

VMV-Arm 遥操作系统

清华大学第三十二届“挑战杯”学生课外学术科技作品竞赛
作品正式文档

2014 年 3 月

目录

一、项目综述

| | |
|---------------------|---|
| 1.1 研究背景概述 | 2 |
| 1.2 项目难点及解决方案..... | 2 |
| 1.3 项目创新点及竞争优势..... | 3 |

二、设计实现

| | |
|---------------------|---|
| 2.1 系统架构 | 3 |
| 2.2 机械设计 | 4 |
| 2.3 主要功能模块介绍 | |
| 2.3.1 视觉..... | 5 |
| 2.3.2 语音 | 5 |
| 2.3.3 记忆..... | 5 |
| 2.4 程序编写 | |
| 2.4.1 程序流程图..... | 5 |
| 2.4.2 UI 界面介绍 | 6 |
| 2.4.3 维度-角度算法..... | 7 |
| 2.4.4 程序优化..... | 8 |
| 2.5 电机通讯 | 8 |

三、性能测试

| | |
|----------------|----|
| 3.1 随动测试 | 9 |
| 3.2 语音测试 | 10 |

四、项目前景

| | |
|-----------------|----|
| 4.1 实验平台 | 10 |
| 4.2 无线应用 | 10 |
| 4.3 APP 开发..... | 11 |

五、附录

| | |
|----------------|----|
| 5.1 物资明细 | 10 |
| 5.2 项目进程 | 11 |
| 5.3 项目计划 | 12 |
| 5.4 项目评价 | 12 |

| | |
|-----------|----|
| 参考文献..... | 13 |
|-----------|----|

VMV-Arm 遥操作系统

一、项目综述

1.1 研究背景概述

图形用户界面（Graphical User Interface, GUI）在 20 世纪八十年代已经出现，并且成为人机交互的主要方式。但是随着时代的发展，交互方式也在不断转变，传统的 GUI 交互由于需要学习软件开发者预先设计的动作而显得复杂。因此，自然用户界面（Natural user interface, NUI）应运而生。在 NUI 下，使用者可以用最自然的行为（动作、语音等）和机器直接交流，实现使用方式的个性化和人性化。

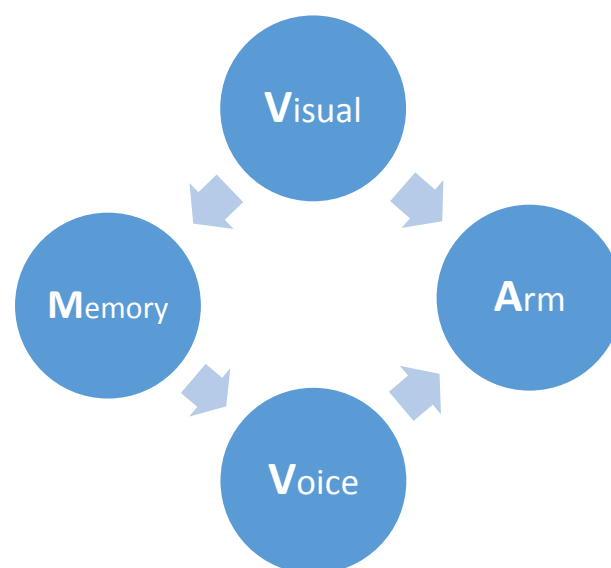


图 1

目前 NUI 主要用于改善控制方式，如微软的 PixelSense 技术，实现更自然地控制计算机，而非机械。我们的作品则希望利用 NUI 系统直接驱动机械，高效地完成人控操作。

遥操作技术可以方便、准确、实时地操纵机器，代替人的行动，有很大的实用价值。但目前遥操作单一依靠力学操纵器控制机器，耦合度高，交互性差。我们的作品基于 NUI 界面，引入遥操作技术，实现了视觉（visual）、语音（voice）、记忆（memory）控制机械臂的人机臂三方交互系统——VMV-Arm。

该系统可以利用视觉捕捉设备（如 Kinect）捕捉人的影像信息，直接控制机械臂同步伴随移动，也可以将捕捉到的信息记忆下来，导入数据库。记忆的内容在数据库中模块化，可根据用户需要被手动调用或语音调用。UI 界面的手动指令也可以直接控制机械臂的定点定向运动。由于机械臂和输入方式为耦合度很低的两个部分，其他输入控制方式还可以根据需求继续开发。

1.2 项目难点及解决方案

本项目完成了一套具有视觉、语音和记忆功能的 NUI 遥操作系统，其功能设计、机械制造、软件开发、硬件实现等均为学生原创。

团队成员在研发过程中对机械臂的传动与控制做了大量的调研，学习了现有的 Kinect 视觉捕捉技术、语音识别技术和舵机传动实现方式，分析了其中的原理和不足，改进形成自己的原创设计。

以 Kinect 为代表的影像视觉捕捉软件，由于扰动存在，往往有很明显的抖动与延迟。我们通过对捕捉信号数字化处理，并采用数值方法有效平滑动作过程，减少了机械臂的抖动。

在语音识别基础功能之上，我们进行了文本关键字筛选与**模糊化处理**，保证了多条相同语意的语音指令能操纵机械臂完成相同动作。

传统的水平多关节机械臂只具备 3 个自由度，不能很好地实现类人化运动，因此我们以“手臂的外形、行为是否能够达到与人手臂基本相近”为初衷，设计加工了 **4 个维度的机械臂**。

这个项目是**多学科交叉的项目**，涉及结构设计、机械加工、算法设计、编程、无线通信、舵机操控等不同学科领域。在研发过程中，团队成员与老师深入讨论，学习了许多学科与实际加工的知识，这对所有成员都是一个巨大的挑战和宝贵的实践经验。

1.3 项目创新点及竞争优势

VMV-ARM 遥操作系统基于 NUI 界面，将视觉、语音、记忆多种控制方式融合为一体。

目前，基于计算机视觉的体感设备已经成熟，如 Kinect 的人体骨骼追踪功能。此类设备使用简便，价格亲民，基于此开发的系统很**容易推广**。此外，语音控制系统正处于高速发展期，被广泛运用在智能手机和智能家电领域，如苹果的 siri 系统。结合这两大技术，我们实现了模块化的记忆数据库，开发出的人机臂三方交互的遥操作系统**具备可行性**。

这样的 NUI 系统有几大优势：

1. 信号捕捉、机械控制和信息传递处理三部分的模块化程度高，具体交互方式可以根据需求专一性开发，发展空间大；
2. Kinect 等体感外设销量庞大，麦克风操作简易，系统的可推广性好；
3. 系统记忆数据库可以根据操作者具体需求实现个性化定义。

二、设计实现

2.1 系统架构

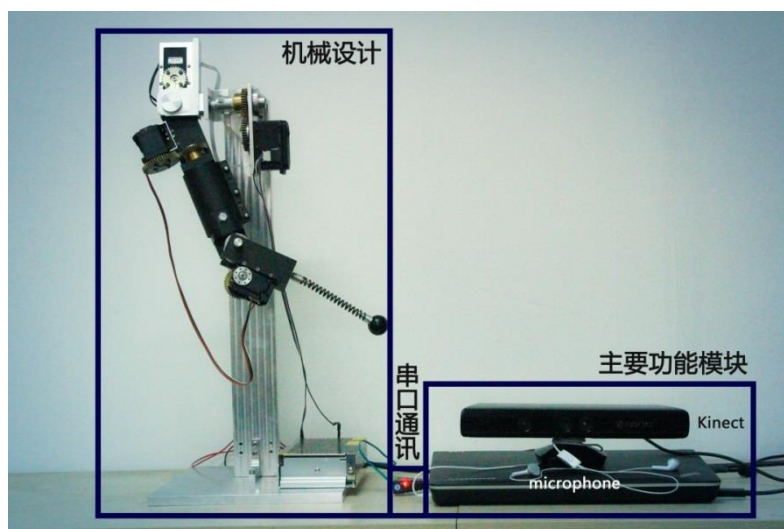


图 2

2.2 机械设计

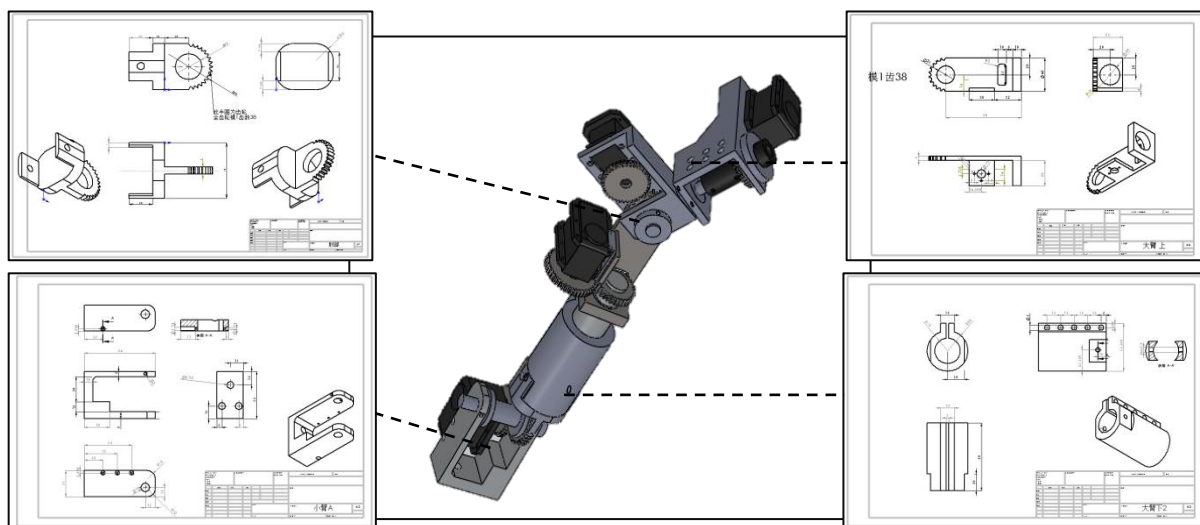


图3 关键节点设计图纸

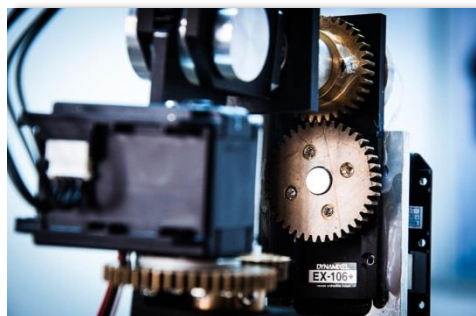


图4 大臂旋转运动关节

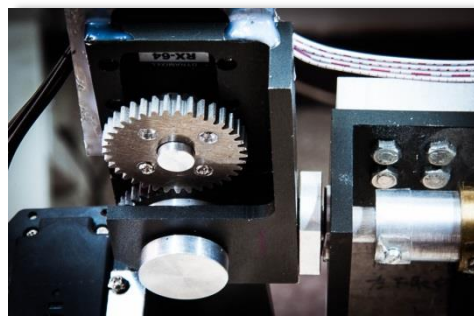


图5 大臂前后运动关节

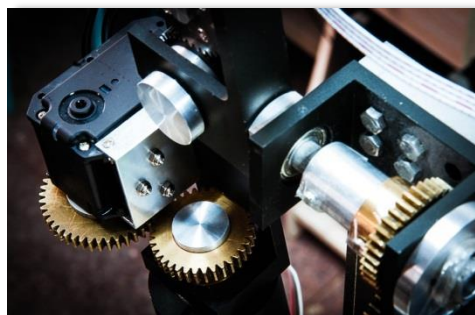


图6 大臂左右运动关节



图7 肘伸缩运动关节

机械设计的目标是**类人化**。机械结构框架、关节活动情况、电机位置等因素，都会直接关系到“手臂的外形、行为是否能够达到与人手臂基本相近”这个设计初衷。从外形到结构，该机械臂都是由本组**自行设计**完成。

根据人手臂的运动方式，机械臂类似地具有四个自由度，分别由四个电机驱动。这四个电机依次控制机械臂的大臂前后运动、大臂左右运动、大臂旋转运动、肘伸缩运动。当四个电机配合起来转动时，就能够实现高度类人化的动作。

2.3 主要功能模块介绍

2.3.1 视觉

Kinect 是一种 3D 体感摄影机，微软在 2011 年 6 月推出了 Kinect for Windows SDK，借助其 Skeleton Tracking（骨骼追踪）的核心技术，可即时动态捕捉人体关节点的坐标。

Kinect 最多可以追踪 20 个骨骼点(如图 7 所示)。实现捕捉骨骼点之后，使用 LINQ 来获取 TrackingState 等于 Tracked 的 SkeletonData 数据。在 SkeletonData 对象的 Joints 属性集合中保存了所有骨骼点的信息。每个骨骼点的信息都是一个 Joint 对象，其中的 Position 的 X、Y、Z 表示了三维位置。其中 X 和 Y 的范围都是-1 到 1，而 Z 是 Kinect 到识别物体的距离。通过这项功能，获取了任意时刻右手、右肘、右键的相对坐标。

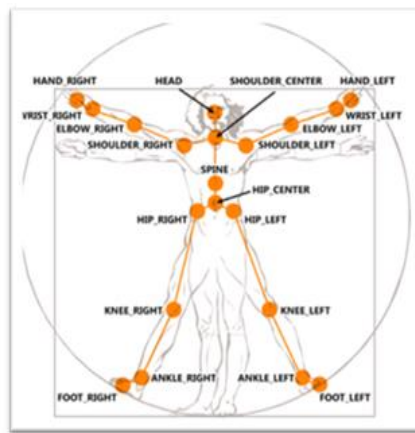


图 8-1

2.3.2 语音

此模块借助 Microsoft Speech SDK 实现。无论说话者是男女老少，声音内容都可以被有效识别。并运用到了**模糊处理**，在后台数据库中补充了多组测试数据，相同语意下的不同说法亦可被成功识别。



图 8-2

2.3.3 记忆

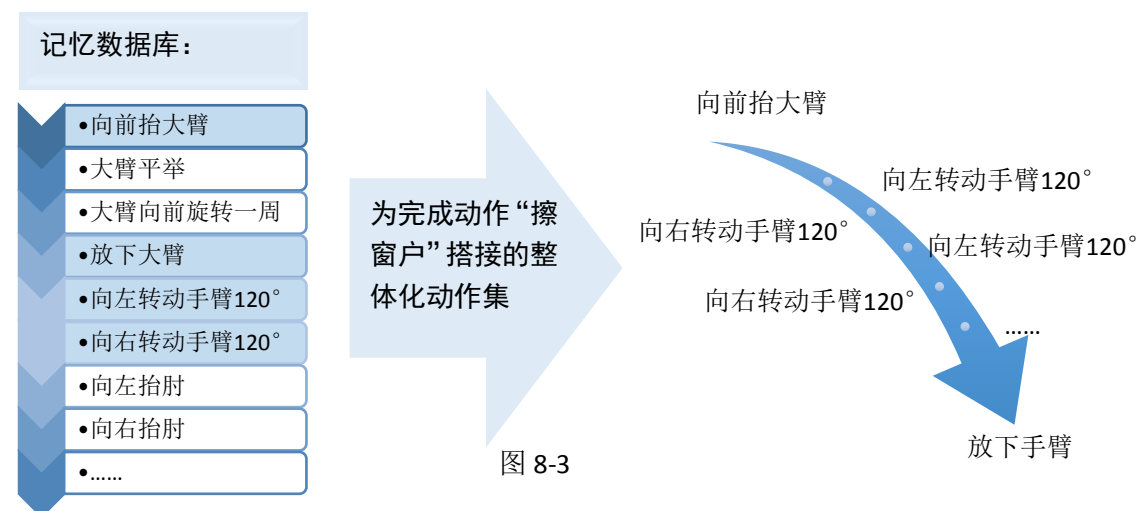


图 8-3

在用户与 VMV-Arm 遥操作系统交互过程中，输入的视觉、语音等信号，都被系统自觉标记记录，扩充形成记忆数据库，我们对数据库中的记忆模块化，组织形成机械臂基本动作集合与视觉、语音控制信号的映射。有特殊需求的用户可以调用记忆数据库模块，自行搭接

成整体化动作集，操纵机械臂完成更为复杂的多级运动。

2.4 程序编写

2.4.1 程序流程图

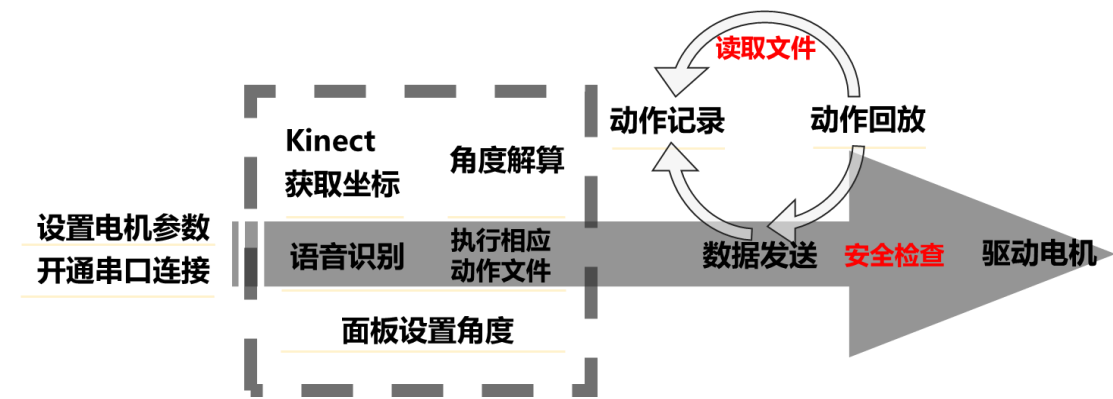


图 9

2.4.2 UI 界面介绍

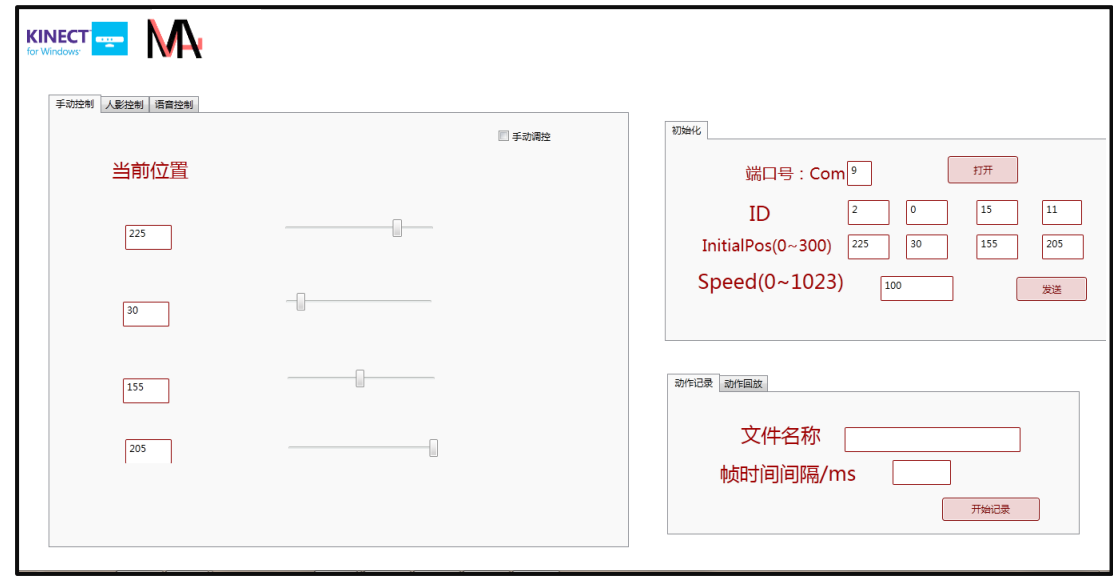


图 10

(1) 初始化模块

打开串口：输入相应的串口号，点击“打开”则打开串口，不打开串口无法进行下面的操作；

设置初始位置：输入对应电机的 ID、初始位置和限制速度，点击发送便可以将机械臂控制到初始位置。

(2) 控制模块

手动控制：选中右上角“手动控制”，拉动滚动条即可控制机械臂运动；

人影控制：选中右上角“Kinect 控制”，在 Kinect 前人体运动可控制机械臂运动，并且在面板中会显示现在人体的关节图像；

语音控制：点击按钮开始语音控制，再次点击控制结束。

(3) 记忆模块

动作记录：输入储存的动作名称，点击相应按钮开始记录，再次点击结束记录；

动作回放：选择相应的文件，点击回放按钮便可进行动作回放。

2.4.3 维度-角度转换算法

在前期的过程中，人手臂关节的坐标获取以及滤波这些准备工作都已经完成。现在的主要问题在于如何把得到的关节坐标转换为电机的转角。本组同学自行学习设计得出该算法，作为该控制程序的核心部分很好的解决了这个问题。

首先，建立空间直角坐标系。将人手臂的前、下、左方向分别设为坐标轴的 X、Y、Z 轴，由于 Kinect 自身的坐标轴（如图 1-3）与实际运算所设的不同，因此在计算之前首先需要坐标转换。为了运算准确便捷，规定人体在 Kinect 前运动时，是面对着 Kinect，因此根据 Kinect 的建系标准（如图 1-3，z 轴表示 Kinect 正对方向），有以下坐标变换：

$$X = -Z_{\text{Kinect}}$$

$$Y = -Y_{\text{Kinect}}$$

$$Z = X_{\text{Kinect}}$$

经过坐标相减就得到肩->轴、轴->手两个方向的单位向量 $v_{se}(x_1, y_1, z_1), v_{ew}(x_2, y_2, z_2)$ ，由于向量的平移不变性，可认为肩部在原点。

设 θ_1 为大臂与 Y 方向的夹角（ $-0.5\pi < \theta_1 < 0.5\pi$ 且在 YZ 平面前方为正）， θ_2 为大臂与 XY 平面的夹角（ $-0.5\pi < \theta_2 < 0.5\pi$ 且在 XY 平面左侧为正）， θ_3 为小臂以大臂方向为轴的旋转角度（ $-0.5\pi < \theta_3 < 0.5\pi$ 且沿大臂向下看顺时针方向为正）， θ_4 为大臂与小臂的夹角（ $0 < \theta_4 < 0.75\pi$ 且屈肘方向为正）。经过几何分析，可以得到：

$$\theta_1 = \arcsin\left(\frac{x_1}{\sqrt{x_1^2 + y_1^2}}\right), x_1 > 0 \text{ 或 } \theta_1 = \pi - \arcsin\left(\frac{x_1}{\sqrt{x_1^2 + y_1^2}}\right), x_1 < 0$$

$$\theta_2 = \arcsin(z_1)$$

$$\theta_4 = \arccos(x_1 * x_2 + y_1 * y_2 + z_1 * z_2)$$

θ_3 的计算则考虑到坐标变换：将 v_{ew} 坐标变换到以大臂方向为 X 轴的坐标系内，该坐标系坐标与原坐标系坐标的有以下关系：

$$x' = x * \cos(\theta_1) \cos(\theta_2) + y * \sin(\theta_1) \sin(\theta_2) + z * \sin(\theta_2)$$

$$y' = -x * \cos(\theta_1) \sin(\theta_2) - y * \sin(\theta_1) \sin(\theta_2) + z * \cos(\theta_2)$$

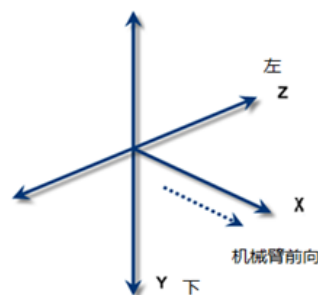


图 11

$$z' = x * \sin(\theta_1) - y * \cos(\theta_1)$$

那么，将原肘手向量 $v_{ew}(x_2, y_2, z_2)$ 代入以上得到变换后的向量 $new_v_{ew}(x_3, y_3, z_3)$ ，进而就可以得到：

$$\theta_3 = \arcsin\left(\frac{y_3}{\sqrt{x_3^2 + y_3^2}}\right)$$

这样，四个电机相对与对应基准位置的所转动角度都能得到，再与初始时各电机角度换算，即可得到现在时刻的电机绝对角度。

2.4.4 程序优化

在主程序中，增设了以下模块，对系统功能进行优化：

(1) 滤波环节

验证每一帧电机转速，若其超过了设置阈值，则将转速设置为阈值，以此防止电机抖动。

(2) 安全检查

设置电机转角变化范围，并在数据发送前进行检查，避免转角越界对操作人员或机械结构造成损害。

2.5 电机通讯

该过程主要实现电脑与电机的通讯，由 PC 机向电机发送转角数据和转动指令，驱动电机转动相应的角度。Dynamixel 大扭矩舵机采用 RS-485 的串口通讯协议。



图 12-1 USB 转 RS-485



图 12-2 RS-485 蓝牙适配器

舵机与控制程序通过发送、接收指令包（packet）的形式来通讯。程序发送给舵机的指令包称为 Instruction Packet，据此可以控制舵机。其格式如下：

Instruction Packet

| HDRL | HDRH | ID | LEN | INST | Addr | Data[n] | Checksum |
|------|------|----|-----|------|------|--------------------|----------|
| FF | FF | X | X | X | X | m_cInstMem[Addr+n] | ? |

开始是标志位的高位、低位，它们标志着一指令的开始，紧接着是舵机的 ID。机械臂上所有舵机的信号线都是互相短接的，指令要通过每个舵机独一无二的 ID 来区分它们。

接下来是指令的类型号，每个号码对应一种指令，常用的指令包括读、写、同步读等。之后是可选的指令参数。最后是为了保证命令发送正确所设置的检验位 **Checksum**。舵机每收到一条指令都会进行 **Checksum** 的检查，如果不正确则说明指令有误，不会执行。

在初始阶段，通过串口调试助手对舵机进行 ID 编址，波特率（115200）、校验位（无）数据位（8）、停止位（1）等参数设置。舵机内部有两块存储器：一块 **EEPROM**，和一块 **RAM**。**RAM** 上的内容断电即失，存储了舵机上的动态数据，如当前转角、指定转速等。通过指令修改 **RAM** 的这些数据，实现了对舵机的控制。

三、性能测试

3.1 随动测试

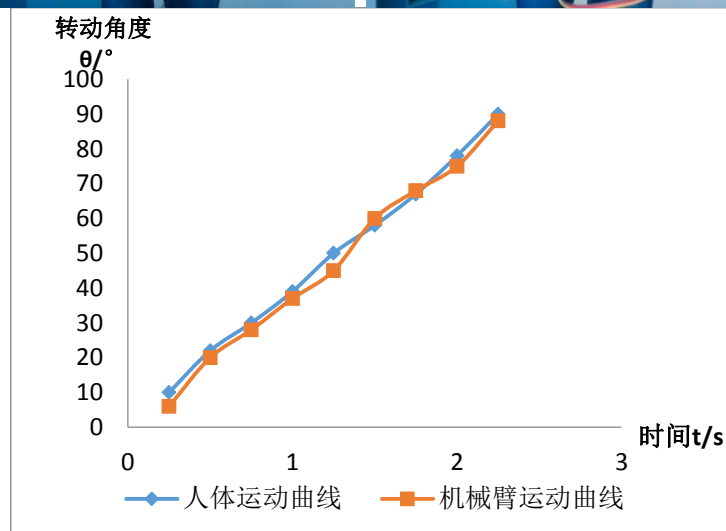
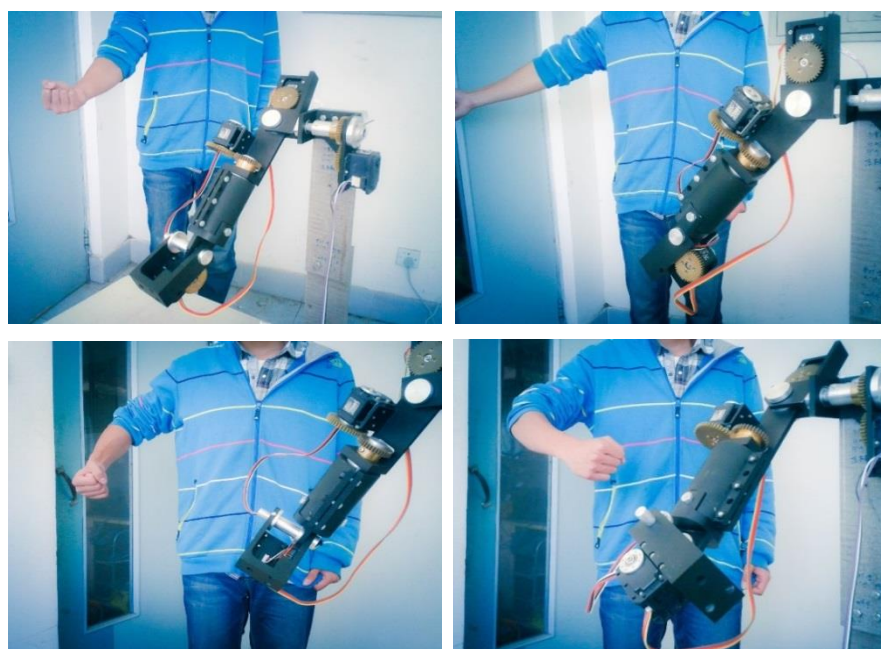


图 13

VMV-Arm 遥操作系统视觉捕捉用户的手臂运动，机械臂能够准确、即时完成类人运动。用户测试向前抬大臂动作，瞬时监测人体运动坐标与机械臂位置坐标，得到拟合曲线，二者有很高的线性相关度，即机械臂能稳定实现随动。

3.2 语音测试

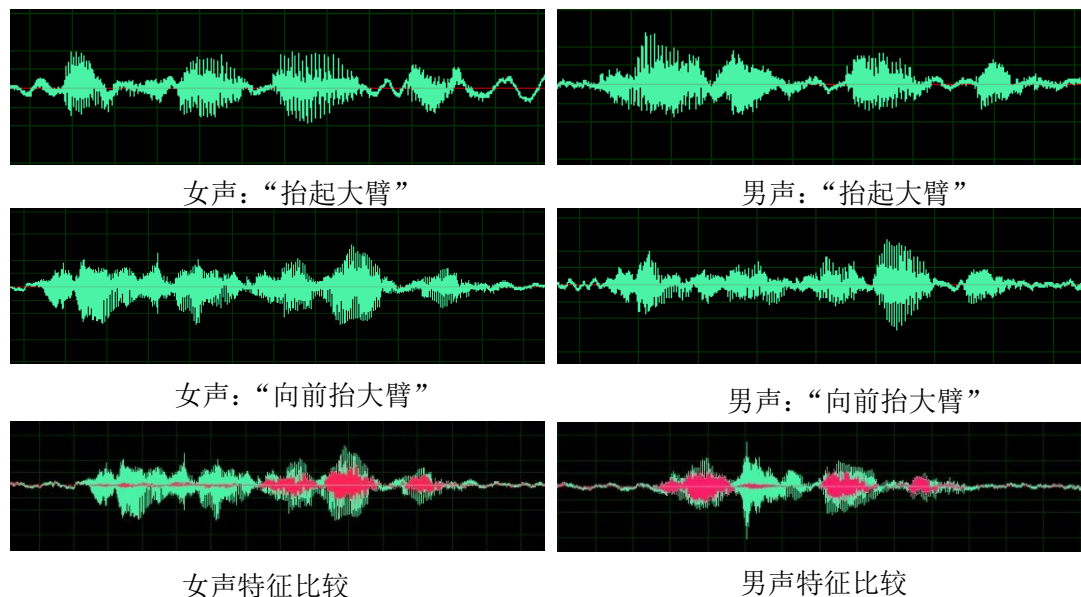


图 14

VMV-Arm 遥操作系统语音识别不同用户（男声、女声）的指令，触发机械臂运动。上图所示为用户“向前抬大臂”和“抬起大臂”指令声波信号，系统语音识别模块可有效滤除噪声，通过筛选识别关键字“抬大臂”，并模糊处理无用文字，两条同语意的不同指令均操纵机械臂实现抬起动作。

四、项目前景

4.1 实验平台，作为多学科交叉的教学模版

本项目具有**多学科交叉**的特点，涉及结构设计、机械加工、算法设计、编程、无线通信、舵机操控等不同学科领域。项目的开发过程是对自动化、电子、机械、精密仪器等专业大学生理论结合实践的有效训练。清华大学基础工业训练中心每年都开展项目式教学，此项目具有很大的案例参考价值，一方面可以分模块作为金工实习机械设计加工的模版，另一方面集成化、系统化和程序化的设计流程可作为学生项目开发的蓝本。

4.2 无线应用，大大提高操作的便捷性与用户体验

免穿戴式的遥操作技术摆脱了穿戴设备的束缚，使操纵机械臂的类人化运动更加简便。

我们进一步的开发将从有线串口发展到无线传送，**扩大机械臂的活动范围与延展性能**。实际应用中，远距离大范围的便捷操作会大大提高用户体验，尤其对于残疾人群有很大帮助。

4.3 手机 APP 开发，可商业化的智能移动终端应用

App 提供了比以往的媒介更丰富多彩的表现形式。移动设备的触摸屏、摄像头、语音孔等都可作为 VMV-Arm 遥操作系统的**多方式前端接口**，同时互联网上线 App 有利于项目的实现商业化应用。

五、 附录

5.1 物资明细

| 物资 | 数量 | 备注 |
|----------------|----|--------|
| EX-106 舵机 | 1 | |
| RX-64 舵机 | 4 | 两个备用 |
| RX-28 舵机 | 4 | 两个备用 |
| 开关电源 | 1 | |
| USB 转 RS485 接口 | 1 | 用于有线通讯 |
| 蓝牙 RS485 适配器 | 1 | 用于无线通讯 |
| Kinect | 1 | |
| 螺丝/螺母 | 20 | |
| 铝料 | | |

5.2 项目进程

2012 年 10 月，组建团队，同指导老师确定项目方案。

2012 年 11 月，完成第一版机械设计图。

2013 年 2 月，实现 Kinect 获取人体关节坐标。

2013 年 3 月，完成第一版机械加工。

2013 年 8 月上旬，完成电机串口调试与驱动。

2013 年 8 月中旬，完成主程序编写。

2013 年 9 月，完成项目第一阶段任务，实现机械臂随人体运动的基本功能。

2014 年 1 月，同清华大学基础工业训练中心建立合作关系。

2014 年 2 月中旬，更换新舵机，完成第二版机械设计图。

2014 年 2 月下旬，完成第二版机械加工，开发记忆功能。

2014 年 3 月，组建新团队，完成项目第二阶段计划，开发无线操控功能和语音功能。

5.3 项目计划

2014 年 3 月下旬，对已实现的视觉捕捉、语音识别功能优化，提高信号控制精度。

2014 年 4 月，编写记忆模块的数据库智能算法，训练样本数据，实现常用人动作集合。

2014 年 5 月，开发移动终端的 App，实现蓝牙传输的遥操作功能。

5.4 项目评价

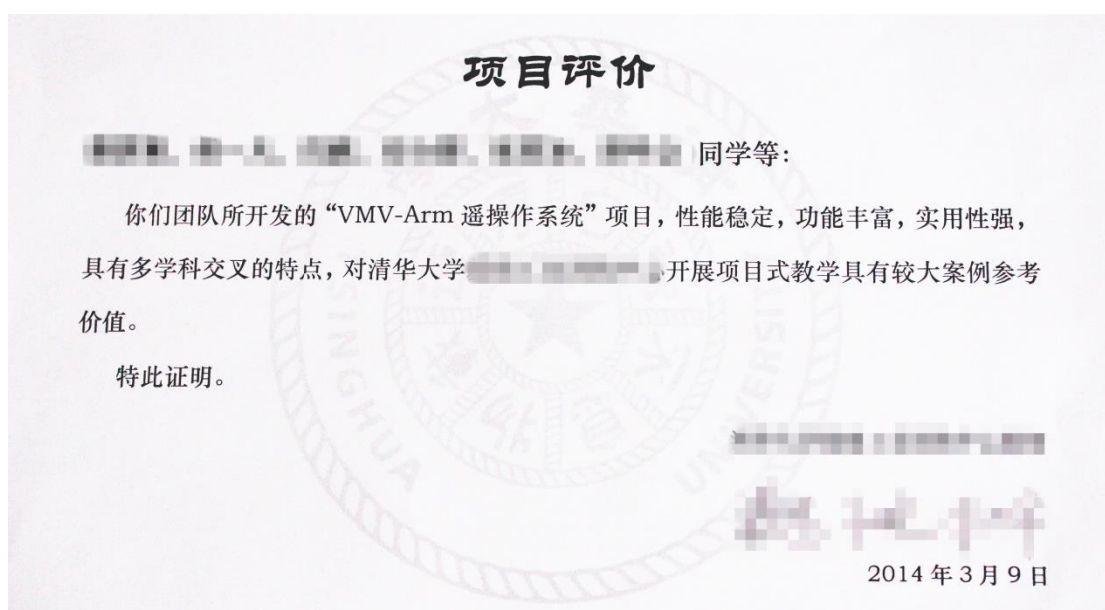


图 15

参考文献

- [1]. Ganganath N, Leung H. Mobile robot localization using odometry and kinect sensor[C]//Emerging Signal Processing Applications (ESPA), 2012 IEEE International Conference on. IEEE, 2012: 91-94.
- [2]. Rakprayoon P, Ruchanurucks M, Coundoul A. Kinect-based obstacle detection for manipulator[C]//System Integration (SII), 2011 IEEE/SICE International Symposium on. IEEE, 2011: 68-73.
- [3]. Pang Y Y, Ismail N A, Gilbert P L S. A real time vision-based hand gesture interaction[C]//Mathematical/Analytical Modelling and Computer Simulation (AMS), 2010 Fourth Asia International Conference on. IEEE, 2010: 237-242.
- [4]. Jiang Q, Kumar V. The inverse kinematics of 3-D towing[M]//Advances in Robot Kinematics: Motion in Man and Machine. Springer Netherlands, 2010: 321-328.
- [5]. 韩峥, 刘华平, 黄文炳, 等. 基于 Kinect 的机械臂目标抓取[J]. 智能系统学报, 2013, 2: 014.
- [6]. Gonzalez F, Dasgupta D, Kozma R. Combining negative selection and classification techniques for anomaly detection[C]//Evolutionary Computation, 2002. CEC'02. Proceedings of the 2002 Congress on. IEEE, 2002, 1: 705-710.
- [7]. Adams R J, Hannaford B. Stable haptic interaction with virtual environments[J]. Robotics and Automation, IEEE Transactions on, 1999, 15(3): 465-474.
- [8]. 邓乐, 赵丁选, 唐新星. 具有力觉临场感的遥操作操纵器关键技术研究[J]. 工程设计学报, 2005, 12(3): 160-161.
- [9]. Bartneck C, Forlizzi J. A design-centred framework for social human-robot interaction[C]//Proceedings of Ro-Man. 2004: 591-594.
- [10]. 蔡自兴. 机器人学[M]. 北京: 清华大学出版社, 2000.