1 上楼梯的示意图

R2L2 重新设计了机器人的腿部结构和身体内部结构; 在腿部的履带可以让

其进行爬楼梯、增加对地形的适应性，如上图 31。

将空间充足的身体部位内部设计为中空储物箱，划分四个扇区，如图 32，可以转动，可以在 RDA 服务期间便捷的储存用户获得的物品。

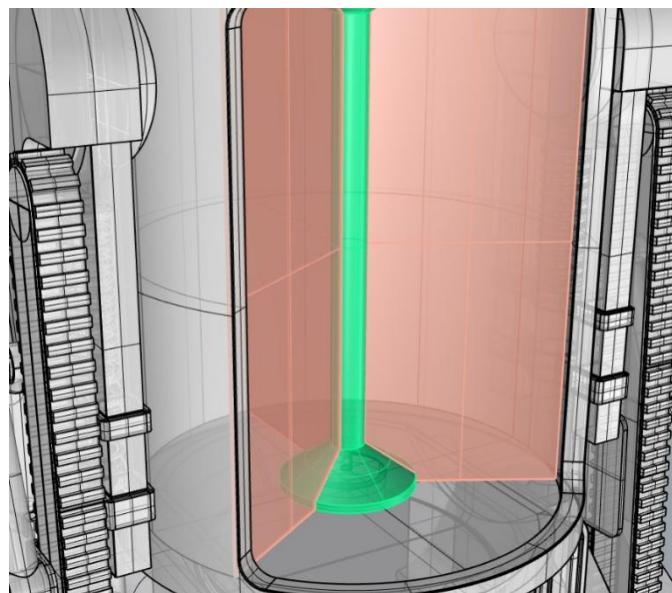


图 32 存储空间示意图

#### (4) 空行机器人

空行机器人部分目前可以有两种方案，一种是传统的旋翼式无人机，如图 33 所示大疆 FPV 穿越机。



图 33 DJI FPV 无人机

旋翼型无人机的优点在于其灵活性和速度，而不足之处在于续航和安全性。

小型无人机普遍续航在半小时以内，而且安全性方面，旋翼式无人机飞行速度快，操纵性要求高，裸露的旋翼有对人造成伤害的风险；

为了解决续航能力和安全性的问题，设计了第二种空行机器人——气球型无

人机——Bubble-A1：

**Bubble-A1**

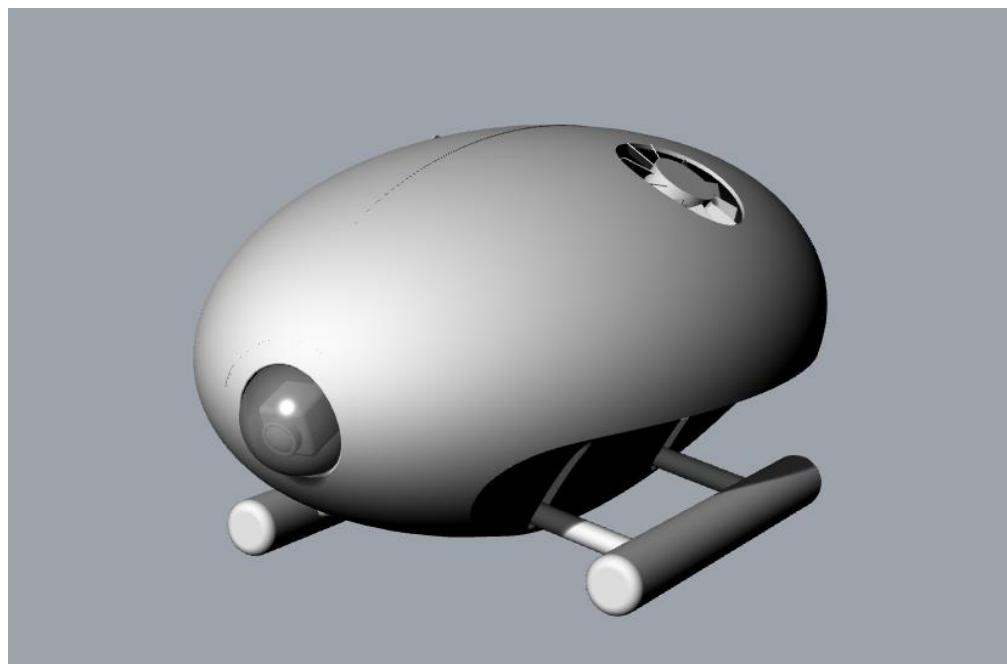
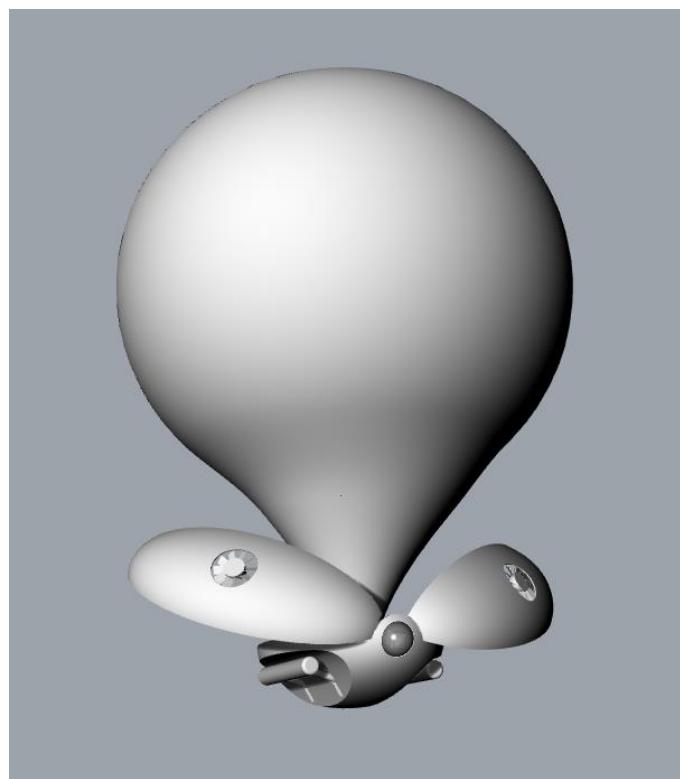


图 34 Bubble-A1 型气球机器人概念图



使用时顶部气囊仓打开，  
基站为机器人进行充气，服务  
结束后，机器人回到基站，气  
囊自动放气收纳回气囊仓；气  
球型机器人在大风天气禁止  
使用。

图 35 Bubble-A1 气囊展开 飞行模式图

### (3) 水行机器人

在此展示一种通用型号的水行机器人—— CUTTLEFISH – W1：

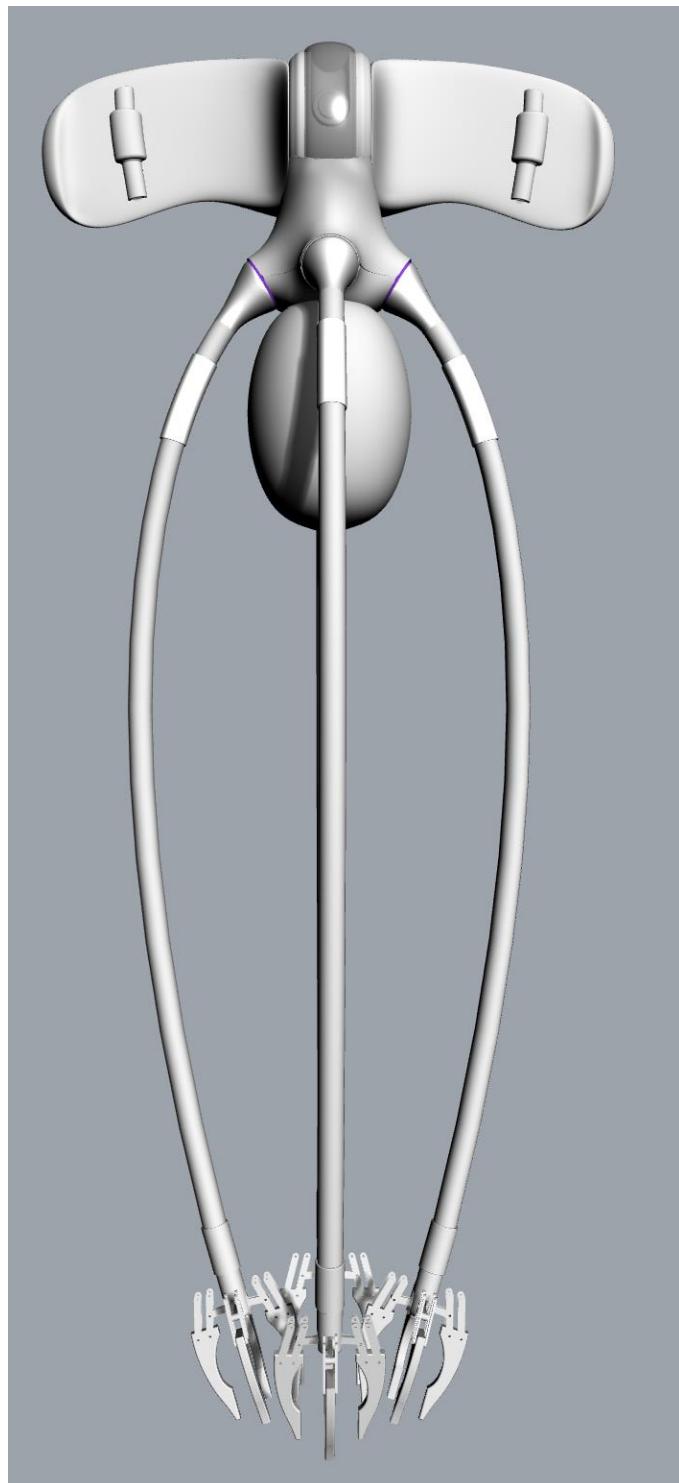


图 36

具有四根长触手型机械臂，极大的提高了灵活度的同时，还可辅助其进行高速游动；另外，四根长触手在机器人做应用在科研领域的改装时，可以挂载多种

设备，并且可灵活移动，深入狭小空间作业。

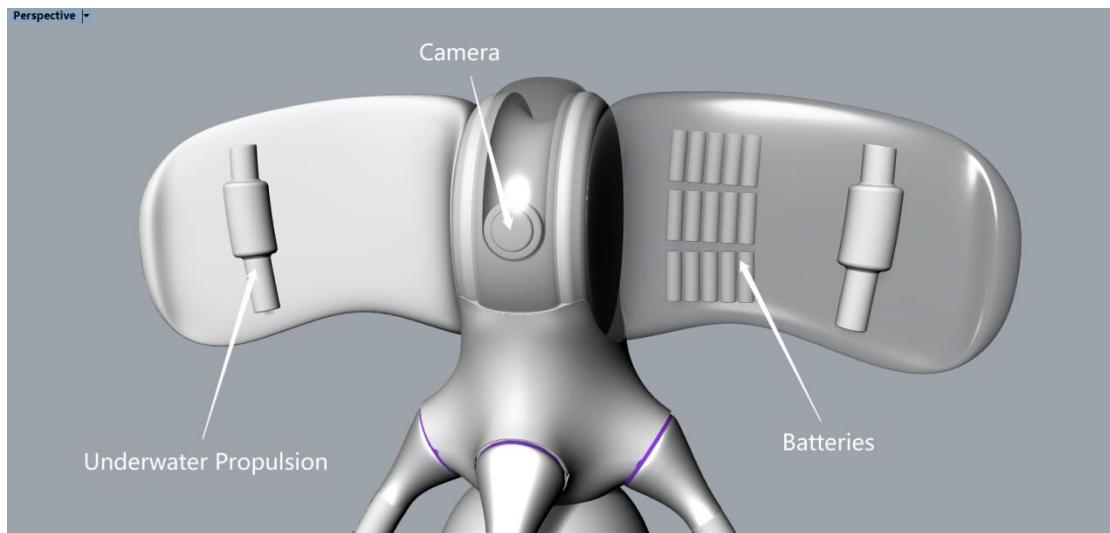


图 37

图 37 简单展示了身体部分结构布局，其中电池组分布在两翼根部，推进器位于两翼末端，尾部挂载椭球形储物仓；镜头组位于中间的玻璃罩内，可 180° 纵向转动，如图 38。

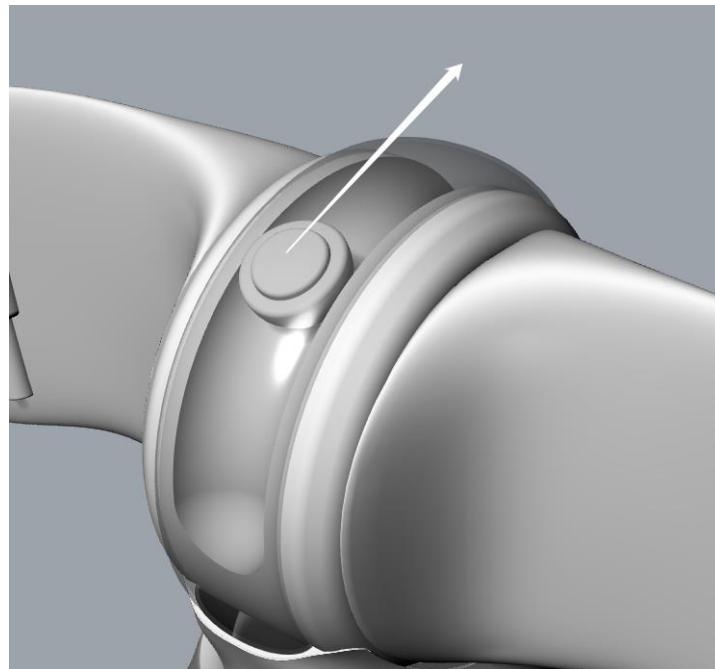


图 38

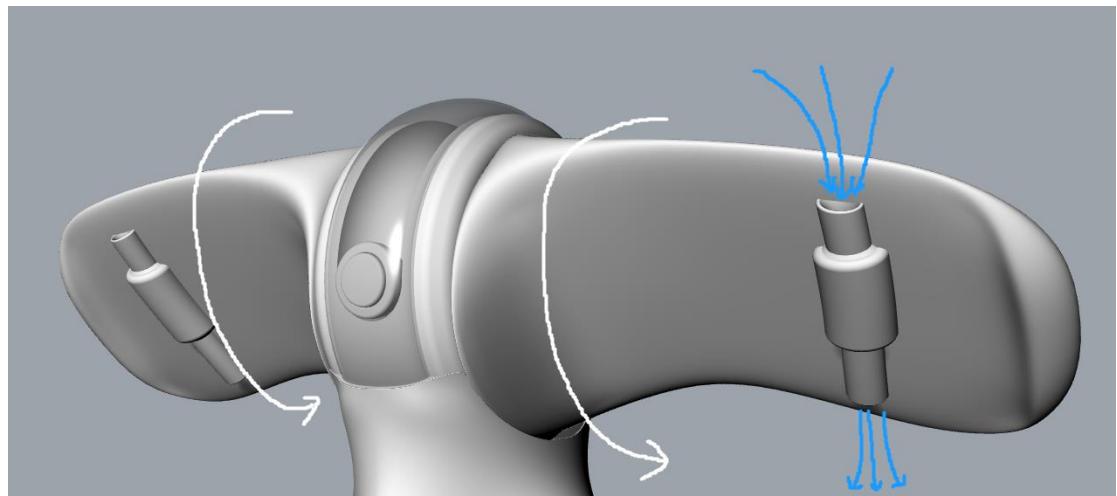


图 39

两侧的滑行翼可以  $360^{\circ}$  转动，以满足不同的移动模式；而每根机械臂根部有一个折叠充气囊，用于在水面的漂浮模式，充气张开效果如图 40：

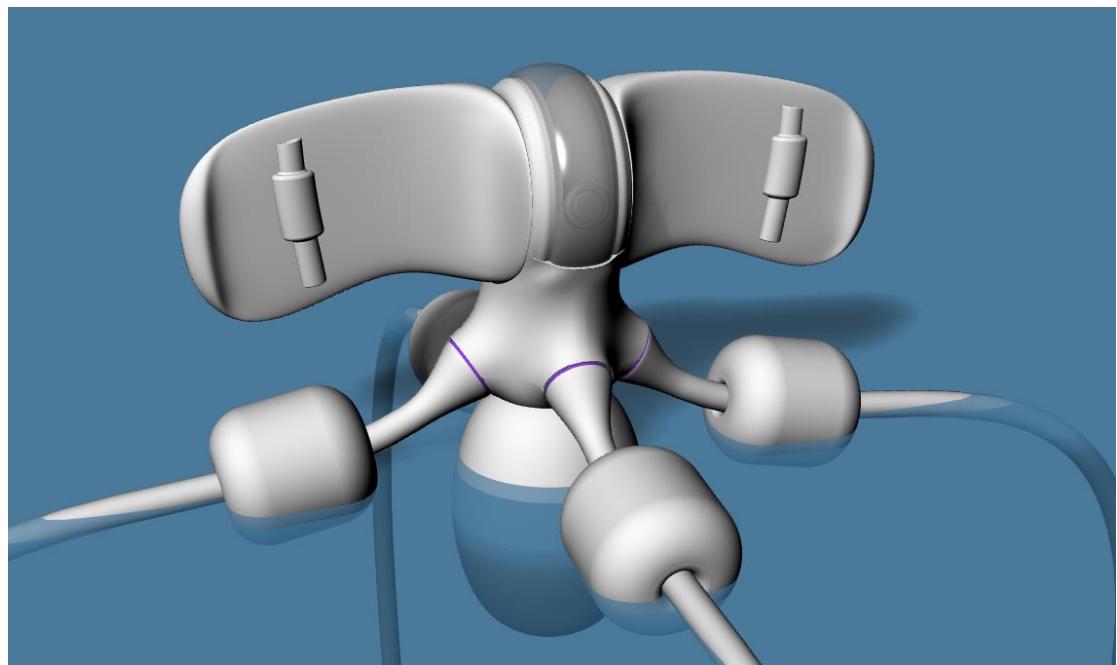


图 40 漂浮模式

#### 4.4 用户客户端篇

本次实验性搭建的前端程序和客户端程序采用“人力前端/客户端”模式，即因为时间原因没有通过一个 GUI 框架整合，有些部分直接用 postman 发送 http 请求操作 API。

#### 4.4.1 用户操作 UI

##### 用例图

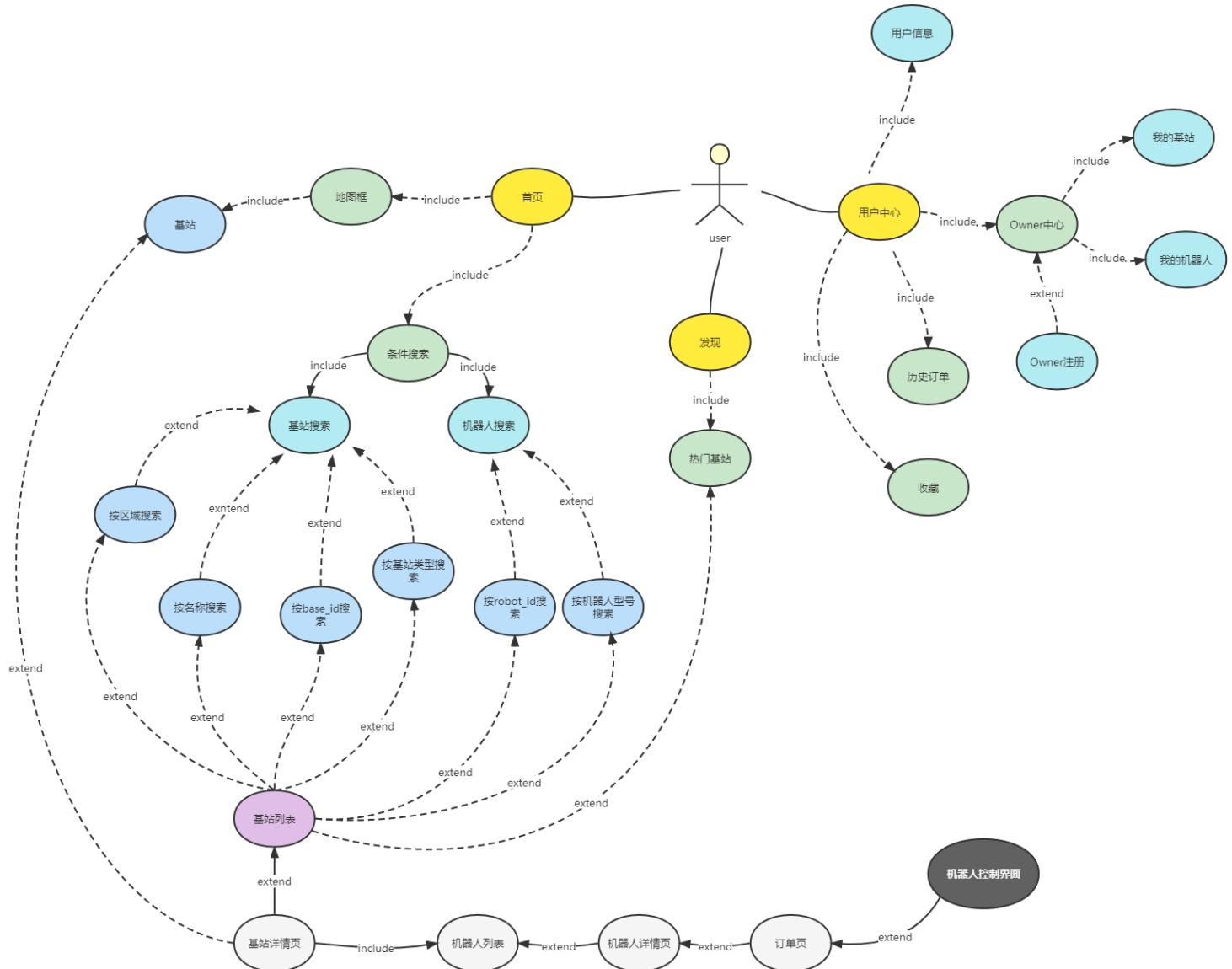


图 41 客户端 UML 用例图

#### 4.4.2 基于 UDP 的 P2P 视频传输接收模块

基于 Windows 系统开发的用户客户端，通过 ws2tcpip 库使用 TCP 和 UDP socket 通信。

客户端视频接收模块 UML 类图如图 42：

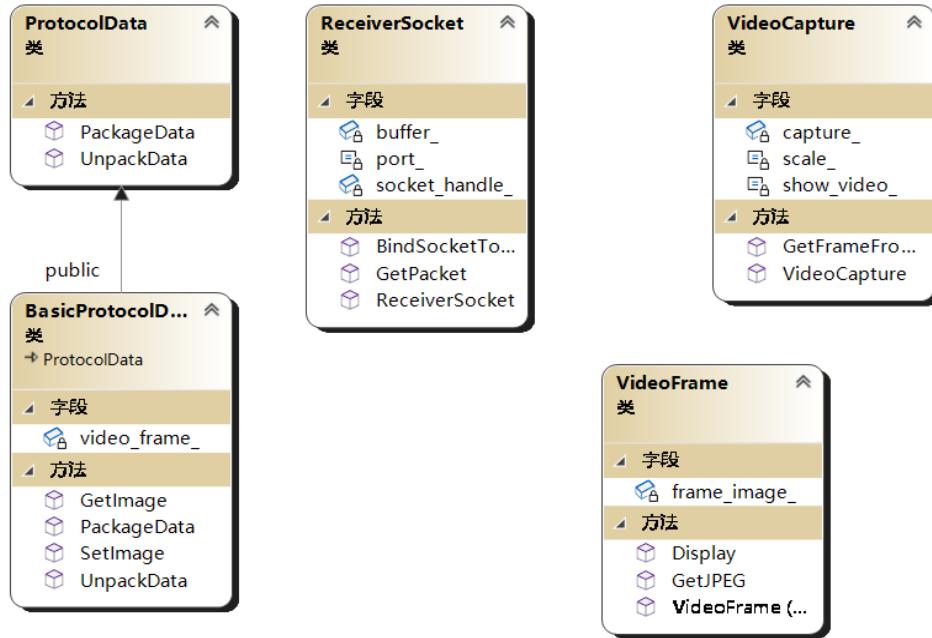


图 42

### 测试：

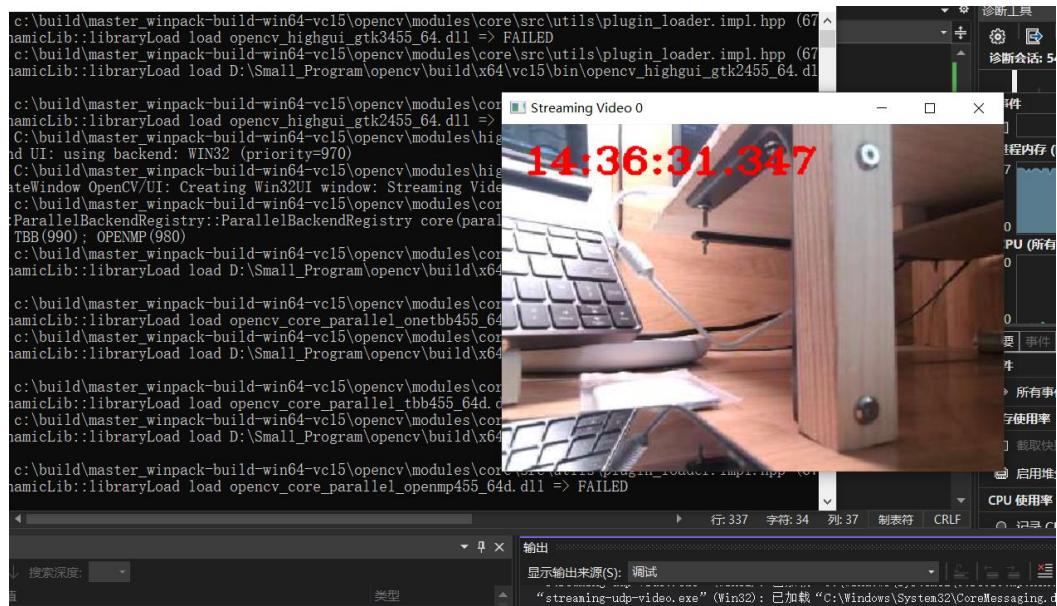


图 43

#### 4.4.4 图像实时超分辨率重建模块

对于机器人前端返回的实时视频，为了减少计算量和占用资源，在传输过程中进行了压缩，而在终端接收之后，通过深度学习方法将视频进行超分辨率重建，从而给用户提供一个清晰的视频。

通过对现存文献的阅读和研究，生成对抗网络模型能够通过使用深层的神经网络从图像中学习复杂的特征分布，同时考虑到深度学习在大量样本任务上的优越表现，文章使用深度学习中对抗生成网络进行视频重建任务的完成。

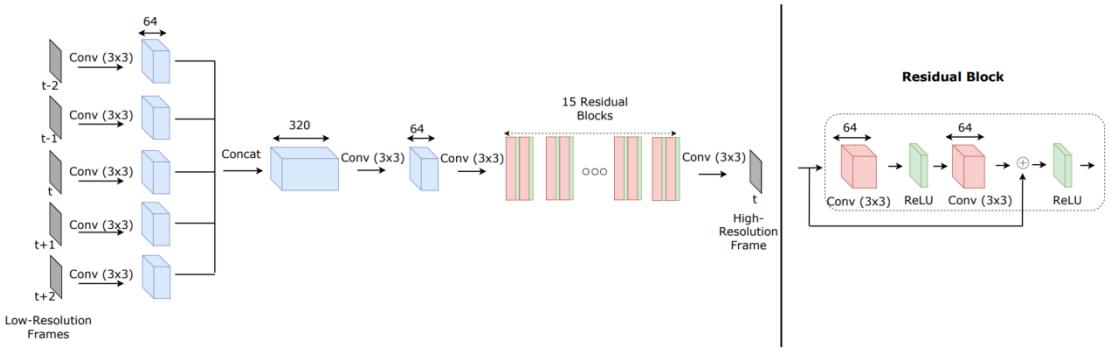
生成对抗网络是一种深度学习网络模型，包括生成器和判别器。生成器是需要学习的模型，通过生成器能够生成一个仿真数据。判断器要判断一个样本是真实样本还是仿真样本，如果判断器判断通过则表示生成样本符合要求，否则需要生成器重新改进学习生成新的样本。通过生成器与判断器的交替训练，最终目标为获得一个性能良好高准确率的完成任务的生成器。

对于本文需要构建的 GAN 网络进行目标分析，需要从单张低分辨率图像生成对应高分辨率图像，即生成器的输入和输出都是单张图像，判别器的输入包括了生成器生成的图像和样本对应的真实图像，输出是一个类别，用于判断生成图像是否为样本对应高分辨率图像。

### **生成器 VSRResnet 网络结构**

对于生成器的网络结构，采用 Lucas 等人提出的 VSRResnet，并在 GAN 训练过程中使用一个全新的判别器来对该生成器进行合适的引导【引用那篇论文】。使用两个正则化器（特征空间和像素空间中的距离损失）进一步增强了 VSR GAN 公式，以获得最终的 VSRResFeatGAN 模型。

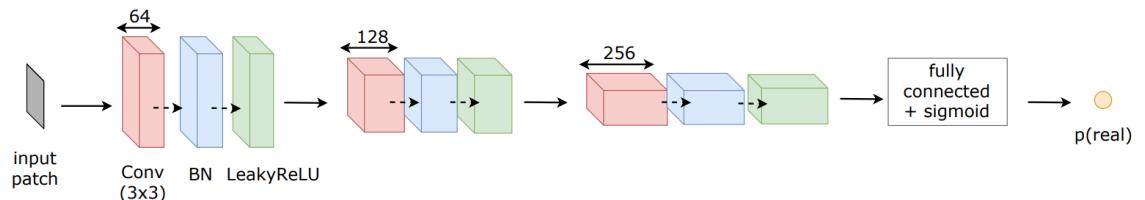
由经验得知，增加模型的深度能够增强网络性能，为视频超分辨率重建提供更好的解决能力。为了增强网络深度同时避免梯度消失问题，VSRResnet 网络结构基于一系列残差块构成，每一个残差块由两个卷积层构成，卷积核大小设置为 3x3，这样能够使用较小的卷积核获取较大面积的感受野，在增强网络容量的同时减少了网络参数的个数。每一层卷积都使用 Relu 作为激活函数。VSRResnet 网络结构如图 44 所示，网络输入为前端返回视频的帧集，对于每一个帧集，使用 64 个大小为 3x3 的卷积核进行特征提取，然后将所有时间段的帧集获取到的特征进行级联获取到视频完整的特征图；然后使用两个卷积操作和 15 个残差块提供最终高分辨率视频的转换；每一个残差块包含两个卷积层，每个卷积层后跟 Relu 激活函数增强模型的非线性拟合能力，对于每个残差块的输入会直接与卷积之后获取的特征相级联，避免由于网络层数加深而带来的梯度消失的问题。

图 44<sup>[9]</sup>

## 判别器结构

传统的 GAN 算法中，生成器的学习是由判别器进行引导完成的，判别器同时被训练以区分生成器生成的图像和训练数据集中的图像。给出一个生成器  $G(z)$ ，例如当  $x$  为训练数据集采样样本时  $D(x)=1$ ，当图像由生成器生成时  $D(G(z))=0$ 。生成器的目标是经过反复训练之后使得判别器能够相信生成的图像为真实图像，即生成器最终收敛到判别器未能识别为“假”的解决方案，就意味着生成器成功学习了训练集图像。

判别器结构如图 45 所示，由三层卷积层和一层全连接层组成，并使用 sigmoid 函数进行二分类，提供该样本是否是真实图像的概率。

图 45<sup>[9]</sup>

判别器输入为训练集中真实高分辨率视频或由生成器生成的高分辨率视频，输出对应着该视频是否为真实视频的概率值。判别器中所有的卷积层都使用大小为 3x3 的卷积核，对于每一个卷积层都使用了 BN 层和 LeakyRelu 作为激活函数。

## 目标函数

$$\min_{\theta} \max_{\phi} L_{GAN}(\phi, \theta) = \mathbb{E}_x[\log D_\phi(x)] + \mathbb{E}_Y[\log(1 - D_\phi(G_\theta(Y)))] \quad (1)$$

生成对抗网络的目标函数如公式 x 所示，其中  $x$  是维度  $N \times N$  的高分辨率

帧， $\mathbf{Y}$  是围绕其低分辨率版本  $\mathbf{y}$  的低分辨率输入帧序列，每个维度  $N \times N$ ， $D$   $\Phi$  是我们的判别器，具有可训练的参数  $\Phi$  和  $G\theta$  是具有可训练参数  $\theta$  的生成器网络，其中这些参数对应于生成器网络和判别器网络的可学习卷积核。

#### 4.4.4 基于 TCP 的 P2P 指令传输发送端

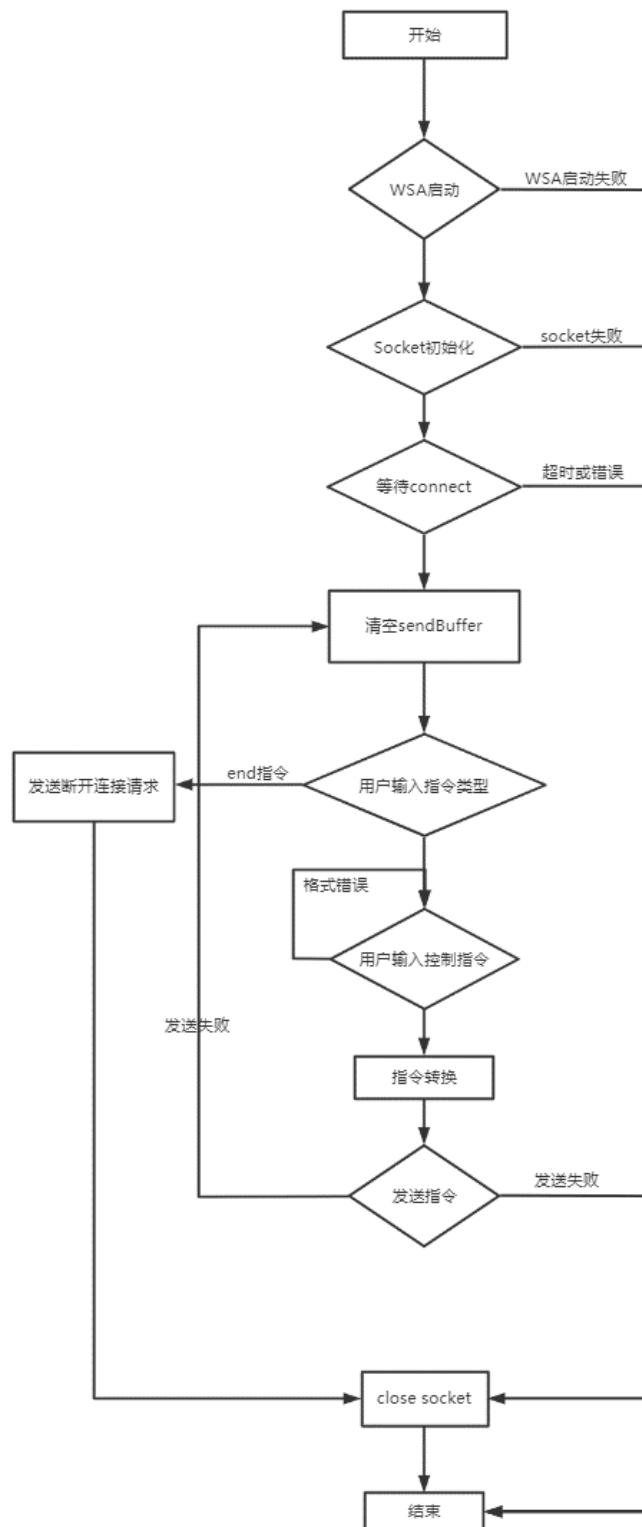


图 46

### 三次握手抓包:

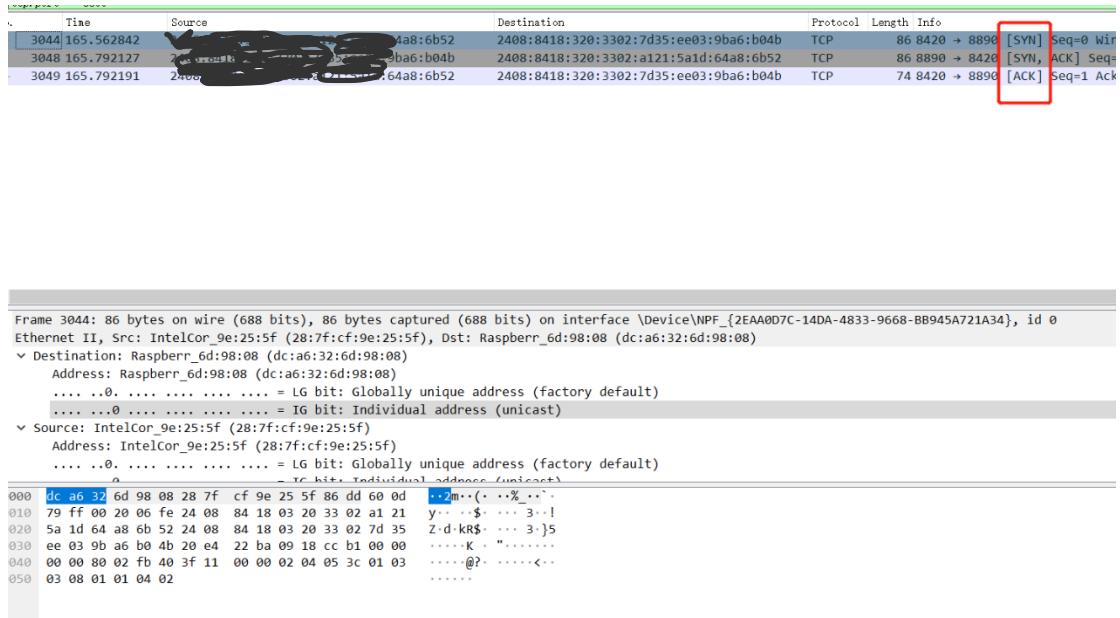


图 47

### 成功建立TCP连接

```

The authenticity of host "192.168.122.193" (192.168.122.193) can't be established.
ECDSA key fingerprint is SHA256:y2YBMSi9UsTme146bG0OYgTVSM0wuk1CS30aTDmaMc.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added "192.168.122.193" (ECDSA) to the list of known hosts.
pi@192.168.122.193's password:
Linux raspberrypi 5.10.103-v7l+ #1529 SMP Tue Mar 8 12:24:00 GMT 2022 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon May 30 14:50:11 2022
pi@raspberrypi: ~
pi@raspberrypi: ~
pi@raspberrypi: ~
pi@raspberrypi: ~ cd codehub/rClient/
pi@raspberrypi: /codehub/rClient $ ls
build include src
pi@raspberrypi: /codehub/rClient $ cd src/
pi@raspberrypi: /codehub/rClient/src $ ls
CMakeLists.txt cmdRecv.cpp cmdService makefile rclient.cpp
pi@raspberrypi: /codehub/rClient/src $ ./cmdService
bindSocketFd = 3
cmd bind ok!
cmd listen ok!
cmd accept success!
Unable to open serial device: No such file or directory
pi@raspberrypi: /codehub/rClient/src $ ./cmdService
bindSocketFd = 3
cmd bind fail: -1
pi@raspberrypi: /codehub/rClient/src $ ./cmdService
bindSocketFd = 3
cmd bind ok!
cmd listen ok!
cmd accept success!
recv 4: 2883

```

The figure shows a terminal session between a Client and a Server. The Client sends a command to the Server, which then processes it and returns a response. The Client's command is: > cmd 90 1 1. The Server's response is: send: 2883. The Client's response is: >

图 48 命令传输

## 总结

本篇论文对远程数字化身系统这一构想的整体样貌做了一个论述，并进行了一个非常简单的实现，但因为时间比较匆忙，而构想庞大，因此略显粗糙，但总算是尽可能详细的将这种技术形态和之下的社会图景描绘了出来；本次通过一个中心化的系统形态来举例构建一个 RDA 系统，但其最理想的形态是一种去中心化的形态，作为元宇宙的一项服务存在，，与多个元宇宙的数字身份关联；而其本身则最终以一套协议的形式存在，即 RDA 协议，任何人只要遵循此协议来实现一个机器人，或一个基站，都可以接入这个网络当中，成为网络介入现实的一条触手，促进系统发展的同时也将从中获利；

这是一种新的技术形态，同时也预示了一种新的人类社会形态的雏形——“单个个体呈分布式存在，并相互融合”。

引用著名科幻小说家、赛博朋克的鼻祖——威廉·吉布森的一句话作为结语——“未来已经到来，只是尚未流行”。

## 参考文献

- [1] Campos C, Elvira R, Rodríguez J J G, et al. Orb-slam3: An accurate open-source library for visual, visual-inertial, and multimap slam[J]. IEEE Transactions on Robotics, 2021, 37(6): 1874-1890.
- [2] Lampert C H. Machine learning for video compression: Macroblock mode decision[C]//18th International Conference on Pattern Recognition (ICPR'06). IEEE, 2006, 1: 936-940.
- [3] Tachi S. Telexistence[M]//Virtual Realities. Springer, Cham, 2015: 229-259.
- [4] Cooper T G, Wallace D T, Chang S, et al. Apparatus for pitch and yaw rotation: U.S. Patent 10,105,128[P]. 2018-10-23.
- [5] 宋爱国. 力觉临场感遥操作机器人 (1): 技术发展与现状[J]. 南京信息工程大学学报: 自然科学版, 2013 (1): 1-19.
- [6] Tachi S, Komoriya K, Sawada K, et al. Telexistence cockpit for humanoid robot control[J]. Advanced Robotics, 2003, 17(3): 199-217.
- [7] Milgram P, Rastogi A, Grodski J J. Telerobotic control using augmented reality[C]//Proceedings 4th IEEE International Workshop on Robot and Human Communication. IEEE, 1995: 21-29.
- [8] Milgram P, Zhai S, Drascic D, et al. Applications of augmented reality for human-robot communication[C]//Proceedings of 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'93). IEEE, 1993, 3: 1467-1472.
- [9] Shi W, Caballero J, Huszár F, et al. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 1874-1883.
- [10] Cao Y, Wang C, Song C, et al. Real-time super-resolution system of 4k-video based on deep learning[C]//2021 IEEE 32nd International Conference on Application-specific Systems, Architectures and Processors (ASAP). IEEE, 2021:

69-76.

- [11]Mystakidis S. Metaverse[J]. Encyclopedia, 2022, 2(1): 486-497.
- [12]Kemp J, Livingstone D. Putting a Second Life “metaverse” skin on learning management systems[C]//Proceedings of the Second Life education workshop at the Second Life community convention. CA, San Francisco: The University of Paisley, 2006, 20.

## 致谢

本论文是在任丙忠老师的悉心指导下完成的，感谢任老师在论文的撰写中给予的细心的指导与帮助。老师那严谨的治学态度、精益求精的科研作风给我们树立了很好的榜样，这将时刻激励我在工作与学习中继续进步。

感谢在系统搭建过程中给予帮助的亲人们朋友们！人类是一个集体，在将来，如此庞大的系统是一个社会性工程，需要各行各业的人们积极参与贡献力量才可以把它从蓝图拉进现实。

我们的征途是星辰大海。