

分类号：TP242

密 级：公开

学校代码：10363

学 号：2220110110



安徽工程大学

Anhui Polytechnic University

硕士学位论文

题目 基于虚拟现实的打磨机器人交互
技术研究

论 文 作 者 王文豪

校 内 导 师 张振（副教授）

校 外 导 师 吴松（工程师）

专业学位类别 机械

研究方向（领域） 智能机器人

论文提交日期： 2025 年 5 月 20 日

分类号：TP242

学校代码：10363

密级：公开

学号：2220110110

基于虚拟现实的打磨机器人交互技术研究

Research on Interaction Technology of Grinding Robots Based on Virtual Reality

学生姓名	王文豪
校内导师	张振（副教授）
校外导师	吴松（工程师）
专 业	机械
研究方向	智能机器人

论文答辩日期：2025 年 05 月 10 日

安徽工程大学学位论文原创性声明

本人郑重声明：我恪守学术道德，崇尚严谨学风。所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已明确注明和引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品及成果的内容。论文为本人亲自撰写，我对所写的内容负责，并完全意识到本声明的法律后果由本人承担。

学位论文作者签名：王文豪

日期：2025 年 5 月 14 日

安徽工程大学学位论文版权使用授权书

学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅或借阅。本人授权安徽工程大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本学位论文属于

保密 ☐，在 _____ 年解密后适用本版权书。

不保密 ☒。

学位论文作者签名：王义豪

指导教师签名：张振

日期：2025 年 5 月 14 日

日期：2025 年 5 月 14 日

基于虚拟现实的打磨机器人交互技术研究

摘 要

《中国制造 2025》强调智能制造和数字化技术在制造业转型升级中的重要性，旨在提高生产效率、产品质量，同时减少资源消耗和安全风险。在制造业的关键工艺环节中，打磨作业作为提升产品外观质量和功能性能的重要工序，在航空航天、电子产品等高精尖领域的地位日益凸显。然而，人工打磨过程中存在显著的不确定性和操作复杂性，打磨机器人在替代人工操作时，也面临交互效率低、适应性差等瓶颈问题。因此，利用虚拟现实及同步控制技术，能为打磨工艺学习提供新思路。本文针对打磨机器人系统中异构模型差异导致的控制精度不足、打磨工艺过程中存在的不确定性与操作复杂性，以及打磨作业缺乏高保真的视觉-力觉融合反馈等关键技术问题展开研究。

首先，针对操作端和执行端结构上的差异性，采用改进的 D-H 参数法分别建立操作端与执行端的连杆坐标系，并基于该坐标系对二者进行正运动学求解。为进一步实现执行端机器人的精确运动控制，采用 Pieper 解法对执行端进行了逆运动学求解。在此基础上，利用蒙特卡洛法对操作端与执行端的工作空间进行求解与可视化，并分析二者工作空间的差异性，设计了操作端与执行端的空间映射算法，为后续的操作端、虚拟执行端、实体执行端之间的控制奠定了理论基础。

其次，为解决打磨环境的不确定性和操作复杂性，本文设计了一套虚拟现实打磨机器人交互系统。该系统首先构建了交互系统的总体框架，明确了功能设计，并完成了虚拟场景建模与 UI 控制界面开发。同时，通过动态建立工件网格模型和球形打磨头变形算法，实现了实时去除效果渲染。此外，研究还深入探讨了碰撞检测、虚拟现实交互技术（如力觉交互和 VR 交互）、坐标系转换以及虚实控制，进一步优化了虚实协同操作能力，有效的提升了打磨机器人的交互效率与操作精度。

最后，通过虚拟现实打磨机器人交互系统的测试实验，构建了系统性实验方案，采用多维度测试方法验证了该系统的有效性。实验结果表明，虚拟位置映射实验误差低于 $\pm 0.03\%$ ；实体机器人位置映射误差控制在 $\pm 1.50\%$ 以内，验证虚实位置映射可靠性；通过力觉交互实验，对比分析了虚实打磨过程中的力反馈数据，结果表明二者误差范围基本吻合，充分证明了虚拟现实打磨机器人系统能够满足真实的力觉交互需求。上述实验结果从位置精度、映射能力和力觉交互等多个维度验证了系统的有效性和实用性。

关键词：虚拟现实，打磨机器人，空间映射，位置映射，力觉交互

RESEARCH ON INTERACTION TECHNOLOGY OF GRINDING ROBOTS BASED ON VIRTUAL REALITY

ABSTRACT

“Made in China 2025” emphasizes the critical role of intelligent manufacturing and digital technologies in the transformation and upgrading of the manufacturing industry, aiming to improve production efficiency and product quality while reducing resource consumption and safety risks. In key manufacturing processes, as a crucial procedure for improving product surface quality and functional performance, grinding operations have gained increasing prominence in high-precision and advanced sectors such as aerospace and electronic products. However, manual grinding processes have significant uncertainties and operational complexities. When replacing manual operations, grinding robots face bottleneck challenges including low interaction efficiency and poor adaptability. Consequently, the integration of virtual reality and synchronous control technologies offers novel approaches for grinding process learning. This study addresses key technical challenges in robotic grinding systems, including insufficient control accuracy caused by heterogeneous model discrepancies, uncertainties and operational

complexities inherent in grinding processes, the lack of high-fidelity visual-force feedback integration during grinding operations.

Firstly, to address the structural discrepancies between the operator-side and executor-side systems, an improved D-H parameter method was adopted to establish link coordinate systems for both the operator and executor ends. Forward kinematic solutions for both systems were derived based on these coordinate systems. To further achieve precise motion control of the executor-side robot, inverse kinematic solutions were obtained using Pieper's method. On this basis, the Monte Carlo method was used to solve and visualize the workspaces of both operator and executor ends, with subsequent analysis of workspace discrepancies. A spatial mapping algorithm between the operator and executor systems was designed, establishing a theoretical framework for subsequent control integration among the operator interface, virtual executor, and physical executor. This approach lays an foundation for coordinated control across heterogeneous systems in robotic grinding applications.

Secondly, to address the uncertainties and operational complexities inherent in grinding environments, in this study, a virtual reality-based interactive robotic grinding system was proposed. The system framework was first constructed, incorporating functional design specifications, virtual scene modeling, and UI control interface development. Real-time material removal rendering was achieved through the dynamic generation

of workpiece mesh models and a spherical grinding head deformation algorithm. Furthermore, the study collision detection, virtual reality interaction techniques (including force feedback interaction and VR interaction), coordinate system transformations, and virtual-physical control mechanisms were systematically investigated. These advancements optimize virtual-physical collaborative operation capabilities, effectively improving the interaction efficiency and operational accuracy of the robotic grinding system. The developed methodology demonstrates significant potential for addressing adaptability challenges in complex manufacturing scenarios.

Finally, through experimental testing of the virtual-reality robotic grinding interaction system, a systematic experimental protocol was developed to validate the system's effectiveness using multi-dimensional evaluation methods. Experimental results show that the virtual position mapping experiment error is maintained below $\pm 0.03\%$, while the physical robot position mapping error is controlled within $\pm 1.50\%$, thereby verifying the reliability of the virtual-physical position mapping. Through haptic interaction experiments, force feedback data between virtual and physical grinding processes was compared, revealing consistent error ranges that fully validated the system's capacity to meet authentic haptic interaction requirements. These experimental outcomes-spanning positional accuracy, mapping capability, and force interaction-collectively

validated the system's effectiveness and practicality. The comprehensive testing framework provides empirical evidence for implementing virtual-physical collaborative grinding systems in industrial applications.

KEY WORDS: Virtual reality, Grinding robot, Spatial mapping, Position mapping, Haptic interaction

目 录

摘 要	I
ABSTRACT.....	III
第 1 章 绪论	1
1.1 研究背景及意义	1
1.2 国内外研究现状	2
1.2.1 打磨机器人研究现状.....	2
1.2.2 虚拟现实仿真技术研究现状.....	4
1.2.3 遥操作机器人国内外研究现状.....	6
1.3 主要研究内容	9
第 2 章 操作端与执行端的运动学建模与空间映射	11
2.1 操作端与执行端选型	11
2.2 操作端的运动学分析	13
2.2.1 连杆坐标系建立.....	13
2.2.2 正运动学计算.....	14
2.2.3 正运动学仿真.....	15
2.3 执行端的运动学分析	16
2.3.1 连杆坐标系建立.....	16
2.3.2 运动学正解.....	17
2.3.3 运动学逆解.....	18
2.4 操作端与执行端的空间映射	21
2.4.1 基于笛卡尔空间的映射.....	21
2.4.2 工作空间差异性分析.....	21
2.4.3 空间映射的实现.....	23
2.5 本章小结	25
第 3 章 基于虚拟环境的打磨机器人交互系统设计	26
3.1 交互系统总体方案	26
3.1.1 交互系统总体框架.....	26
3.1.2 交互系统的功能设计	27
3.1.3 交互平台的选型.....	28
3.2 虚拟场景建模	29
3.3 UI 控制界面设计	32
3.4 实时去除效果渲染	34
3.4.1 工件网格模型的建立.....	34

3.4.2 球形打磨头变形算法	36
3.5 本章小结	38
第 4 章 虚实交互与控制技术研究	39
4.1 碰撞检测	39
4.1.1 层次包围盒算法	39
4.1.2 碰撞体与刚体	41
4.2 虚拟现实交互技术	42
4.2.1 力觉交互	42
4.2.2 VR 交互	43
4.3 坐标系转换	45
4.4 打磨机器人运动学算法验证	47
4.5 虚实控制研究	48
4.6 本章小结	49
第 5 章 虚拟现实打磨机器人交互系统测试实验	51
5.1 实验平台搭建	51
5.2 位置映射实验	52
5.2.1 虚拟位置映射实验	52
5.2.2 实体机器人位置映射实验	54
5.3 力觉交互实验	56
5.4 虚实打磨实验	59
5.5 本章小结	60
第 6 章 总结与展望	61
6.1 总结	61
6.2 展望	62
参考文献	63
攻读硕士期间发表的学术成果	68
致谢	69

第1章 绪论

1.1 研究背景及意义

在“十四五”规划的推动下，智能制造成为制造业转型升级的核心方向^[1]。然而，在智能制造的浪潮中，一些高技能、高风险的工艺操作对非技术人员而言仍是一大难题。在此背景下，工业机器人加工工艺应用程序的开发成为行业的热点，尤其是打磨工艺^[2]。作为众多生产环节中不可或缺的重要工序，打磨技术广泛应用于汽车制造、航空航天、机械加工、电子设备等多个领域，其主要目的是去除工件表面多余材料，提升表面质量，确保尺寸精度和表面光洁度。然而，传统打磨工艺在实际应用中面临诸多挑战。由于高度依赖操作工的经验和技能，存在效率低、质量不稳定、劳动强度大、危险性高等问题^[3-5]。

随着制造业迈向智能化，打磨机器人凭借其高效、精准的特性成为生产环节中的重要力量，推动了行业的技术革新。然而，传统打磨机器人的操作复杂性和潜在危险性，在很大程度上限制了其在更广泛场景中的应用^[6]。操作人员往往需要经过繁琐且漫长的学习过程，才能熟练驾驭机器人的编程与操作技能，这不仅耗费大量人力成本，也使得技术的普及面临障碍^[7]。与此同时，在一些极端环境中，人工操作打磨机器人不仅效率低下，且存在严重的安全隐患，这无疑成为打磨机器人技术进一步拓展应用领域的“绊脚石”。

近年来，虚拟现实（Virtual Reality，VR）^[8] *MERGEFORMAT *MERGEFORMAT 8^[10] 领域取得显著突破，为解决传统打磨工艺中的诸多挑战提供了全新思路。作为一种通过计算机生成虚拟环境并实现人机交互的技术，虚拟现实凭借其沉浸性、交互性和多感知性^[11,12]三大核心特征，能够为用户创造身临其境的体验，使其在虚拟环境中进行各类操作并实时获得反馈。随着硬件设备的不断升级和软件开发工具的日益完善，虚拟现实技术已在工业设计^[13-15]、教育培训、医疗手术等领域实现了深度应用，并展现出强大的发展潜力和广阔的前景，为各行业的数字化转型和创新发展注入了新的活力。

将虚拟现实技术应用于打磨机器人领域，为解决打磨工艺过程中存在的不确定性与操作复杂性提供了创新性解决方案。在工艺验证阶段，操作者可以在虚拟环境中进行打磨路径规划和工艺模拟，从而减少实际调试时间，提高工艺验证效

率^[16,17]。同时，通过虚拟环境中的多次模拟和优化，实际操作中的工艺错误和材料浪费得以有效减少，进而降低生产成本。在操作实施环节，虚拟现实系统能够实现虚拟环境与实际设备的实时数据交互，使操作者可以远程控制打磨机器人，确保操作精度的同时远离危险工作环境。系统提供的实时视觉反馈和力觉反馈，让操作者能够精确控制打磨过程，提升打磨质量和产品合格率。这些优势使虚拟现实技术成为打磨机器人领域实现智能化、精准化操作的重要技术支撑，为打磨工艺的转型升级注入了强大动力。

1.2 国内外研究现状

随着工业 4.0 与智能制造的快速发展，传统打磨机器人面临交互效率低、适应性差等瓶颈^[18]。虚拟现实（VR）技术以其沉浸式交互与高仿真能力，为打磨工艺虚拟化提供了全新的技术路径。通过构建虚实融合的作业环境，VR 技术不仅能够提升操作精度与效率，还可实现复杂场景下的智能决策与人机协作，成为推动工业机器人技术革新的重要驱动力^[19]。

1.2.1 打磨机器人研究现状

打磨机器人作为工业自动化领域的重要技术装备，近年来在国内外得到了广泛研究和实践应用。其高效、精准的作业能力不仅显著提升了生产效率，还改善了工作环境，降低了人工劳动强度，成为汽车制造、航空航天、家电生产等高精度需求领域不可或缺的技术手段^[20]。

德国的 KUKA 公司研发了一种型号为 KR 120 R2900 Fextra 打磨机器人^[21]，如图 1-1 所示，采用 750 公斤负载的三轴定位机构与机器人结合，能够在不损害研磨设备的前提下，适应多种异形毛刺的处理环境，同时确保加工效果的高标准。



图 1-1 KUKA 打磨机器人

Song^[22]提出了一种基于力控制的高精度机器人去毛刺技术，如图 1-2 所示，该技术通过整合 CAD 模型数据，实现了对复杂几何形状工件的精准处理。研究聚焦于通过自动化机器人去毛刺工艺，解决传统手工去毛刺过程中存在的噪声、灰尘、振动等对工人有害的问题，并提高去毛刺的精度和效率。Li^[23]提出了一种用于机器人抛光的新的高带宽末端执行器，该执行器带有主动力控制功能。将这种末端执行器作为微型机器人，构建了一个用于抛光加工的宏-微机器人系统，如图 1-3 所示。其中宏机器人在打磨操作中提供姿态控制，而微机器人则提供恒力控制。通过在这种配置中最小化主轴方向的惯性，末端执行器获得了 200 Hz 的力控带宽。通过不同接触力和进给速度的对比实验，证明了所提出的宏-微机器人控制力的方法，具有更小的超调量、更快的响应速度和更短的稳定时间。

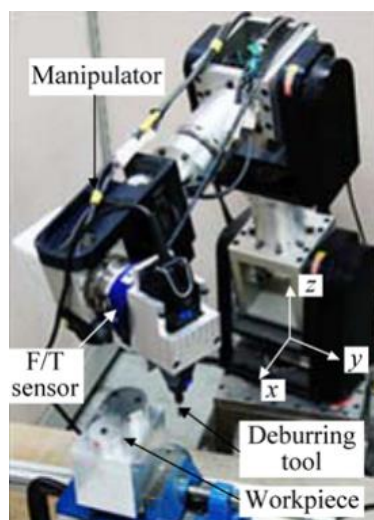


图 1-2 去毛刺机器人



图 1-3 机器人打磨平台

孙一兰^[24]针对导弹弹体内壁隔热层的特殊打磨需求,设计了一种四自由度打磨机器人,通过 S 形加减速规划和等弧长时间分割法实现高精度、平稳的运动轨迹,显著提升了打磨效率与质量。郭兆根^[25]通过运动学和动力学分析,建立了机器人模型,并在笛卡尔空间和关节空间内分别采用多种插补算法进行轨迹规划,结合遗传算法优化打磨机器人的轨迹时间效率。同时,基于阻抗控制原理,提出了一种自适应柔顺控制方法,通过实验验证,证明了该方法在打磨过程中能够有效稳定接触力,提高打磨质量和效率。

1.2.2 虚拟现实仿真技术研究现状

随着全球工业化进程的加速,制造业面临着日益激烈的竞争和不断变化的市场需求。在这样的背景下,虚拟现实仿真技术应运而生^[26],成为提升制造业竞争力的关键技术之一。工艺虚拟化通过虚拟现实、计算机仿真等技术手段,对生产过程进行数字化建模和仿真分析,从而在虚拟环境中实现对生产工艺的优化和验证^[27-29]。这不仅能够显著提高生产效率、降低成本,还能有效缩短产品上市时间,提升产品质量。

目前,国内外在虚拟仿真领域进行了大量研究。这些研究涉及多个方面,包括制造过程的建模、仿真、优化以及数字孪生技术的应用等^[30]。针对传统深孔加工实验教学中存在的成本高、安全性低等问题,刘晓峰^[31]提出了一种基于虚拟现实技术的解决方案。以 T2120 深孔加工机床为原型,利用 SolidWorks 和 3Ds MAX 建立机床的三维模型,并通过 Unity3D 开发工具构建了虚拟加工环境,结合深孔加工相关理论知识,开发了虚拟深孔加工系统,如图 1-4 所示,实现了切削力和切削温度的可视化仿真。

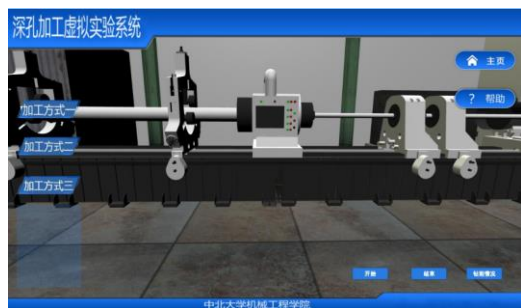


图 1-4 虚拟深孔加工系统图

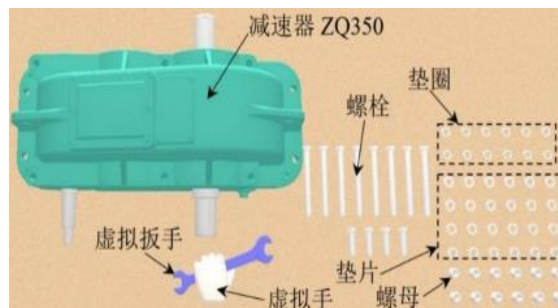


图 1-5 虚拟减速箱装配系统

刘家武^[32]通过自主研发的六维力觉交互设备,构建了虚拟减速箱装配仿真环境,如图 1-5 所示。该平台实现了人机之间的实时力学信息传递,操作人员可

根据触觉反馈灵活调整装配动作，显著增强了虚拟现实系统的沉浸感和交互性能。倪建龙^[33]设计了一套虚拟砂带研磨教学平台，如图 1-6 所示。该平台运用了自主研发的抛光力学模型与加工效果模拟算法，使用户能够在虚拟环境中感受到接近真实的抛光过程。通过与实际生产数据的对比分析，验证了该平台在教学应用中的有效性，达到了预期的教学目标。

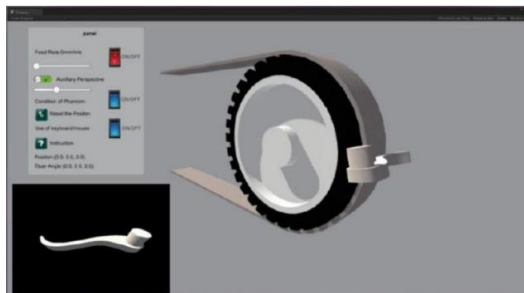


图 1-6 虚拟砂带研抛工艺示教系统

任豪豪^[34]以 IRB 1410 焊接机器人为基础，利用 Unity3D 开发了一套交互式焊接仿真系统，如图 1-7 所示。该系统通过建立焊接机器人的运动学正逆解模型，结合焊接路径的智能规划，并借助 Unity3D 的粒子效果、用户界面、音频模块以及硬件交互等技术，打造了一个兼具真实体验与互动功能的虚拟焊接环境。

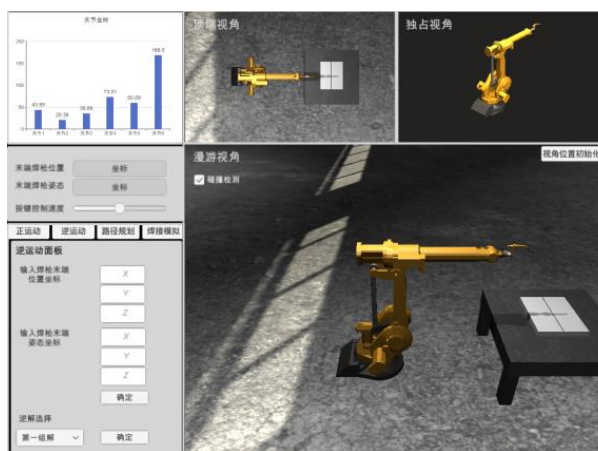


图 1-7 虚拟焊接机器人仿真系统

唐磊^[35]以铸钢件飞边打磨为背景，通过结合点云三维重建技术和改进的 D-H 参数法，开发了一套完整的机器人打磨虚拟仿真系统，如图 1-8 所示。系统不仅实现了铸钢件三维模型的快速重建和打磨路径规划，还通过 Unity3D 平台实现了打磨过程的实时可视化和虚拟交互。Ni^[36]提出了一种直观的用户界面，用于使用带有触觉反馈的增强现实（AR）对焊接机器人进行远程编程，使用深度相机来重建工件的表面，并结合触觉输入设备，用户能够直接定义焊接路径。此外，开

发的 AR 用户界面, 允许用户可视化和调整焊枪的方向, 实现了焊接工艺虚拟化, 如图 1-9 所示。

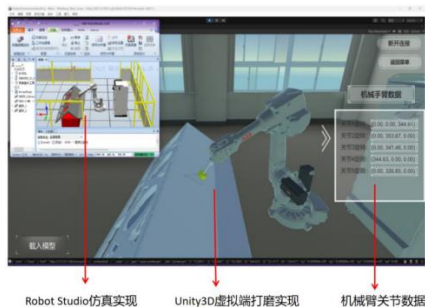


图 1-8 虚拟打磨机器人系统



图 1-9 焊接机器人视觉 AR 和触觉界面

Togias^[37]提出了一种基于虚拟现实(VR)的遥操作系统, 旨在通过直观的 VR 界面实现工业机器人的远程编程和控制, 从而提高现有工作站的灵活性。系统允许用户在虚拟环境中设计机器人路径和动作, 并通过模拟验证后推送到实际机器人控制器执行。Lin^[38]基于 Unity3D 引擎, 开发了一套机器人虚拟实验室系统, 该系统旨在借助虚拟仿真技术, 显著提升机器人结构相关教学与训练的成效。该系统包括主界面、装配爆炸功能按钮、自动装配功能按钮和自动显示功能按钮, 如图 1-10 所示。通过运用 3D 建模与虚拟现实技术, 成功实现了机器人的虚拟装配、动态展示以及人机交互功能。学生能够以第一人称视角在虚拟环境中自由漫游并进行操作, 沉浸式体验机器人相关知识的学习与实践过程。

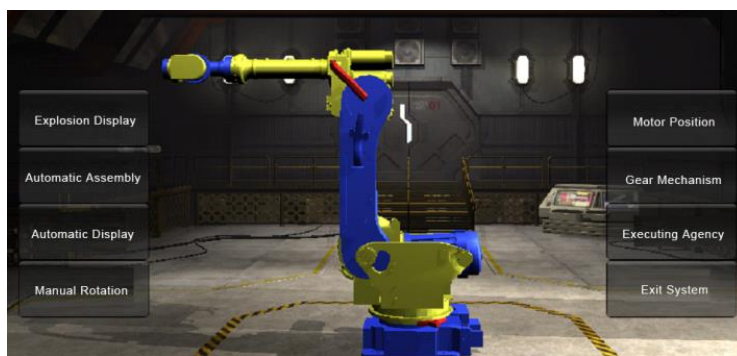


图 1-10 机器人虚拟实验室系统

1.2.3 遥操作机器人国内外研究现状

遥操作机器人凭借其在高危及复杂环境下的独特优势, 成为机器人技术与远程通信技术融合的前沿领域。其在空间探索、深海作业、灾难救援、医疗手术等关键场景中的应用, 不仅提升了任务效率, 还显著降低了人员风险。

Xia^[39]提出了一种基于虚拟现实（VR）和触觉模拟器的分层直观控制方法，用于增强水下遥控操作车（ROV）的远程操作。通过分布式传感器系统收集水下环境的水动力数据，并利用数字孪生（DT）模块进行数据融合和模拟，生成多层次的感官反馈，包括远场增强的 3D 视觉反馈、近场触觉服反馈和微场触觉手套反馈。该系统旨在通过沉浸式的感官反馈，提升操作员对 ROV 工作环境的感知能力，降低操作难度和错误率，从而在复杂的水下任务中提高操作效率和安全性。

日本航天局研制远程机器人 JEMRMS^[40, 41]，如图 1-11 所示。该系统从端部分由两台六自由度机械臂及感知模块组成，感知模块集成了视觉检测装置以及机械臂末端的力/位感知单元；主端部分配备了两台操控设备及虚拟显示界面。通过轨道远程操控技术，该系统成功实现了空间站对接与组件安装等核心操作，并对机械臂在反向作用力下产生的振动现象进行了深入研究。

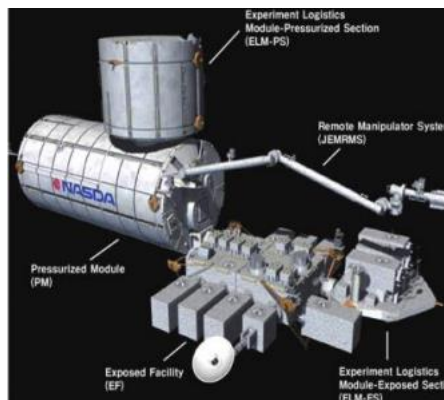


图 1-11 JEMRMS 遥操作空间机器人

Su^[42]提出了一种基于隐马尔可夫模型（HMM）的混合现实增强遥操作框架，如图 1-12 所示。他们利用 HMM 生成算法学习和预测焊工的运动，结合平滑和预测技术，有效减少了遥操作中的时间延迟。实验结果表明，该算法显著降低了均方根误差（RMSE），相比未使用 HMM 的系统减少了 66%，并可能进一步优化至 38%。该研究为低成本、高效的遥操作焊接提供了技术支持，提升了操作成功率和焊接质量。

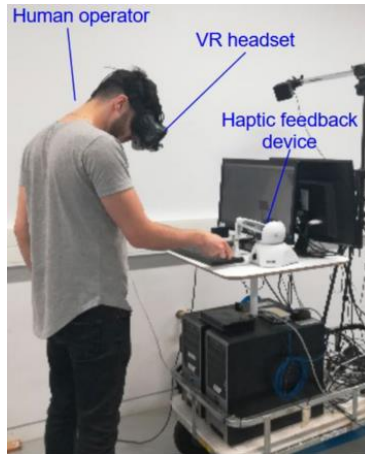


图 1-12 遥操作焊接机器人

中国空间站机器人系统（CSSRMS），如图 1-13 所示。凭借其先进的遥操作技术，显著提升了空间任务的执行效率。该系统由“天和”核心舱机器人与“问天”实验舱机器人组成，具备单舱及跨舱爬行能力，并可通过级联方式形成组合臂，有效扩展操作范围。在路径规划方面，CSSRMS 采用最短直线规划的笛卡尔空间路径规划方法，结合二次规划优化算法进行关节空间构型规划，确保操作的高效性和精准性。此外，其高精度视觉系统为地面控制人员提供了实时反馈，增强了任务执行的可靠性和灵活性^[43]。

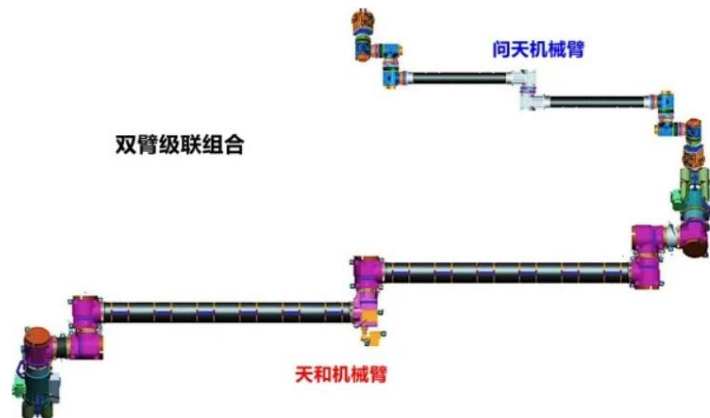


图 1-13 空间站机器人系统

新冠疫情期间，咽拭子采样任务艰巨，宋甫俊^[44]在其研究中提出基于人机共享控制的咽拭子机器人采样策略。他设计三状态协商机制与意图估计的共享控制策略用于侵入过程，搭建混合共享控制框架和虚拟约束模型用于刮拭过程，并搭建实验系统，如图 1-14 所示。经实验验证，该策略提升了任务执行效率、灵活性和智能化程度，为低成本人机协同采样提供了支撑。

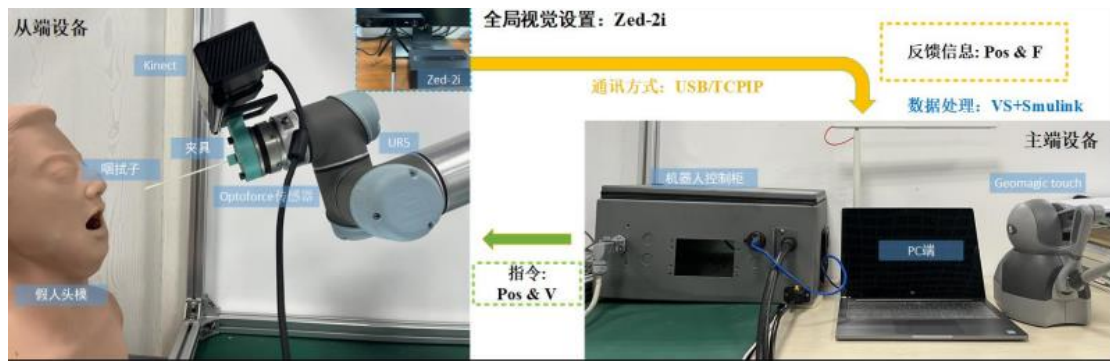


图 1-14 遥操作咽拭子机器人

综上所述，当前国内外在关于打磨机器人、机器人虚拟仿真技术以及遥操作机器人上已经有了一定的研究，但是在虚拟现实与打磨机器人交互技术方面打磨作业缺乏高保真的视觉-力觉融合反馈，来模拟打磨过程中的实时的材料去除效果与力觉交互，因此，借助虚拟现实技术，研究基于虚拟现实的打磨机器人交互技术，使得打磨机器人在虚拟环境中的预演更加真实可靠，进一步提升了遥操作的安全性，同时也为训练平台的高效运行奠定了基础。

1.3 主要研究内容

本文将机器人技术与虚拟现实技术相结合，探索两者在打磨领域的深度融合。通过解决异构模型差异导致的控制精度不足、打磨工艺过程中存在的不确定性与操作复杂性，以及打磨作业缺乏高保真的视觉-力觉融合反馈等问题，设计了一个基于虚拟现实打磨机器人交互平台，以实现更高效、更精确的人机交互。本文各部分研究内容如下：

第一章，介绍了课题的研究背景及意义，阐述了打磨机器人、虚拟现实仿真技术和遥操作机器人领域的国内外研究现状，基于现有研究的技术缺口，明确了本文的研究方向，对本文的主要研究内容进行了简要概括。

第二章，对操作端和执行端的机器人进行了选型，并建立了相应的运动学模型。通过连杆坐标系的建立、正运动学仿真和计算，以及运动学正解和逆解，对操作端和执行端的运动学进行了深入分析。探讨了基于笛卡尔空间的映射方法，以及如何实现异构模型之间的空间映射，为虚实交互奠定了基础。

第三章，聚焦于打磨机器人交互系统的设计，首先提出了系统的总体框架、功能需求，虚拟场景建模以及用户界面控制界面的设计，重点研究了动态创建工

件网格模型以及球形打磨头的变形算法，以实现打磨过程中实时的材料去除效果。

第四章，系统的研究了虚拟现实中的交互与控制技术，涉及碰撞检测（包括层次包围盒算法和碰撞体与刚体的处理）、虚拟现实交互技术（力觉交互和 VR 交互）、坐标系转换、打磨机器人运动学算法的验证以及虚实控制研究。通过整合这些技术，研究旨在提升虚拟环境与物理实体的协同操作能力。

第五章，设计了虚拟现实打磨机器人交互系统的测试实验，包括实验平台搭建、位置映射实验、力觉交互实验以及虚实打磨实验，并对实验的结果进行了分析。

第六章,总结全文的主要内容，并对当前存在的不足以及未来需要进行的工作进行了展望。

第2章 操作端与执行端的运动学建模与空间映射

机器人运动学分析是实现精准遥操作的核心基础，其核心在于建立机器人关节角度与末端执行器空间坐标之间的数学关系，并通过运动学正解和逆解实现操作端与执行端的运动关联^[45]。在遥操作过程中，操作端通过运动学正解计算末端坐标，经映射处理后发送至执行端，执行端通过运动学逆解求解关节角度并实现控制^[46]。由于操作端与执行端通常为异构系统^[47]，直接基于关节角度的映射方式难以实现精确控制，因此采用基于笛卡尔空间映射方法结合蒙特卡洛法，建立两者工作空间的映射关系。这种方法不仅克服了异构系统的控制难题，还显著提升了遥操作系统的控制精度和整体性能，为实现复杂环境下的精准遥操作提供了可靠的技术支持。

2.1 操作端与执行端选型

本文选用 3D Systems 公司研发的 Geomagic Touch 六自由度触觉交互设备作为操作端，如图 2-1 所示。该装置基于六自由度串联机器人设计，可精确获取空间位置、姿态及运动参数等实时信息。借助这一设备，操作人员不仅可以远程操控机器人的运动，还能获得真实的力觉反馈，具备响应灵敏、操控直观等优势，具体参数见表 2-1。



图 2-1 Geomagic Touch 操作端设备

表 2-1 Geomagic Touch 参数表

工作空间	自由度	二次开发	最大可施加力
160×120×70 mm	6	OpenHaptics SDK	3.3 N

执行端是睿尔曼公司的六自由度超轻量仿人机器人 RM65-B,如图 2-2 所示。该机器人具备 $\pm 0.05\text{ mm}$ 的重复定位精度,并配备电流、位置、速度及力矩的精确测量功能。通过以太网通信接口,机器人可快速响应指令并实现与现代工业网络的集成。此外, RM65-B 提供多语言 SDK (C/C++/Python) 的完善软件开发支持,便于二次开发。凭借其高精度、强负载和灵活的操作性能,该机器人特别适用于复杂工业自动化场景。具体参数见表 2-2。



图 2-2 执行端机器人

表 2-2 RM65-B 部分参数

型号	RM65-B
自由度	6
额定负载	5 kg
重复定位精度	$\pm 0.05\text{ mm}$
工作半径	610 mm
通讯方式	WIFI/网口/蓝牙/USB 串口/RS485
控制方式	拖拽示教/示教器/API/JSON
关节运动范围	J1 $\pm 178^\circ$ 、J2 $\pm 130^\circ$ 、J3 $\pm 135^\circ$ 、J4 $\pm 178^\circ$ 、J5 $\pm 128^\circ$ 、J6 $\pm 360^\circ$
关节最大速度	J1-J2 180 $^\circ/\text{s}$ 、J3-J6 225 $^\circ/\text{s}$

2.2 操作端的运动学分析

2.2.1 连杆坐标系建立

改进的 D-H 参数是机器人运动学中一种用于描述机械臂连杆之间几何关系的参数化方法^[48]。它通过优化坐标系定义和参数化方式,使得参数化过程更加直观且易于理解。改进的 D-H 参数包含四个核心参数,分别描述了相邻连杆之间的空间关系: a_{i-1} 表示两个关节公垂线的长度,即连杆长度, α_{i-1} 表示关节轴从 $i-1$ 转到 i 的转动角度, θ_i 表示绕公共轴线的夹角, d_i 表示公共轴线间的距离。本文采用改进的 D-H 参数法,构建了操作端的改进 D-H 参数,如图 2-3 所示。

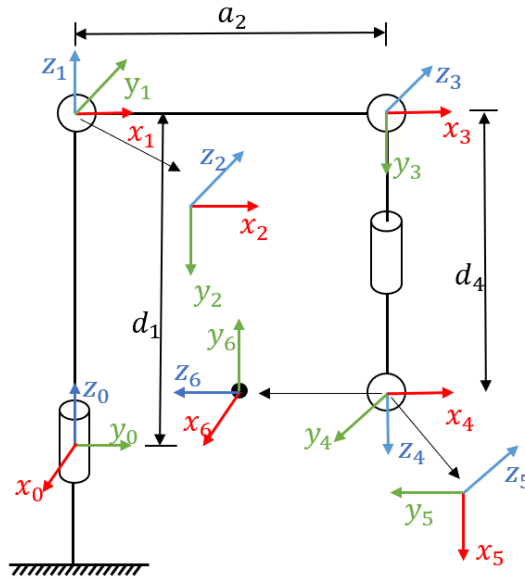


图 2-3 操作端设备的改进 D-H 坐标系

根据改进 D-H 参数法,建立操作端的 D-H 参数如表 2-3 所示。

表 2-3 操作端设备的改进 D-H 参数

连杆 i	a_{i-1} (mm)	α_{i-1} (°)	d_i (mm)	θ_i (°)	关节角范围 (°)
1	0	0	179.12	90	(-56, 56)
2	0	-90	0	0	(-100, 0)
3	133.35	0	0	0	(-47, 69)
4	0	-90	133.35	0	(-141, 148)
5	0	90	0	90	(-87, 57)
6	0	-90	0	90	(-150, 150)

2.2.2 正运动学计算

在机器人学中，已知机器人的各个关节的旋转角，可以在机器人相邻的连杆坐标系之间构建一个运动学方程，这个方程由 4×4 齐次矩阵变换而成^[49]。在机器人学的领域中，机器人的相邻坐标系 $i-1$ 和 i 之间存在着如下的变换矩阵：

$${}_{i-1}^iT = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & a_{i-1} \\ \sin \theta_i \cos \alpha_{i-1} & \cos \theta_i \cos \alpha_{i-1} & -\sin \alpha_{i-1} & -d_i \sin \alpha_{i-1} \\ \sin \theta_i \sin \alpha_{i-1} & \cos \theta_i \sin \alpha_{i-1} & \cos \alpha_{i-1} & d_i \cos \alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-1)$$

则机器人六个相邻连杆间的齐次变换矩阵为：

$$\begin{aligned} {}^0T_1 &= \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ -s_1 & c_1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^1T_2 &= \begin{bmatrix} c_2 & -s_2 & 0 & a_1 \\ 0 & 0 & 0 & 1 \\ -s_2 & -c_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^2T_3 &= \begin{bmatrix} c_3 & -s_3 & 0 & a_2 \\ s_3 & c_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ {}^4T_3 &= \begin{bmatrix} c_4 & -s_4 & 0 & a_3 \\ 0 & 0 & 1 & d_4 \\ -s_4 & -c_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^4T_5 &= \begin{bmatrix} c_5 & -s_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s_5 & c_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & {}^5T_6 &= \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s_6 & -c_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}$$

其中， c_i 是 $\cos \theta_i$ 的简写， s_i 是 $\sin \theta_i$ 的简写， c_{ij} 是 $\cos(\theta_i + \theta_j)$ 的简写， s_{ij} 是 $\sin(\theta_i + \theta_j)$ 的简写。

因此操作末端执行器的位姿矩阵可表示为：

$${}^0iT = {}^0T_1 {}^1T_2 {}^2T_3 \dots {}^{i-1}T_i = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^0R_{3 \times 3} & {}^0P_{3 \times 1} \\ 0 & 1 \end{bmatrix} \quad (2-2)$$

式中，矩阵 $\mathbf{R}$ 为末端相对于基坐标系的姿态信息，矩阵 $\mathbf{P}$ 为末端相对于基坐标系的位置矢量。向量 $\mathbf{n}$ 、 $\mathbf{o}$ 、 $\mathbf{a}$ 为操作末端坐标系在基坐标系下的姿态矢量，向量 $\mathbf{p}$ 为末端坐标系在基坐标系下的位置矢量。其中：

$$\mathbf{n} = \begin{bmatrix} s_1 s_4 c_5 c_6 + c_1 s_2 (s_3 c_4 + c_3 s_5 - s_3 s_4) - c_1 c_2 (2c_3 + s_3) + s_1 c_4 s_6 \\ -c_1 s_4 c_5 c_6 + s_1 s_2 (s_3 s_4 - s_3 c_4 - c_3 c_5) - s_1 c_2 (2c_3 + s_3) - c_1 c_4 s_6 \\ s_2 s_3 s_5 c_6 + s_2 c_3 (s_4 s_6 - c_4 c_5) - c_2 (2s_3 + c_3) \end{bmatrix} \quad (2-3)$$

$$o = \begin{bmatrix} s_1 c_4 c_6 + c_1 s_2 (s_3 c_4 - s_3 s_4 + c_3 s_5) - c_1 c_2 (2c_3 + s_3) - s_1 s_4 c_5 s_6 \\ c_1 s_4 c_5 s_6 + s_1 s_2 (s_3 s_4 - s_3 c_4 - c_3 s_5) - s_1 c_2 (2c_3 + s_3) - c_1 c_4 c_6 \\ -s_2 s_3 s_5 s_6 + s_2 s_3 (s_4 c_6 - c_4 c_5) - c_2 (2s_3 + c_3) \end{bmatrix} \quad (2-4)$$

$$a = \begin{bmatrix} s_1 s_4 s_5 + c_1 s_2 (s_3 c_4 - c_3 s_5) - c_1 c_2 (c_3 + s_3) \\ -s_1 s_2 (c_3 c_5 + s_3 c_4) - c_1 s_4 s_5 - s_1 c_2 (s_3 + c_3) \\ -s_2 c_3 c_4 s_5 + s_2 s_3 c_5 + c_2 (s_3 + c_3) \end{bmatrix} \quad (2-5)$$

$$p = \begin{bmatrix} d_6 s_1 s_4 s_5 + c_1 s_2 s_3 c_4 + c_1 c_2 (c_3 + s_3) - d_6 c_1 s_2 c_3 c_5 - a_2 c_1 s_2 + d_4 c_1 s_{23} \\ -a_2 s_1 s_2 - d_6 c_1 s_4 s_5 - s_1 s_2 s_3 c_4 + s_1 c_2 (s_3 + c_3) - d_6 s_1 s_2 c_3 c_5 - d_4 s_1 s_{23} \\ a_2 c_2 - d_6 s_2 (s_3 c_5 + c_3 c_4 s_5) - c_2 (s_3 + c_3) + d_4 c_{23} + d_1 \end{bmatrix} \quad (2-6)$$

由此可根据各关节角度值求解操作端的运动正解。

2.2.3 正运动学仿真

本文采用 MATLAB 中 Robotics Toolbox 工具箱对目标进行运动学仿真验证，通过上述的操作端设备各关节的 D-H 参数在工具箱中使用 Link 函数创建各连杆，并利用 SerialLink 函数将各连杆连接后得到机器人模型，最后运用 Teach 函数创建了机器人可视化图形界面，如图 2-4 所示。

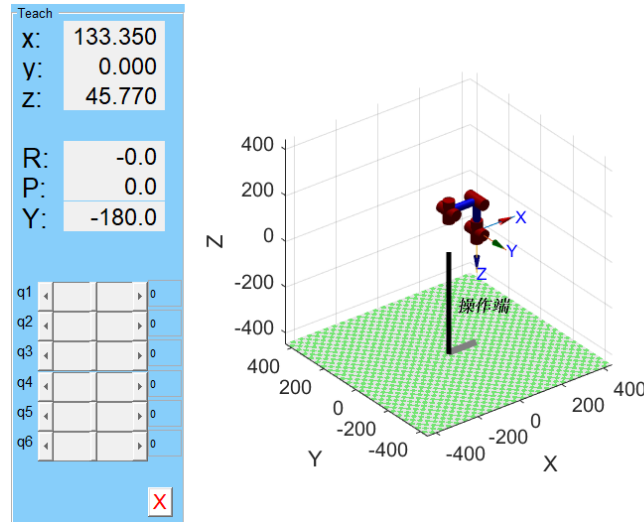


图 2-4 操作端的运动学建模

通过 fkine 函数求解机械臂末端执行器的位姿，任取一组关节角度值 $\pi/3$ 、 $\pi/3$ 、 $\pi/4$ 、 $\pi/2$ 、 $\pi/6$ 、 $\pi/4$ ，计算末端位姿矩阵如下：

$${}^0_6T \left(\frac{\pi}{3}, \frac{\pi}{3}, \frac{\pi}{4}, \frac{\pi}{2}, \frac{\pi}{6}, \frac{\pi}{4} \right) = \begin{bmatrix} -0.494 & -0.177 & -0.8513 & 53.807 \\ 0.6934 & 0.5104 & -0.5085 & -31.066 \\ -0.5245 & 0.8415 & -0.1294 & 98.149 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-7)$$

式 (2-7) 中末端位置与图 2-5 中的结果完全一致, 表明机器人正运动学模型建立正确。

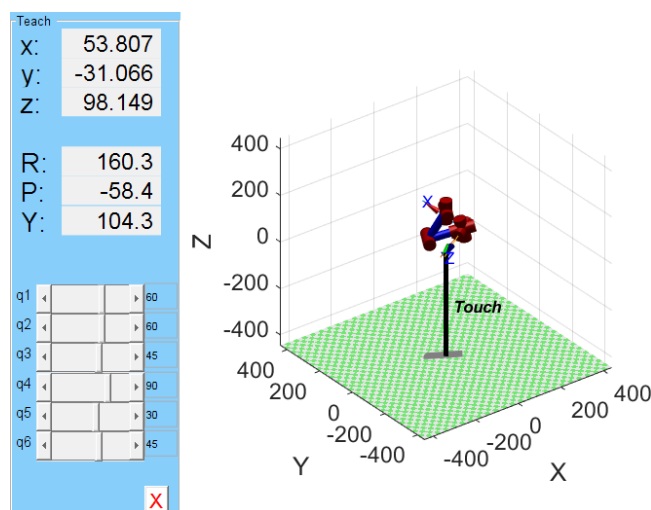


图 2-5 操作端正运动学仿真

2.3 执行端的运动学分析

2.3.1 连杆坐标系建立

执行端机器人的结构由一系列连杆通过多个关节串联构成, 形成一个空间开链式运动系统。该系统具备六个自由度, 其中前三个关节主要负责控制机器人末端的位置, 而后三个关节则用于调整末端的姿态。在机器人的末端, 通过采用 3D 打印的连接件, 将六维力传感器和打磨机安装在执行端机器人上, 并建立了相应的连杆坐标系。图 2-6 是执行端机器人的改进 D-H 坐标系, 表 2-4 为执行端机器人的改进 D-H 参数。

表 2-4 执行端的改进 D-H 参数

连杆 i	a_{i-1} (mm)	α_{i-1} ($^{\circ}$)	d_i (mm)	θ_i ($^{\circ}$)	关节角范围 ($^{\circ}$)
1	0	0	240.5	0	(-178, 178)
2	0	90	0	90	(-130, 130)
3	256	0	0	90	(-135, 135)
4	0	90	210	0	(-178, 178)
5	0	-90	0	0	(-128, 128)
6	0	90	426.5	0	(-360, 360)

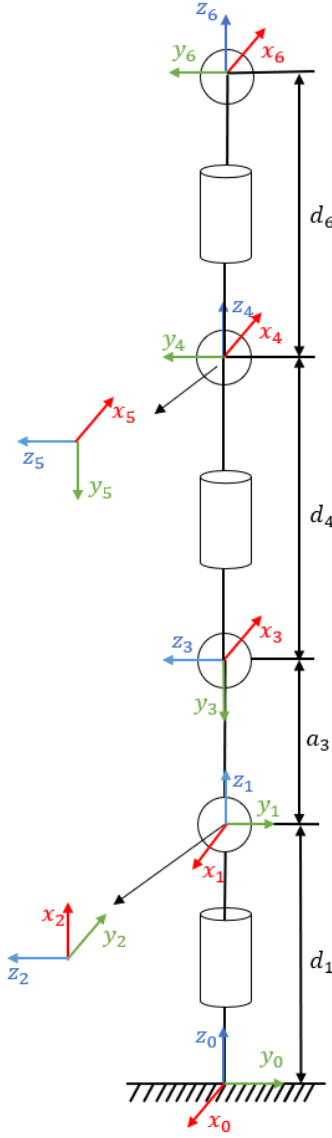


图 2-6 执行端的改进 D-H 参数坐标系

2.3.2 运动学正解

根据式(2-1)求解执行端各连杆之间的齐次变换矩阵，结合改进的 D-H 参数表，可得到执行端机器人相邻连杆之间的变换矩阵：

$${}^0_i T = {}^0_1 T {}^1_2 T {}^2_3 T \dots {}^{i-1}_i T = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^0_i R_{3 \times 3} & {}^0_i P_{3 \times 1} \\ 0 & 1 \end{bmatrix} \quad (2-8)$$

$$n = \begin{bmatrix} s_1 s_4 c_5 c_6 + c_1 s_2 (s_3 c_4 + c_3 s_5 - s_3 s_4) - c_1 c_2 (2c_3 + s_3) + s_1 c_4 s_6 \\ -c_1 s_4 c_5 c_6 + s_1 s_2 (s_3 s_4 - s_3 c_4 - c_3 c_5) - s_1 c_2 (2c_3 + s_3) - c_1 c_4 s_6 \\ s_2 s_3 s_5 c_6 + s_2 c_3 (s_4 s_6 - c_4 c_5) - c_2 (2s_3 + c_3) \end{bmatrix} \quad (2-9)$$

$$o = \begin{bmatrix} s_1 c_4 c_6 + c_1 s_2 (s_3 c_4 - s_3 s_4 + c_3 s_5) - c_1 c_2 (2c_3 + s_3) - s_1 s_4 c_5 s_6 \\ c_1 s_4 c_5 s_6 + s_1 s_2 (s_3 s_4 - s_3 c_4 - c_3 s_5) - s_1 c_2 (2c_3 + s_3) - c_1 c_4 c_6 \\ -s_2 s_3 s_5 s_6 + s_2 s_3 (s_4 c_6 - c_4 c_5) - c_2 (2s_3 + c_3) \end{bmatrix} \quad (2-10)$$

$$a = \begin{bmatrix} s_1 s_4 s_5 + c_1 s_2 (s_3 c_4 - c_3 s_5) - c_1 c_2 (c_3 + s_3) \\ -s_1 s_2 (c_3 c_5 + s_3 c_4) - c_1 s_4 s_5 - s_1 c_2 (s_3 + c_3) \\ -s_2 c_3 c_4 s_5 + s_2 s_3 c_5 + c_2 (s_3 + c_3) \end{bmatrix} \quad (2-11)$$

$$p = \begin{bmatrix} d_6 s_1 s_4 s_5 + c_1 s_2 s_3 c_4 + c_1 c_2 (c_3 + s_3) - d_6 c_1 s_2 c_3 c_5 - a_2 c_1 s_2 + d_4 c_1 s_{23} \\ -a_2 s_1 s_2 - d_6 c_1 s_4 s_5 - s_1 s_2 s_3 c_4 + s_1 c_2 (s_3 + c_3) - d_6 s_1 s_2 c_3 c_5 - d_4 s_1 s_{23} \\ a_2 c_2 - d_6 s_2 (s_3 c_5 + c_3 c_4 s_5) - c_2 (s_3 + c_3) + d_4 c_{23} + d_1 \end{bmatrix} \quad (2-12)$$

综上所述，便可得到执行端机器人的正运动学方程，可根据各关节旋转角度值，求取其运动学正解。

2.3.3 运动学逆解

逆运动学是根据机器人末端执行器相对于基坐标系的位姿，反解出机器人各关节的角度值，其分析求解过程与正运动学相反，求解方程是一个非线性问题，常见求解方法主要分为解析法和数值法两类^[50]。以六自由度机器人为例，其逆运动学求解步骤为：先建立正向运动学模型，确定末端执行器相对于基座的位置和姿态；再根据末端执行器的期望位姿，利用解析法或数值法求解关节变量；最后验证求解结果并进行必要调整。

执行端机器人具有六个旋转关节的串联型六自由度机器人，且后三个关节的关节轴交于同一点，可以采用 Pieper 解法^[51]实现机器人逆运动学的求解。

当三个末端轴交汇时，坐标系{4}、{5}和{6}的原点均位于同一位置。这一位置在基础坐标系中的表示为：

$${}^0P_{4ORG} = {}^0_1T_2^1T_3^2T^3P_{4ORG} = [x \quad y \quad z \quad 1]^T \quad (2-13)$$

根据式（2-1）可知，当 $i = 4$ 时：

$${}^0P_{4ORG} = {}^0_1T_2^1T_3^2T^3P_{4ORG} = {}^0_1T_2^1T_3^2T^3 \begin{bmatrix} a_3 \\ -d_4 s \alpha_3 \\ d_4 c \alpha_3 \\ 1 \end{bmatrix} = {}^0_1T_2^1T^2T^3 \begin{bmatrix} f_1(\theta_3) \\ f_2(\theta_3) \\ f_3(\theta_3) \\ 1 \end{bmatrix} \quad (2-14)$$

其中 f_i 的表达式为：

$$f_1 = a_3 c_3 + d_4 s \alpha_3 s_3 + a_2 \quad (2-15)$$

$$f_2 = a_3 c \alpha_2 s_3 - d_4 s \alpha_3 c \alpha_3 c_3 - d_4 s \alpha_2 c \alpha_2 - d_3 s \alpha_2 \quad (2-16)$$

$$f_3 = a_3 s \alpha_2 s_3 - d_4 s \alpha_3 s \alpha_3 c_3 + d_4 c \alpha_2 c \alpha_2 + d_3 c \alpha_2 \quad (2-17)$$

对 0_1T 和 1_2T 应用齐次变换矩阵得到：

$${}^0_4P = [c_1 g_1 - s_1 g_2 \quad s_1 g_1 + c_1 g_2 \quad g_3 \quad 1]^T \quad (2-18)$$

g_1 、 g_2 和 g_3 的计算过程如下：

$$g_1 = c_2 f_1 - s_2 f_2 + a_1 \quad (2-19)$$

$$g_2 = s_2 c \alpha_1 f_1 + c_2 c \alpha_1 f_2 - s \alpha_1 f_3 - d_2 s \alpha_1 \quad (2-20)$$

$$g_3 = s_2 s \alpha_1 f_1 + c_2 s \alpha_1 f_2 + c \alpha_1 f_3 + d_2 c \alpha_1 \quad (2-21)$$

令 $r = x^2 + y^2 + z^2$ ，由式（2-19）、（2-20）、（2-21）可知：

$$r = g_1^2 + g_2^2 + g_3^2 = f_1^2 + f_2^2 + f_3^2 + a_1^2 + d_2^2 + 2d_2 f_3 + 2a_1(c_2 f_1 - s_2 f_2) \quad (2-22)$$

由式（2-22）写出 Z 轴分量的方程，可得系统的两个方程：

$$r = (k_1 c_2 + k_2 s_2) 2a_1 + k_3 \quad (2-23)$$

$$z = (k_1 s_2 - k_2 c_2) s \alpha_1 + k_4 \quad (2-24)$$

对于 k_1 、 k_2 、 k_3 、 k_4 ，其计算如下：

$$k_1 = f_1 \quad (2-25)$$

$$k_2 = -f_2 \quad (2-26)$$

$$k_3 = f_1^2 + f_2^2 + f_3^2 + a_1^2 + d_2^2 + 2d_2 f_3 \quad (2-27)$$

$$k_4 = f_3 c \alpha_1 + d_2 c \alpha_1 \quad (2-28)$$

之后求解 θ_3 ，分三种情况：

（1）若 $a_1 = 0$ ，则 $r = k_3$ ，此时 k_3 的右边是仅关于 θ_3 的函数。代入式（2-29）后，由包含 $\tan(\frac{\theta_3}{2})$ 的一元二次方程可以解出 θ_3 ；

$$u = \tan \frac{\theta}{2}, \quad \cos \theta = \frac{1-u^2}{1+u^2}, \quad \sin \theta = \frac{2u}{1+u^2} \quad (2-29)$$

（2）若 $s \alpha_1 = 0$ ，则 $z = k_4$ ，此时 z 已知。代入式（2-29）后利用一元二次方程解出 θ_3 ；

(3) 若不是上述两种情况, 从式 (2-23) 和式 (2-24) 中消去 s_2 和 c_2 , 得到

$$\frac{(r-k_3)^2}{4a_1^2} + \frac{(z-k_4)^2}{s^2\alpha_1} = k_1^2 + k_2^2 \quad (2-30)$$

代入式 (2-29) 解出 θ_3 后, 可得到一个一元四次方程, 由此解出 θ_3 。

在机器人逆运动学分析中, Pieper 方法用于计算前三个关节变量, 通常会产生 4 组或 8 组结果。后三个关节的角度主要用于确定末端执行器的姿态, 以通过一组欧拉角 (ψ, θ, φ) 表示, 该部分可通过欧拉角分解法进行计算。欧拉角变换 $Euler(\psi, \theta, \varphi)$ 可以通过三个旋转矩阵的乘积来表示:

$$\begin{aligned} Euler(\psi, \theta, \varphi) &= Rot(z, \psi)Rot(y, \theta)Rot(z, \varphi) \\ &= \begin{bmatrix} c_\psi c_\theta c_\varphi - s_\psi s_\varphi & -c_\psi c_\theta s_\varphi - s_\psi c_\varphi & c_\psi s_\theta \\ s_\psi c_\theta c_\varphi + c_\psi s_\varphi & -s_\psi c_\theta s_\varphi + c_\psi c_\varphi & s_\psi s_\theta \\ -s_\theta c_\varphi & s_\theta s_\varphi & c_\theta \end{bmatrix} \end{aligned} \quad (2-31)$$

根据姿态矩阵 R 可求出其对应的欧拉角:

$$\psi = atan2(a_y, a_x) + \pi \quad (2-32)$$

$$\theta = atan2(c_\psi a_x + s_\psi a_y, a_z) \quad (2-33)$$

$$\varphi = atan2(-s_\psi n_x + c_\psi n_y, -s_\psi o_x + c_\psi o_y) \quad (2-34)$$

假设建立欧拉角 $Euler(\psi, \theta, \varphi)$ 的坐标系为 $\{E\}$, 则从基坐标系 $\{0\}$ 到坐标系 $\{E\}$ 的其次变换矩阵为:

$${}^0_E T = {}^0_3 T Rot(x, \alpha_3) \quad (2-35)$$

式中, α_3 表示改进 D-H 参数表中第四轴的连杆扭角。欧拉角表示的姿态矩阵的计算方式如式 (2-8) 所示:

$$R = {}^0_E R^T \bullet {}^0_6 R \quad (2-36)$$

其中 ${}^0_E R^T$ 表示齐次变换矩阵 ${}^0_E T$ 中的旋转矩阵的转置; ${}^0_6 R$ 表示齐次变换矩阵 ${}^0_6 T$ 中的旋转矩阵。

可通过式 (2-32)、(2-33)、(2-34) 求出相应的欧拉角, 所求得的欧拉角与后三轴的关节变量的关系为:

$$\theta_4 = \psi, \theta_5 = -\theta, \theta_6 = \varphi \quad (2-37)$$

由于机器人腕关节的翻转，还可以得到后三轴的另一组解：

$$\theta'_4 = \theta_4 + \pi, \theta'_5 = -\theta_5, \theta'_6 = \theta_6 + \pi \quad (2-38)$$

通过 Pieper 方法可得多组解，依据关节运动范围的限制，可以排除部分无效解。在剩余的有效解中，通常选取与机器人当前位姿最为接近的“最短行程解”。

2.4 操作端与执行端的空间映射

由于操作端设备与执行端机器人在结构上存在显著差异，操作端的工作范围较小，而执行端机器人的工作空间较大。若直接通过操作端的关节角度变换矩阵映射到执行端机器人各关节进行控制，会导致机器人可作业范围受限且精度降低。相比之下，笛卡尔空间映射方法对末端执行器的位置控制具有更高的精度和灵活性^[52]。因此，本系统采用基于笛卡尔空间的映射方法，以实现更精确和高效的控制。

2.4.1 基于笛卡尔空间的映射

操作端设备与执行端工作空间的映射方法，如图 2-7 所示。首先，分析操作端与执行端工作空间的差异性；其次，依据工作空间的差异性，设计合适的映射函数并配置相应参数；再次，实时获取操作端设备的各个关节角度数据，运用正运动学来计算其位姿，再将位姿数据输入至映射函数之中，算出执行端的目标位姿；最后，以目标位姿为基础，通过逆运动学求出执行端各关节的转角，保证其能够抵达预定的目标位姿。

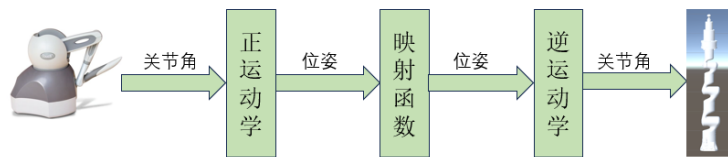


图 2-7 空间映射方法

2.4.2 工作空间差异性分析

操作端和执行端工作空间的求解是实现操作端与执行端空间映射的前提。本文利用 MATLAB，采用蒙特卡洛法^[53]，根据操作端和执行端的改进 D-H 参数和各关节运动角度范围，进行工作空间求取。

(1) 操作端工作空间的求取

在 MATLAB 平台中，借助 Robotics Toolbox 工具箱，通过配置 D-H 参数构建操作端设备的模型。设置随机值数量 N 为 100000，生成随机数据并计算运动学正解，最终得到操作端设备工作空间的点云分布，如图 2-8 所示。

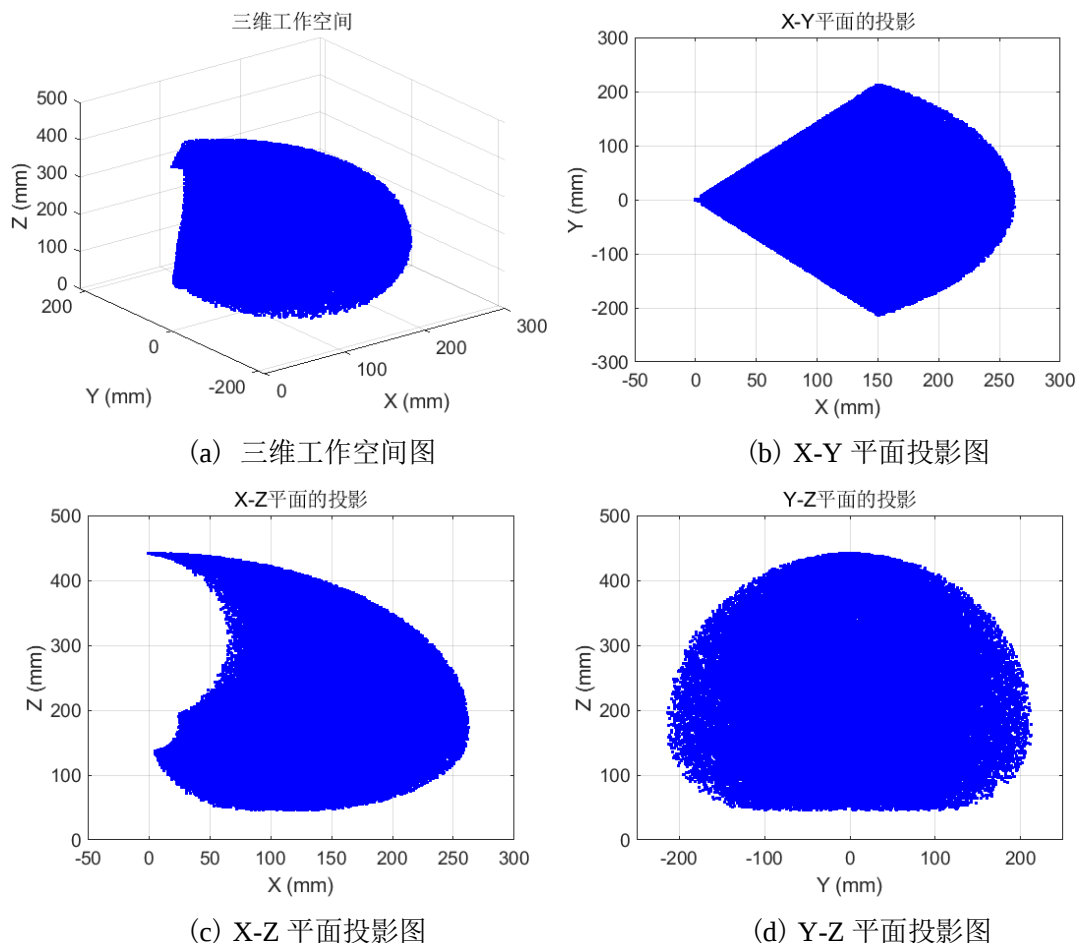
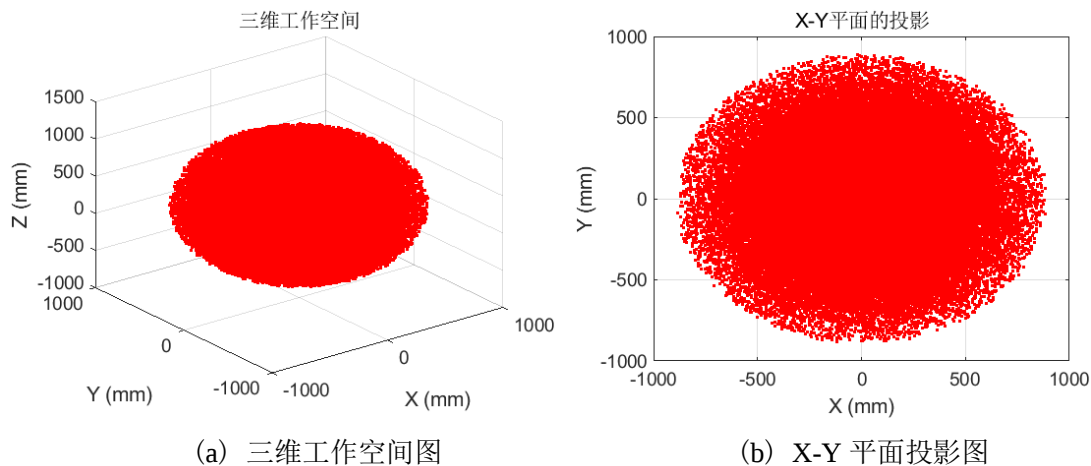


图 2-8 操作端的工作空间图

(2) 执行端工作空间的求取

执行端工作空间的求取同操作端一样，最终得到的点云分布如图 2-9 所示。



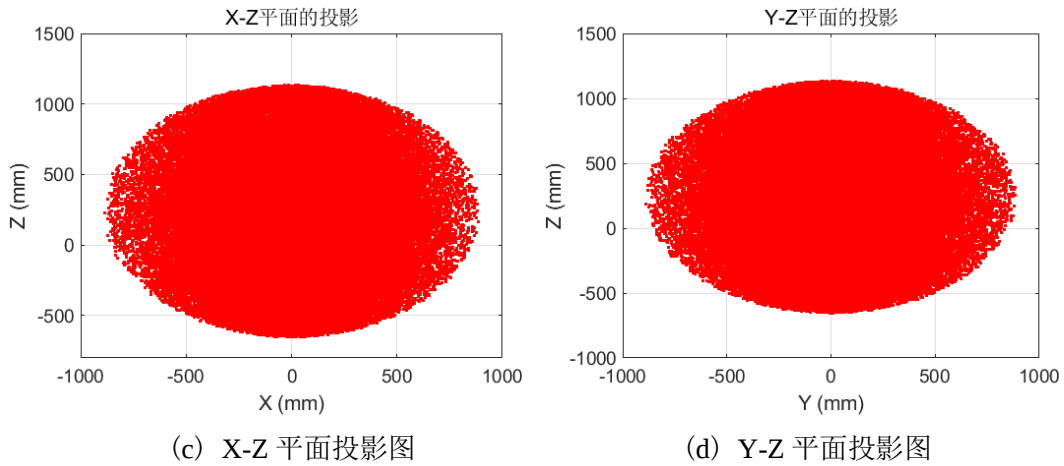


图 2-9 执行端的工作空间图

2.4.3 空间映射的实现

根据上述求得的工作空间图，列出其工作空间范围表，如表 2-5 所示。

表 2-5 操作端与执行端的工作空间

坐标轴	操作端的最大最小范围/mm	执行端的最大最小范围/mm
X	(-0.58,262.32)	(-884.56,885.97)
Y	(-215.25,216.01)	(-883.89,887.05)
Z	(45.92,442.36)	(-556.38,1131.52)

设 S_t 为操作端的末端位置向量， L_r 为执行端的末端目标位置向量（即映射后的位置向量），空间映射系数为 K ，则 S_t 与 L_r 之间存在如下关系：

$$S_t = KL_r + B_i \quad (2-39)$$

其中 B_i 代表一个常量，它确保了经过比例调整后的操作端工作空间的原点与执行端工作空间的原点位置相匹配。则可得：

$$K = \frac{S_{tMax} - S_{tMin}}{L_{rMax} - L_{rMin}} \quad (2-40)$$

$$B_i = \frac{S_{tMax} + S_{tMin} - K(L_{rMax} + L_{rMin})}{2} \quad (2-41)$$

式中 S_{tMax} 、 S_{tMin} 表示执行端工作空间的最大最小值； L_{rMax} 、 L_{rMin} 表示操作端工作空间的最高、最低值。由表 2-5 计算可得，比例系数 K 和常量 B_i 在 X 、 Y 、 Z 三个方向上的分量分别为式（2-42）和（2-43）所示：

$$\begin{cases} K_x = \frac{S_{txMax} - S_{txMin}}{L_{rxMax} - L_{rxMin}} \\ K_y = \frac{S_{tyMax} - S_{tyMin}}{L_{ryMax} - L_{ryMin}} \\ K_z = \frac{S_{tzMax} - S_{tzMin}}{L_{rzMax} - L_{rzMin}} \end{cases} \quad (2-42)$$

$$\begin{cases} B_{ix} = \frac{S_{txMax} + S_{txMin} - K_x(L_{rxMax} + L_{rxMin})}{2} \\ B_{iy} = \frac{S_{tyMax} + S_{tyMin} - K_y(L_{ryMax} + L_{ryMin})}{2} \\ B_{iz} = \frac{S_{tzMax} + S_{tzMin} - K_z(L_{rzMax} + L_{rzMin})}{2} \end{cases} \quad (2-43)$$

根据表 2-5 计算可得 K_x 、 K_y 、 K_z 分别为 6.73、4.11、4.26， B_x 、 B_y 、 B_z 分别为 -883.40、0.02、-752.47。如图 2-10 所示，映射后的工作空间点云图呈现出高度一致的匹配效果，二者几乎重合，充分满足了操作端对执行端的精确控制要求。

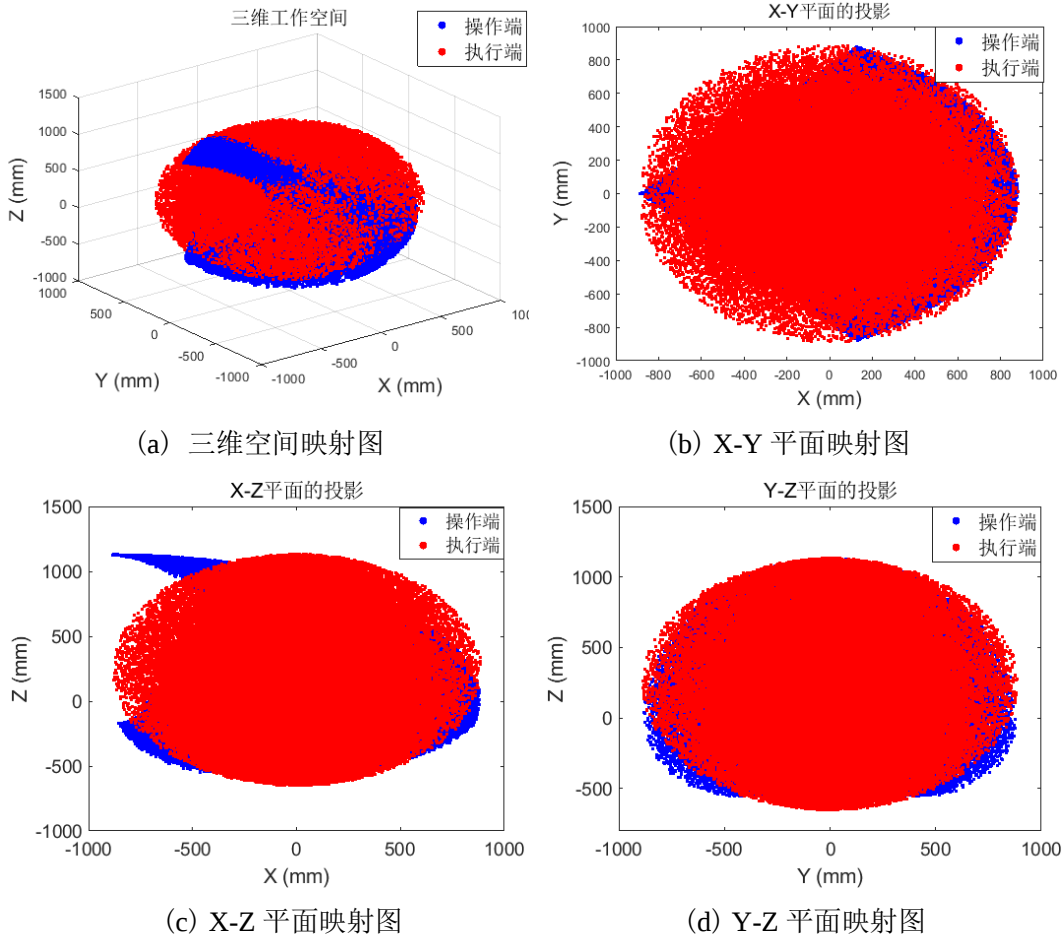


图 2-10 映射后工作空间点云对比图

2.5 本章小结

本章系统地研究了操作端与执行端的运动学建模及空间映射问题。首先，完成了操作端和执行端的选型。随后，分别对操作端和执行端进行了运动学分析，涵盖了连杆坐标系的建立、正运动学仿真与计算，以及执行端的运动学正解和逆解。在空间映射方面，采用了基于笛卡尔空间的映射方法，详细分析了操作端与执行端工作空间的差异性，并设计了两端之间的空间映射，为操作端与执行端的虚实控制奠定了基础。

第3章 基于虚拟环境的打磨机器人交互系统设计

本章介绍了基于虚拟环境的打磨机器人交互系统设计的总体方案。首先，对交互系统的总体方案进行了详细分析，包括系统的总体框架、功能设计以及平台的选型。其次，在 Unity3D 中建立虚拟打磨场景，并对 UI 界面进行设计。随后，通过动态生成工件网格模型，结合球形打磨头变形算法实现了打磨过程材料的实时去除效果。

3.1 交互系统总体方案

在执行打磨任务时，机器人的应用面临着一系列挑战，首先，由于打磨本身的技术要求较高，操作者需要具备一定的专业技能才能熟练操控机器人；其次，打磨过程中产生的粉尘对操作者的健康构成威胁，需要采取相应的防护措施；最后，由于打磨作业可能涉及高速旋转的磨具和锋利的工件，存在一定的安全隐患。鉴于以上考虑，需要开发一个基于虚拟环境的打磨机器人交互系统，该系统旨在通过模拟真实打磨环境，提供一个安全、高效的训练平台。

3.1.1 交互系统总体框架

交互系统总体框架如图 3-1 所示。该系统利用 SolidWorks 创建打磨机器人三维模型，并通过 Blender 调整模型的旋转轴和比例大小，确保模型在虚拟环境中的准确性和可操作性。操作端设备通过 USB 接口与系统连接，通过映射算法脚本，实现对打磨机器人的精确控制。通过调用 HDAPI 函数用于实现力觉交互，使用户在操作过程中能够感受到真实的力反馈，增强了沉浸感。通过为打磨机器人和工件添加材质、贴图、刚体和碰撞体，然后动态创建工件的网格模型，结合球形打磨头变形算法，进一步提升了虚拟环境的真实感。基于 TCP/IP 协议的 Socket 网络通信技术支持远程数据传输与 VR 遥操作，用户可通过 VR 头盔控制虚拟环境中的打磨机器人，也可以直接远程控制实体打磨机器人实现高效的远程协作。最后，系统配备了直观的 UI 人机交互界面，优化了用户的操作体验与系统的易用性。

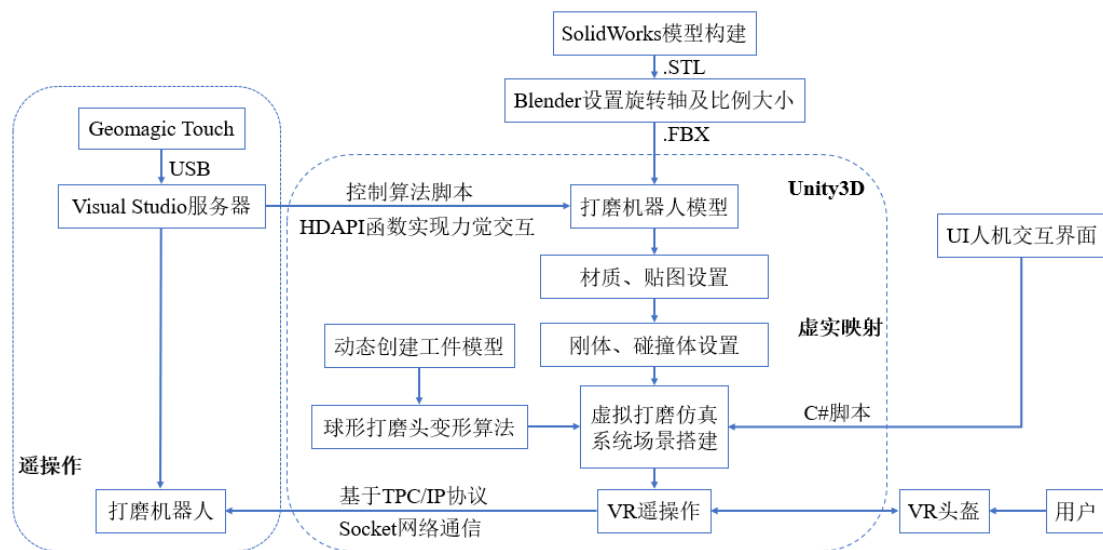


图 3-1 交互系统的总体框架

3.1.2 交互系统的功能设计

基于交互系统的框架设计，本节将其划分为虚实映射与遥操作两大模块，具体如图 3-2 所示。

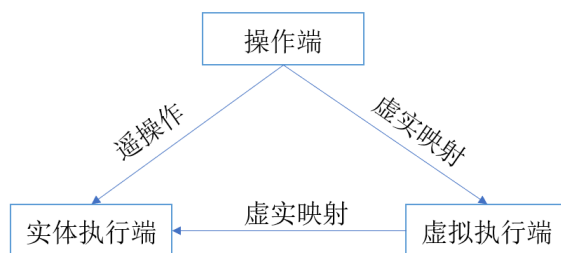


图 3-2 交互系统的架构

(1) 虚实映射功能

虚实映射系统由网络通信、力反馈、打磨功能与 VR 视觉四大模块构成，实现操作端、虚拟端与实体端的同步控制。网络通信采用三层架构：上层 C#服务器基于 Visual Studio 开发，负责实时采集操作端关节角度，通过正运动学计算位姿信息，结合空间映射算法生成虚拟执行端控制指令，经 Socket 协议传输至中间层；中间层 Unity3D 客户端作为虚拟执行端，接收服务器数据驱动虚拟机器人运动，并将末端位姿实时回传服务器；下层机器人控制器接收服务器指令驱动实体机器人。打磨功能模块通过动态生成工件网格模型，结合球形打磨头变形算法实现材料去除的实时渲染，并与力反馈模块联动。力反馈模块借助 OpenHaptics SDK，实时解析虚拟环境中工具-工件交互数据，通过物理引擎分析接触状态，计

算碰撞力学效应并转化为触觉反馈指令，驱动设备输出精确力觉信号，使操作者感知打磨过程中的力学特性。VR 交互模块是指在虚拟现实系统平台中与 VR 交互，能够更加逼真的看见虚拟执行端机器人的运动，让操作变得更加真实。

（2）遥操作功能

遥操作功能主要由网络通信模块和力觉交互模块协同实现。在网络通信方面，系统采用基于 Socket 的 C/S 架构：首先，通过 Visual Studio 搭建的服务器实时采集操作端的关节数据；随后，系统对这些数据进行正运动学解算和空间映射处理，得到执行端的位姿信息；最后，通过 TCP Socket 协议将处理后的位姿数据传输至实体执行端控制器，从而实现对执行端的精确控制。力觉交互模块则是实体执行端与工件接触时，将执行末端端六维力传感器采集的力数据，实时反馈至操作端，实现遥操作过程中的力觉交互，从而提升操作的精确性和沉浸感。

3.1.3 交互平台的选型

（1）MATLAB 平台

MATLAB 以其高效的数值计算能力和机器人工具箱（Robotics Toolbox），在机器人建模与仿真中展现了强大的功能，能够显著降低代码复杂度，成为理论研究的首选工具。然而，MATLAB 在实时渲染和交互性能方面表现较弱，难以支持打磨机器人交互系统对高帧率渲染和流畅交互的需求。此外，其界面设计能力有限，通常需要依赖额外的工具箱或第三方工具，难以实现直观、美观且易用的用户界面。

（2）Web Robots 平台

Web Robots 是一款开源的机器人仿真平台，支持多种机器人模型和传感器仿真。它提供跨平台支持（Windows、macOS、Linux）和多语言编程接口（如 Python、C++），内置丰富的物理引擎和图形化界面，适合教育、研究和工业应用。然而，Web Robots 在图形渲染、实时性和扩展性方面的不足，限制了其在高精度仿真和复杂系统中的应用。

（3）Unity3D 平台

Unity3D 以其强大的三维可视化、物理引擎和交互设计功能，成为机器人仿真系统的理想开发平台。它支持高精度模型渲染和环境模拟，内置碰撞检测功能，为路径规划和运动控制提供可靠验证。同时，Unity3D 具备跨平台能力和丰富的

资源库，显著降低开发难度，满足从理论研究到实际应用的多层次需求，是机器人交互系统开发的理想工具。图 3-3 呈现了 Unity3D 的主要界面结构，包含工具栏、菜单栏以及多个操作面板。这些面板分别用于场景展示、游戏预览、层级管理、项目资源查看和属性调整，具体分为 Scene、Game、Hierarchy、Project 和 Inspector 五个功能区域。

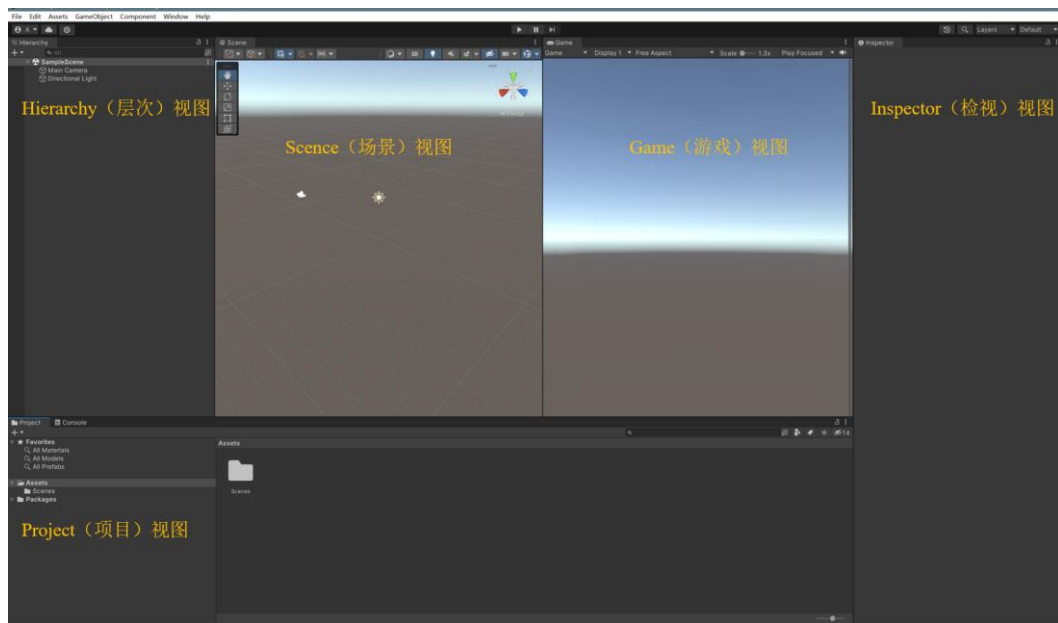


图 3-3 Unity3D 的主界面

3.2 虚拟场景建模

在基于虚拟环境的打磨机器人交互系统中，虚拟场景建模是一个至关重要的环节。它不仅为操作者提供了一个逼真的交互环境，而且对于提高机器人操作的精确性和安全性具有重要意义。本节将详细介绍虚拟场景建模的过程，包括场景的设计、建模技术的选择以及模型的优化。具体的机器人建模流程见图 3-4。

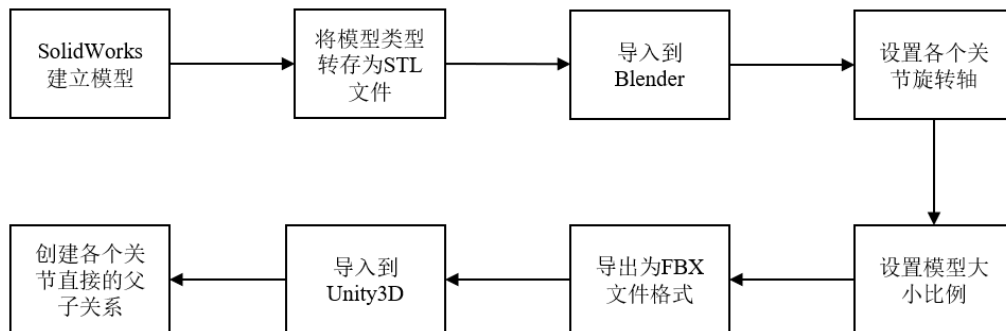


图 3-4 机器人建模流程

（1）模型优化

在 SolidWorks 中，机器人模型通常包含众多细微的组件，直接导入这些未经简化的模型将带来繁重的处理负担，并可能在模型导入导出过程中导致软件运行缓慢或卡顿。因此，在将模型导入到 Unity3D 之前，需在 SolidWorks 对模型进行简化处理，确保其主体结构不变的同时，删除冗余的非核心组件。

（2）模型的调整

将 SolidWorks 创建的机器人模型导入 Unity3D 时，需要通过 Blender 进行调整，主要是因为两者在坐标系、单位、模型格式、以及模型优化等方面存在差异，Blender 可以作为中间工具来解决这些问题，确保模型在 Unity3D 中的正确显示和高效运行，Blender 对模型的整个处理流程如图 3-5 所示。

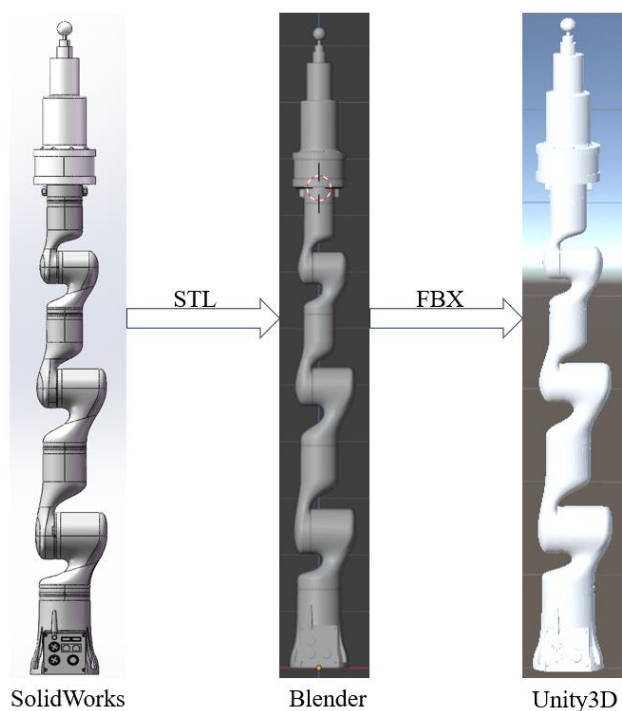


图 3-5 Blender 处理模型流程图

在 Unity3D 环境下，物体的旋转是基于其自身坐标轴进行的。因此，针对机器人模型，需在 Blender 中对其各关节的坐标轴进行调整，确保每个坐标轴位于对应关节的旋转中心位置，同时将机器人模型放置在坐标原点处。具体调整方式如图 3-6 所示。

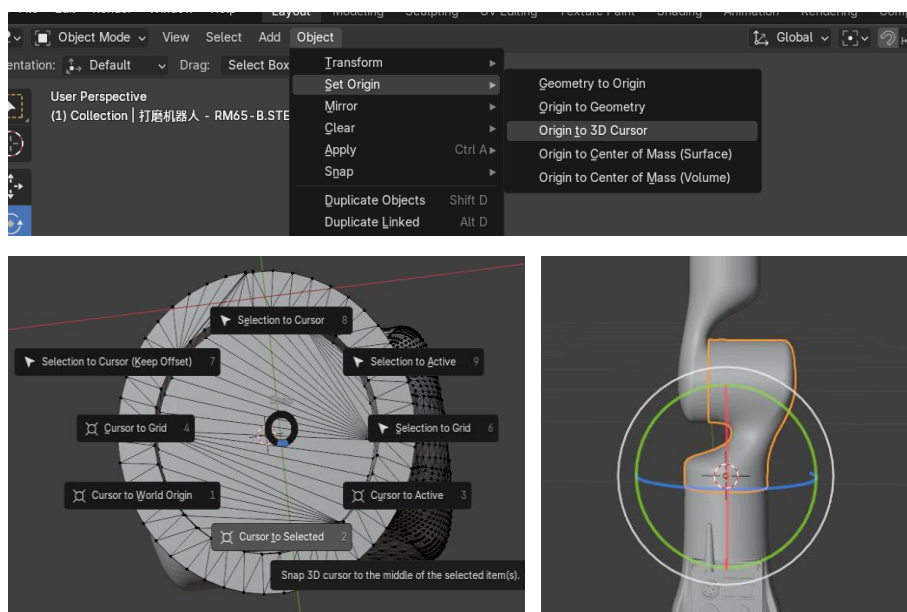


图 3-6 关节坐标轴的调整

虽然 SolidWorks 和 Blender 均遵循右手直角坐标系,但两者在坐标轴的方向上存在差异。故将模型导入到 Blender 时,需要将坐标轴进行转换。具体的导入设置见图 3-7。此外,由于 SolidWorks 默认使用毫米作为单位,而 Blender 和 Unity3D 则默认使用米作为单位,为确保机器人模型在 Unity3D 中的尺寸与 SolidWorks 中建模时保持一致,需在导入模型时将“缩放”因子设置为 0.001。完成上述调整后,转换为 Unity3D 能接受的 FBX 格式,并拖入场景中,将模型置于坐标原点,以实现精确的控制。

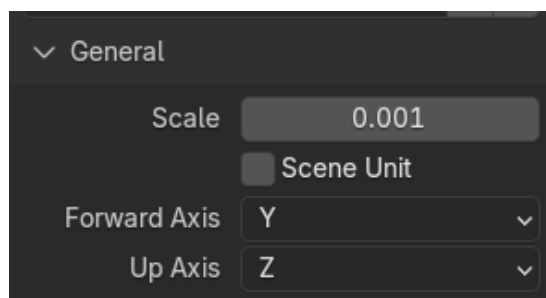


图 3-7 导入设置

(3) 模型树的建立

Unity3D 中创建父子关系是为了正确表达机器人的层级结构,实现层级化运动传递,简化运动学计算。通过父子关系,父关节的运动能够自动传递到子关节,确保模型结构的逻辑一致性,同时便于动画和脚本控制,使机器人在仿真中的行为与实际物理系统保持一致,这是实现机器人精确控制和仿真的关键步骤。在

Unity3D 的 Hierarchy 视图中，通过拖动关节构建模型树，建立各关节之间的父子关系，如图 3-8 所示。

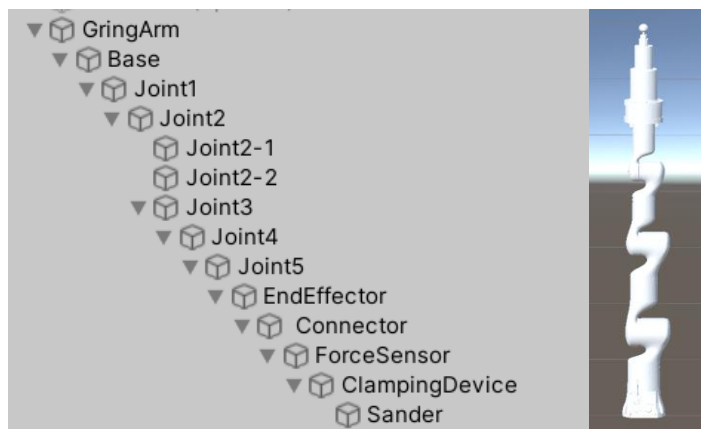


图 3-8 机器人模型树结构

3.3 UI 控制界面设计

Unity3D 中的用户界面（UI）设计主要基于 UGUI 组件，这些组件分为可视化组件和交互式组件^[54]。可视化组件如 Text、Panel、Mask 和 Image，用于显示信息和构建视觉结构。交互式组件包括 Button、Scrollbar、Toggle、Slider、Input Field 和 Scroll Rect，使用户能够通过点击、滑动或输入与界面进行交互。如图 3-9 所示，UI 界面通过合理布局这些组件，实现了功能性与美观性的平衡。可视化组件确保了信息的清晰呈现，而交互式组件则增强了用户的操作体验。这种设计提升了界面的直观性和易用性，优化了用户的操作效率，符合现代软件界面设计的高标准要求。通过科学的组件配置与布局，界面在交互性和信息传达方面表现出色，满足了用户对高效操作和视觉体验的需求。

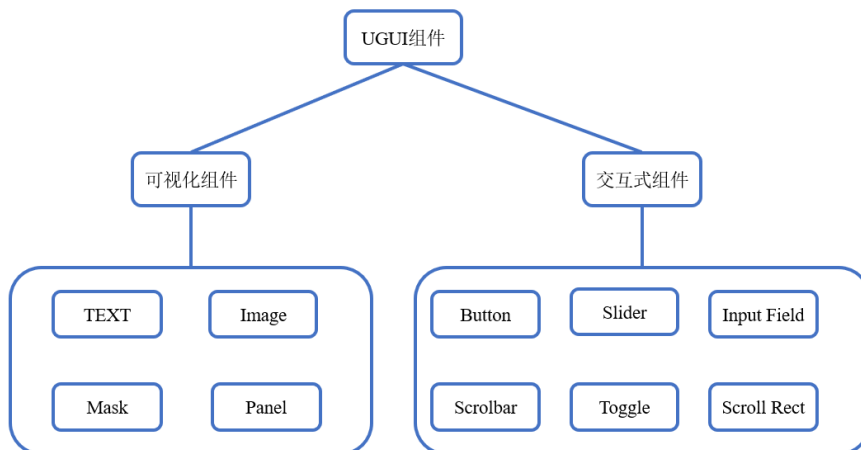


图 3-9 UGUI 常用控件及分类

本系统针对交互系统进行了组件优化设计,重点实现了操作端设备与虚拟环境之间的实时数据交互。系统包含两个 **Button** 组件,分别用于管理操作端设备与平台的通信连接和断开操作,从而确保数据传输的稳定性和可控性。此外,系统设计了 **Joint** 和 **Position** 两个控制区域,分别用于实现角度和的位置精确控制。其用户界面设计如图 3-10 所示。

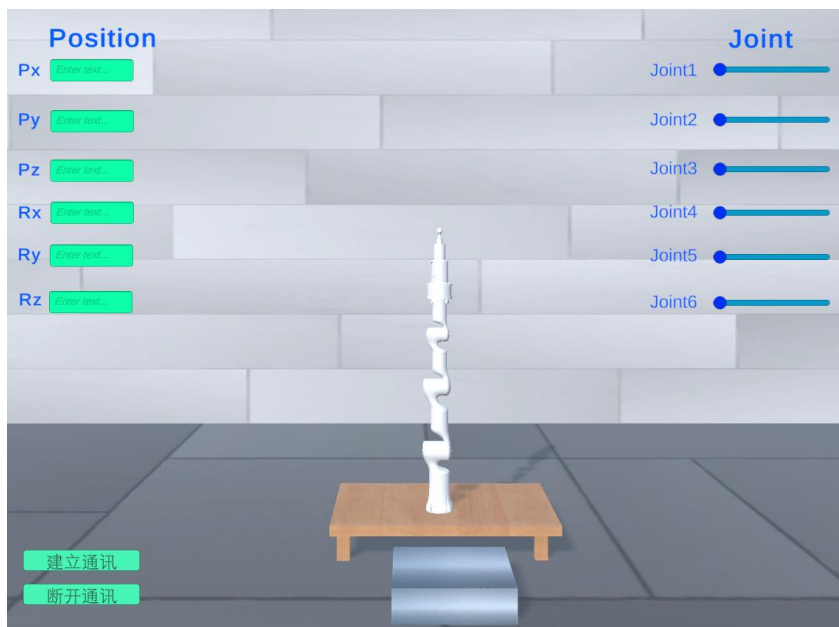


图 3-10 UI 界面

在 **Joint** 控制区域方面,系统采用 **Slider-Text** 联动机制:为机器人的六个自由度分别配置了 **Slider** 调节组件和 **Text** 显示组件,其中 **Slider** 组件用于调节各关节角度,**Text** 组件则实时显示对应的角度数值。同时,系统通过独立的 **Text** 组件实时显示机器人末端执行器坐标系相对于 **Unity3D** 世界坐标系的位姿变化。该位姿显示组件与 **Joint** 控制区的角度控制组件建立了双向数据联动机制:当用户通过 **Slider** 组件调整关节角度时,系统将自动进行正向运动学计算,驱动虚拟机器人执行相应运动,并将计算得到的末端位姿实时更新在位姿显示组件中。

在 **Position** 控制区域,系统提供了基于 **Input Field** 组件的末端位姿直接输入功能。当用户输入目标位姿数据时,系统将通过逆运动学计算自动求解各关节角度,驱动虚拟机器人运动,并将计算结果实时更新至 **Joint** 控制区对应的 **Text** 显示组件。此外,**Joint** 控制区与 **Position** 控制区域之间建立了完善的数据同步机制:在 **Joint** 控制区进行关节角度调整时,**Position** 控制区域将同步更新末端位姿

信息；反之，当在 **Position** 控制区域输入末端位姿时，**Joint** 控制区也将同步更新各关节角度值。这种双向联动机制显著提升了系统的交互效率和操作直观性。

3.4 实时去除效果渲染

在 Unity3D 中，**Mesh Filter** 组件包含网格（**Mesh**），用于描述模型的几何形状，而 **Mesh Renderer** 组件则包含模型的材质，用于定义模型表面的外观属性。网格是描述模型形状的数据集合，主要由顶点（**Vertices**）和三角面（**Triangles**）构成。其中，顶点位置坐标数组和三角面索引数组是定义网格几何形状的两个核心组成部分。顶点位置坐标数组存储了每个顶点在三维空间中的 X 、 Y 、 Z 坐标值，而三角面索引数组则是一个整数数组，用于描述顶点如何连接以形成三角形面片。通过指定顶点的索引，可以构建三角形面片，进而组合成模型的完整表面。在三角面索引数组中，顶点组合的顺序称为绕序（**Winding Order**）。Unity3D 通过绕序确定三角形面的朝向，识别其属于正面还是背面，进而决定是否需要渲染或剔除。默认情况下，Unity3D 采用顺时针绕序，即当围绕三角面的顶点按顺时针方向遍历时，该三角面被视为正面，反之则为背面。这里的“面”特指三角面片，而“绕序”则是指构成三角面的三个顶点的索引连接顺序。

3.4.1 工件网格模型的建立

工件模型共有四个属性：模型的长 a （ X 方向的长度）、宽 c （ Z 方向的长度）、高 b （ Y 方向的长度）以及网格细分数。顶点数组构建时，首先基于几何中心创建上表面：沿 X 、 Y 、 Z 方向将模型均匀细分为 N 份，每条边生成 $N+1$ 个顶点，共形成 $(N+1) \times (N+1)$ 个顶点。将工件的长分为 i 份，宽分为 j 份，由于网格细分数是 N ，那么 i 的取值范围是 $[-N/2, N/2]$ ， j 的取值范围是 $[-N/2, N/2]$ 。故可得上表面中任意一点在工件坐标系下的位置为：

$$\begin{cases} x = \frac{ia}{N} \\ y = \frac{c}{2} \\ z = \frac{jb}{N} \end{cases} \quad (3-1)$$

构建前后左右四个侧面的顶点数组，每个侧面的顶点数是 $N+1$ ，所以侧面的顶点坐标数组共有 $4(N+1)$ ，那么前后左右表面任意一点的位置为：

$$\left\{ \begin{array}{l} x = \frac{ia}{N} \\ y = -\frac{c}{2} \\ z = -\frac{b}{2} \end{array} \right. \quad \left\{ \begin{array}{l} x = \frac{ia}{N} \\ y = -\frac{c}{2} \\ z = \frac{b}{2} \end{array} \right. \quad \left\{ \begin{array}{l} x = -\frac{a}{2} \\ y = -\frac{c}{2} \\ z = \frac{ib}{N} \end{array} \right. \quad \left\{ \begin{array}{l} x = \frac{a}{2} \\ y = -\frac{c}{2} \\ z = \frac{ib}{N} \end{array} \right. \quad (3-2)$$

底面由四个顶点构成，坐标分别为 $(-a/2, -b/2, -c/2)$ 、 $(a/2, -b/2, -c/2)$ 、 $(a/2, -b/2, c/2)$ 、 $(-a/2, -b/2, c/2)$ 。由于索引数组存储的是顶点的下标，且工件模型的长和宽又被分为 i 和 j 份，则 i 和 j 的取值范围均为 $[0, N]$ 。逐顶点构建三角面，上表面共有 $(N+1) \times (N+1)$ 个顶点，但其中有两条边的顶点并不参与构建三角面，因此实际参与构建三角面的顶点数量为 $N \times N$ 个。由于每个顶点会参与两个三角面的构建，且每个三角面由三个顶点构成，所以上表面的索引数组中元素的总数为 $6N \times N$ 。然后逐顶点设置构建两个三角面的元素，则可得两个三角面的数组 A_1 和 B_1 分别为：

$$\left\{ \begin{array}{l} A_1 = [i(N+1) + j, i(N+1) + j + 1, (i+1)(N+1) + j + 1] \\ B_1 = [i(N+1) + j + 1, (i+1)(N+1) + j + 1, (i+1)(N+1) + j] \end{array} \right. \quad (3-3)$$

设置前后左右四个侧面的索引数组，由于网格细分数为 N 份，每个网格由两个三角面组成，故前后左右表面的两个三角面中数组分别为 A_2 、 B_2 、 A_3 、 B_3 、 A_4 、 B_4 、 A_5 、 B_5 ：

$$\left\{ \begin{array}{l} A_2 = [i(N+1), (N+1)(N+1) + i + 1, (N+1)(N+1) + i] \\ B_2 = [i(N+1), (i+1)(N+1), (N+1)(N+1) + i + 1] \end{array} \right. \quad (3-4)$$

$$\left\{ \begin{array}{l} A_3 = [i(N+1) + N, 2(N+1)(N+1) + i, 2(N+1)(N+1) + i + 1] \\ B_3 = [i(N+1) + N, 2(N+1)(N+1) + i + 1, (i+1)(N+1) + N] \end{array} \right. \quad (3-5)$$

$$\left\{ \begin{array}{l} A_4 = [i, (2N+1)(N+1)(N+1) + i, (2N+1)(N+1)(N+1) + i + 1] \\ B_4 = [i, (2N+1)(N+1)(N+1) + i + 1, i + 1] \end{array} \right. \quad (3-6)$$

$$\left\{ \begin{array}{l} A_5 = [i(N+1) + N, (N+2)(N+1) + i, (N+2)(N+1) + i + 1] \\ B_5 = [i(N+1) + N + 1, (N+2)(N+1) + i + 1, (i+1)(N+1) + N] \end{array} \right. \quad (3-7)$$

由于底面由四个顶点组成，那么底面只有两个三角面，构成三角面的索引数组 $A_6 = (0, 1, 3)$ ， $B_6 = (0, 3, 2)$ 。最后给工件添加 MeshFilter 和 MeshRenderer 组件，并且将网格赋给 MeshFilter 组件，成功构建工件的网格模型。

3.4.2 球形打磨头变形算法

球形打磨头变形算法的核心是通过模拟球形工具与工件网格表面的交互，动态调整网格顶点的位置，使其贴合球形打磨头的表面，流程如图 3-11 所示。

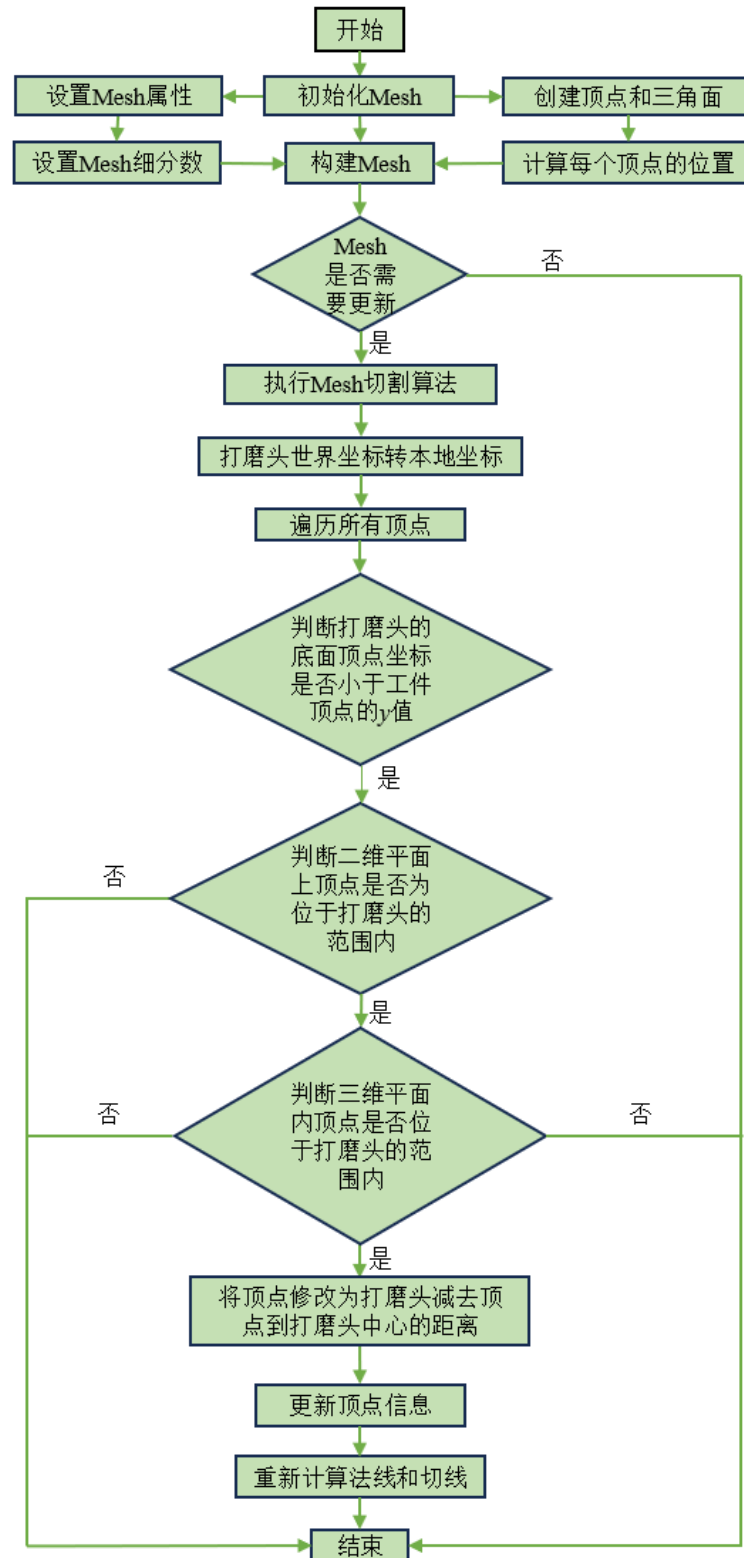


图 3-11 球形打磨头变形算法

(1) 首先, 算法需要初始化网格 (Mesh) 的属性, 包括顶点数组、三角面数组以及网格的细分参数 (如细分数)。这些属性决定了网格的分辨率和复杂度。根据设置的属性, 算法构建网格的初始形状。通常, 网格的初始形状是一个规则的几何体 (如立方体), 通过细分参数将其划分为多个小面片。根据网格的细分参数, 计算每个顶点的初始位置。这些顶点位置决定了网格的初始形状。

(2) 在每次交互中, 算法首先判断网格是否需要更新。如果需要更新 (如打磨头的位置或半径发生了变化), 则进入变形算法。变形算法是变形的核心部分, 负责根据打磨头的位置和半径, 调整网格顶点的位置。

(3) 为了简化计算, 算法将打磨头的世界坐标转换为网格的本地坐标。这一步通过坐标变换实现, 确保所有计算都在同一坐标系下进行。算法遍历网格的所有顶点, 检查每个顶点是否位于打磨头的范围内。这一步骤是变形算法的核心, 决定了哪些顶点需要被调整。

(4) 球形打磨头的固定半径为 r , 其几何特性由球面方程定义:

$$(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 = r^2 \quad (3-8)$$

其中, (x_0, y_0, z_0) 是球形打磨头的中心坐标, (x, y, z) 是球面上的任意一点。球形打磨头的半径决定了其与网格交互的范围。半径越大, 打磨头影响的区域越广; 半径越小, 打磨头的影响区域越集中。

(5) 算法在二维平面上判断顶点是否位于打磨头的范围内。通过计算顶点与打磨头中心的距离:

$$d_{2D} = \sqrt{(x_1 - x_0)^2 - (z_1 - z_0)^2} \quad (3-9)$$

如果 $d_{2D} < r$, 则顶点在二维平面上位于打磨头的范围内。

如果顶点在二维平面上位于打磨头的范围内, 算法进一步在三维空间中判断顶点是否位于打磨头的范围内。通过计算顶点与打磨头中心的欧几里得距离:

$$d_{3D} = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2 + (z_1 - z_0)^2} \quad (3-10)$$

如果 $d_{3D} < r$, 则该顶点位于球形打磨头的范围内, 需要进行变形处理; 否则, 顶点保持不变。

(6) 对于位于打磨头范围内的顶点, 算法根据打磨头的几何特性调整其坐标。具体来说, 算法将顶点的 y 坐标修改为打磨头表面对应的值:

$$y_1 = y_0 - \sqrt{r^2 - (x_1 - x_0)^2 - (z_1 - z_0)^2} \quad (3-11)$$

这种操作模拟了打磨头对网格表面的“凹陷”效果，使网格表面贴合球形打磨头的形状。

(7) 在实时交互场景中，球形打磨头的位置和半径可以动态调整。每次打磨头移动或半径变化时，算法会重新检测网格顶点并进行变形处理。变形后的顶点坐标会更新到网格中，从而实现网格的动态变形。为了确保光照和渲染效果正确，变形后的网格需要重新计算法线。通过实时刷新网格，用户可以即时看到打磨头对网格表面的材料去除效果，从而实现实时的打磨材料去除效果。

3.5 本章小结

本章详细阐述了打磨机器人交互系统的总体设计、方案确定、虚拟场景建模、人机交互及打磨功能开发等内容，采用 Unity3D 引擎的逼真虚拟环境和高效开发优势，构建了虚拟仿真平台，对打磨机器人模型进行优化、调整和导入，并通过特定算法实现虚拟场景中的打磨功能，同时设计了便捷的 UI 控制界面和交互式虚拟场景漫游功能，以提高操作者对机器人的操控体验和沉浸感，为后续系统的实现与应用提供了全面的技术支撑和理论依据。

第4章 虚实交互与控制技术研究

本章通过对层次包围盒算法、碰撞体与刚体的检测以及力反馈交互技术进行分析,探讨了 VR 交互技术,包括头戴式立体显示器的使用和 HTC VIVE Pro2 系统与 Unity3D 的集成,研究了空间坐标转换问题并验证了机器人运动学算法的准确性,确保操作者的动作和意图能够准确地传输到远程机器人,同时反馈实时状态信息给操作者,为操作者提供了一个沉浸式的交互环境。

4.1 碰撞检测

碰撞检测技术广泛应用于多个领域,包括虚拟仿真、机器人运动规划、电子游戏设计以及图形处理等^[55]。以虚拟仿真为例,当打磨机器人执行任务时,必须通过碰撞模拟来预测可能的接触情况,以防止机器人在操作过程中穿透场景中的其他物体,导致不真实的效果。在虚拟仿真中,物体通常由无数的点、线和多边形组合而成,因此,当这些基本元素相互交叉时,就可以判定物体之间发生了接触。然而,如果采用逐一检查每个点、线和多边形的方法来进行碰撞检测,将会极大地消耗计算机的 CPU 资源。为了解决这一问题,本系统采用了层次包围盒算法来优化碰撞检测的效率。通过将 Unity3D 的碰撞检测系统与操作端设备结合,系统能够为操作者提供即时的力反馈,从而增强操作的沉浸感。

4.1.1 层次包围盒算法

层次包围盒是一种高效的空间数据结构,用于加速碰撞检测和空间查询。其原理是通过构建嵌套的包围盒来近似表示复杂几何对象,形成层次化的树形结构。构建过程中,首先将所有对象用一个较大的包围盒完全包围,作为根节点;然后根据预定义的分割策略(如中点分割或空间划分等),递归地将包围盒划分为子包围盒,直至每个子包围盒内仅包含少量对象或满足停止条件。在查询时,通过逐层判断查询对象是否与当前节点的包围盒相交,若不相交则跳过该节点及其子节点,从而显著减少不必要的计算,提高效率。层次包围盒算法依据几何形态的差异^[56],主要分为三种类型:首先是包围球算法,利用球体作为基本包围单元;其次是轴对齐矩形包围盒 AABB 算法,其边界与坐标轴平行;最后是方向

矢量包围盒 OBB 算法，能够根据物体方向动态调整包围盒的朝向，如图 4-1 所示。

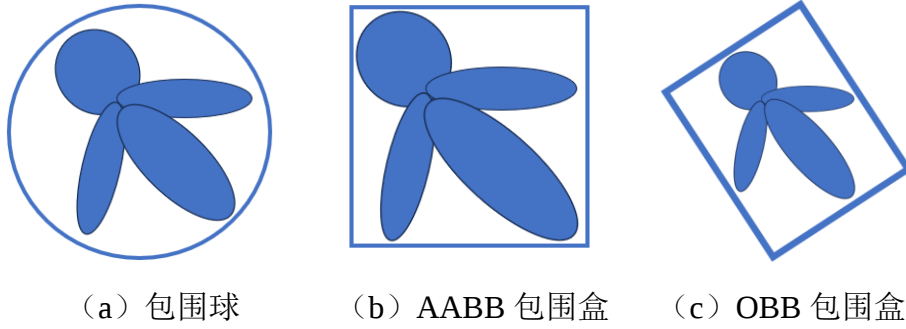


图 4-1 层次包围盒算法

包围球算法是一种用于计算一组点的最小包围球的算法，其目的是找到一个半径最小的球，使得所有给定点都位于该球的内部或边界上^[57]。其球体检测范围为：

$$R_B = \{(x, y, z) | (x - o_x)^2 + (y - o_y)^2 + (z - o_z)^2 \leq r_B^2\} \quad (4-1)$$

其中， (o_x, o_y, o_z) 表示球心的坐标， r_B 表示球的半径。该公式定义了一个以 (o_x, o_y, o_z) 为球心，半径为 r_B 的球体。如果两个物体的包围球不相交，即它们中心点之间的距离大于它们半径之和，那么这两个物体不可能发生碰撞，从而可以快速排除。这种方法计算速度快，适合初步筛选大量物体，但包围球的紧密性较差，对于非球形物体，包围球可能会包含大量无用的空间，导致碰撞检测的精度下降，容易出现误报。

AABB 包围盒算法定义了一个轴对齐的包围盒，其检测范围为：

$$R_B = \{(x, y, z) | a_x \leq x \leq b_x, a_y \leq y \leq b_y, a_z \leq z \leq b_z\} \quad (4-2)$$

其中， a_x 、 a_y 、 a_z 表示包围盒在各坐标轴上的最小值，而 b_x 、 b_y 、 b_z 表示最大值。通过计算两个物体的 AABB 是否在空间上重叠，可以快速判断这两个物体是否可能发生碰撞。如果两个 AABB 不重叠，则对应的物体不可能碰撞，从而避免了更复杂的精确碰撞检测。这种方法因其高效性而被广泛应用于需要处理大量物体的场景中。AABB 包围盒算法因其计算简单和执行速度快而受到青睐，适用于初步的碰撞检测阶段。不过，它的一个主要缺点是紧密性差，这意味着包围盒可能无法紧密贴合物体的实际形状，特别是在物体形状复杂或不规则时，导

致包围盒的体积大于物体本身，增加了不必要的检测计算，降低了碰撞检测的精确度^[58]。

OBB 包围盒算法^[59]通过定义一个中心点 C 和三个基向量 $\vec{v}_1$ 、 $\vec{v}_2$ 、 $\vec{v}_3$ 来定义一个定向包围盒，其空间有效范围：

$$R_B = \{C + ar_1\vec{v}_1 + br_2\vec{v}_2 + cr_3\vec{v}_3 \mid a, b, c \in (-1, 1)\} \quad (4-3)$$

其中 r_1 、 r_2 、 r_3 是沿基向量方向的半轴长度。OBB 的优点是能更紧密地贴合物体形状，提高碰撞检测的准确性，但缺点是计算和相交测试比 AABB 更复杂，需要更多的计算资源。

4.1.2 碰撞体与刚体

Unity3D 引擎利用 Rigidbody 和 Collider 组件来模拟物体间的碰撞行为。Rigidbody 组件赋予物体物理属性，使其能够响应碰撞和力的作用，同时也能触发碰撞事件。Collider 组件则定义物体的碰撞体积，Unity3D 提供了多种类型的碰撞体，如 Box、Sphere、Capsule 等，以适应不同形状的物体需求，见表 4-1。

表 4-1 Unity3D 中的碰撞体类型

碰撞体类型	中文名称	形状特点	包围盒类型
Box Collider	盒碰撞体	长方体	AABB
Sphere Collider	球碰撞体	球体	包围球
Capsule Collider	胶囊碰撞体	圆柱体	类似包围球
Mesh Collider	网格碰撞体	由物体网格形状确定	OBB
Terrain Collider	地形碰撞体	针对于地形	特殊包围盒

要实现物体间的碰撞检测，需要确保至少有一个物体具有刚体组件，并且所有物体都装备了碰撞体组件。如果被动物体也装备了刚体组件，应避免勾选 Is Kinematic 属性，以确保其能正确响应碰撞。Unity3D 中的刚体分为静态碰撞体、刚体碰撞体和运动学刚体碰撞体。静态碰撞体常用于地形、障碍物等不移动的物体，物理引擎会优化其性能。刚体碰撞体的物理状态由物理引擎持续模拟计算，会对碰撞及受力做出反应。运动学碰撞体，不会对碰撞和力做出反应，不受物理系统影响，但仍会对其他刚体产生物理影响，如撞击其他刚体。Unity3D 中的虚拟打磨机器人通过这些组件的合理配置，可以模拟出精确的物理碰撞效果。碰撞检测函数使用方法如表 4-2 所示。

表 4-2 碰撞检测方法

碰撞检测方法	函数
碰撞开始	OnCollisionEnter(Collision collisionInfo)
持续碰撞	OnCollisionStay(Collision collisionInfo)
结束碰撞	OnCollisionExit(Collision collisionInfo)

根据虚拟执行端的结构形状，为其关节添加 Box Collider 组件，为打磨头添加 Sphere Collider 组件，以实现打磨过程高效的碰撞检测，如图 4-2 所示。

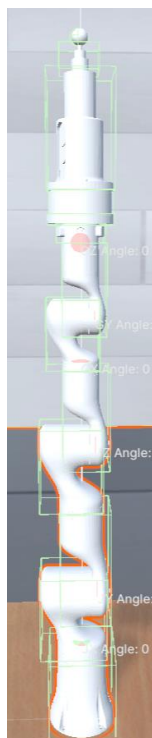


图 4-2 执行端碰撞体包裹图

4.2 虚拟现实交互技术

4.2.1 力觉交互

力觉交互技术是一种基于触觉反馈的人机交互方式，其核心机制是通过操作端设备将虚拟环境中的交互力实时反馈给操作者。当操作者操纵设备与虚拟物体发生接触时，系统会精确计算碰撞点信息，并将相应的力反馈传递给操作者，使其能够感知虚拟环境中机器人末端执行器与目标物体之间的交互力，从而显著提升操作的沉浸感和真实感。

在 Unity3D 引擎中，虚拟阻力的生成主要基于 God-Object 算法^[60]，该算法结合胡克定律实现了力的实时计算与反馈。God-Object 算法的核心原理是通过计算虚拟工具与虚拟物体之间的最短距离来生成相应的虚拟阻力。如图 4-3 所示，该算法将虚拟工具简化为一个代理点，通过迭代计算代理点到虚拟物体表面的最短距离，并依据胡克定律计算接触力。其数学模型可表示为：

$$F = -kd \quad (4-4)$$

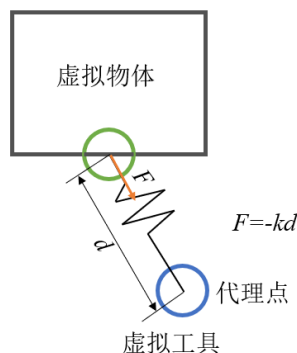


图 4-3 God-Object 算法图

其中， k 表示虚拟工具的弹性系数， d 为代理点与虚拟目标之间的最短距离，负号表示交互力的方向与位移方向相反，即从代理点指向虚拟物体表面。根据式（4-4），当代理点与虚拟目标之间的距离增大时，系统将产生与之成正比的交互力，这种线性关系确保了力反馈的准确性和实时性。该算法不仅能够精确模拟虚拟环境中的力学特性，还能有效降低计算复杂度，为实时交互提供了可靠的技术支持。在 Unity3D 中实现操作端的力觉交互功能，需遵循以下步骤：首先，通过导入 OpenHaptics SDK 的动态链接库到 Unity3D 项目中，接着，通过编写脚本初始化设备，获取设备句柄，并设置力反馈参数。在运行时，根据虚拟环境中物体的物理属性和用户的操作，计算出需要反馈给用户的力，并通过 API 将力反馈指令发送到设备。

4.2.2 VR 交互

头戴式显示器是一种将微型显示屏集成在头戴设备中的显示技术产品，它通过紧贴用户眼睛的屏幕，为用户呈现出高分辨率、大视角的视觉内容。该设备通常配备有定位传感器等部件，能够实时追踪用户的头部运动，从而让用户在虚拟或增强的环境中获得身临其境的交互体验，广泛应用于虚拟现实（VR）、增强现实（AR）以及混合现实（MR）等多个领域。目前市面上头戴式立体显示器有很

多种，本文选用的是 HTC VIVE Pro2，如图 4-4 所示。HTC Vive Pro 2 是一款高端虚拟现实（VR）头显，以其高分辨率（双眼 4896×2448 像素）、120 Hz 刷新率和 120°宽广视场角为用户带来极致沉浸式体验。它配备精准的 SteamVR Tracking 2.0 追踪系统、高品质空间音频和舒适的轻量化设计，支持多种输入设备和丰富的 SteamVR 内容，广泛应用于 VR 游戏、专业培训、工业设计和教育领域，为用户带来极致的沉浸式 VR 体验。



图 4-4 HTC VIVE Pro2 设备

HTC VIVE Pro2 的硬件系统由显示头盔、操作手柄和激光追踪器组成。显示部分采用双屏配置，分别向左右眼投射独立影像。该设备属于眼镜形态的立体显示装置，其定位技术基于激光灯塔原理，利用两个发射器发出红外激光。头盔和手柄内置感应模块，能够捕捉激光信号，并通过分析光束角度来精确确定设备在三维空间中的具体方位。HTC VIVE Pro2 系统搭建如图 4-5 所示。

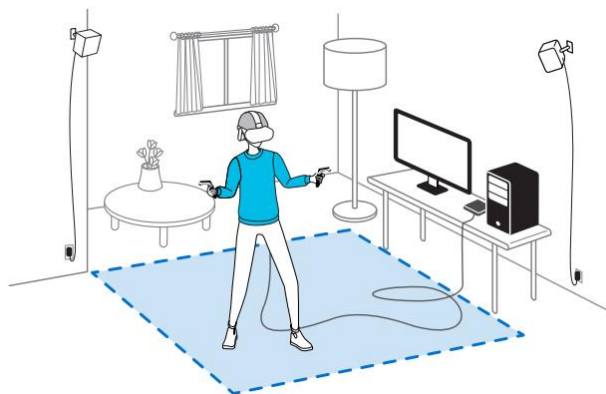


图 4-5 HTC VIVE Pro2 虚拟现实设备工作原理

在 Unity3D 平台上开发面向 HTC VIVE Pro2 的虚拟现实应用时，Unity3D 的 XR 系统提供了强大的支持。Unity XR 是一个开放框架，兼容多种 VR 设备，包括 HTC VIVE Pro2。为了实现与 HTC VIVE Pro2 的交互，开发者需导入 Steam

VR Plugin，这是专为 HTC VIVE 系列设计的 Unity3D 插件。该插件提供设备追踪、手柄输入处理和空间定位等功能，极大地简化了开发流程。开发者可通过 Unity Asset Store 下载并导入该插件，然后启用 Unity3D 的 XR 系统并完成相关配置。HTC VIVE Pro2 系统搭建主要如下：

(1) 安装激光定位器并规划空间。如图 4-5 所示，为了最大限度地扩大游玩区，将激光定位器安装在房间内的对角位置，每个定位器的水平视场为 150° ，垂直视场为 110° 。空间设置需要至少为 $2\text{ m} \times 1.5\text{ m}$ 的可活动空间。

(2) 使用连接器将 HTC VIVE Pro2 头盔与电脑主机相连。确认连接无误后，开启连接器的电源，头盔随即完成连接并进入工作状态。

(3) 软件安装与配置。首先，从 Steam 官方网站获取并安装客户端程序，随后在 Steam 平台的“库”中搜索并完成 Steam VR 应用的安装。最后启动 Steam VR，并将头盔和手柄放置在激光定位器覆盖的空间内，完成设备的自动连接与校准。

(4) 使用区域设置。首先，将手柄对准显示器并按下扳机键，确定操作空间的正前方参考方向。接着，把手柄平稳放在地面上，完成地板校准。最后，用户手持设备在激光定位器覆盖区域内移动，手柄会通过振动反馈当前位置。走完闭合区域后，系统将自动确定可用操作空间。

4.3 坐标系转换

在机器人运动学的建模过程中，坐标系的准确转换是实现高精度运动控制的核心环节。Unity3D 平台默认采用左手坐标系，而机器人运动学的建模通常依赖于右手坐标系，如图 4-6 所示。为了确保两者之间的兼容性，必须通过变换矩阵实现坐标系转换。

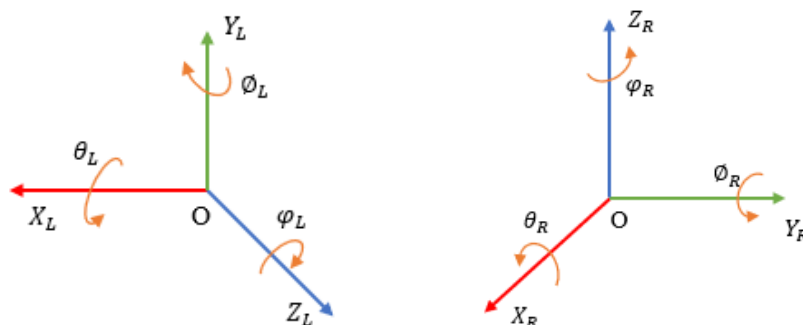


图 4-6 左右手坐标系对应关系

在 Unity3D 的左手坐标系中，物体的位置由平移矢量 $P_L = (P_{XL}, P_{YL}, P_{ZL})$ 表示。为了将其转换为右手坐标系下的位置 $P_R = (P_{XR}, P_{YR}, P_{ZR})$ ，需要引入一个位置变换矩阵 T_P 。根据图 4-6 左右手坐标系对应关系，位置变换矩阵可以定义为：

$$T_P = \begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (4-5)$$

通过该矩阵，左手坐标系下的位置 P_L 可以转换为右手坐标系下的位置 P_R ：

$$P_R = T_P \cdot P_L \quad (4-6)$$

其中， P_L 和 P_R 均为齐次坐标形式，即 $P_L = (P_{XL}, P_{YL}, P_{ZL}, 1)^T$ ， $P_R = (P_{XR}, P_{YR}, P_{ZR}, 1)^T$ 。

在 Unity3D 中，物体的旋转由绕 X、Y、Z 轴的欧拉角 $(\theta_L, \psi_L, \varphi_L)$ ，对应的旋转矩阵为 R_L 。为了将其转换为右手坐标系下的旋转矩阵 R_R ，需要引入姿态变换矩阵 T_R 。根据左右手坐标系对应关系，姿态变换矩阵可以定义为：

$$T_R = \begin{bmatrix} 0 & 0 & -1 \\ 1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix} \quad (4-7)$$

通过该矩阵，左手坐标系下的旋转矩阵 R_L 可以转换为右手坐标系下的旋转矩阵 R_R ：

$$R_R = T_R \cdot R_L \cdot T_R^{-1} \quad (4-8)$$

其中， $T_R^{-1} = T_R$ ，因为 T_R 是对称矩阵。

为了同时描述位置和姿态的转换，可以使用齐次变换矩阵将平移和旋转信息结合在一起，形式如下：

$$H = \begin{bmatrix} R & P \\ 0 & 1 \end{bmatrix} \quad (4-9)$$

其中， R 是旋转矩阵， P 是平移矢量。

在 Unity3D 中，左手坐标系的齐次变换矩阵 H_L 可以表示为：

$$H_L = \begin{bmatrix} T_L & P_L \\ 0 & 1 \end{bmatrix} \quad (4-10)$$

通过位置变换矩阵 T_P 和姿态变换矩阵 T_R ，可以将其转换为右手坐标系的齐次变换矩阵 H_R ：

$$H_R = T_P \cdot H_L \cdot T_R \quad (4-11)$$

4.4 打磨机器人运动学算法验证

验证机器人正逆运动学解的准确性是实现虚实映射的关键。本文在 Unity3D 中验证机器人运动学的准确性。如图 4-7 所示，平台右侧设有 Joint 控制区域，用户可通过调节关节滑条输入机器人各关节角度值。当设置六个关节角度分别为 35.21° 、 33.37° 、 47.91° 、 -17.22° 、 60.51° 、 57.08° 时，系统通过编写的正运动学计算脚本，实时求解末端执行器在 Unity3D 世界坐标系下的位姿，并将计算结果动态显示在 Position 区域。如图 4-7 所示，当前末端执行器的位姿为 $(0.26, 0.17, -0.56, -0.61, 0.46, 2.71)$ ，与图 4-8 所示的末端位姿一致，验证了正运动学算法的正确性，为后续实现精确的虚实控制奠定了理论基础。



图 4-7 正运动学下的关节角和位姿

如图 4-8 所示，在 Position 控制区域中，通过输入末端的目标位姿来验证逆运动学算法的准确性。在 P_x 、 P_y 、 P_z 、 R_x 、 R_y 、 R_z 输入框中分别输入 0.26、0.17、-0.56、-0.61、0.46、2.71 后，系统通过编写的逆运动学脚本实时计算出机器人各关节的角度值，并将计算结果动态传输至 Joint 控制区域的滑条进行可视化显示。实验结果显示，六个关节的角度值分别为 35.21° 、 33.37° 、 47.91° 、 -17.22° 、 60.51° 、

57.08°, 与图 4-8 中正运动学验证时设置的关节角度完全一致。这一结果不仅证明了在 Joint 区域控制下机器人能够准确运动至指定位姿点, 同时也验证了逆运动学算法的正确性。此外, 当机器人末端执行器到达目标位姿时, Joint 区域实时显示的关节转角与正运动学验证时的输入角度值完全吻合, 进一步证实了本文所实现的正逆运动学算法的可靠性和准确性。



图 4-8 逆运动学下的位姿和关节角

4.5 虚实控制研究

实现操作端与虚拟仿真平台之间的虚实通信是设计打磨机器人交互系统的关键环节。本系统采用网络通信, 传输层协议选择可靠性较高的 TCP 协议, 通过 Socket 进行连接通信。服务器端监听客户端的连接请求, 建立连接后实现数据的收发。

系统采用分布式架构, 主要由操作端、虚拟执行端和实体执行端三部分构成, 通过 Visual Studio 服务器实现各模块间的数据交互与协同控制。先通过 OpenHaptics SDK 实时采集操作端设备的关节转角值, 计算其末端位姿, 并将数据作为系统输入参数传输至 Visual Studio 服务器。服务器端对数据进行预处理, 包括坐标系转换和空间映射, 将操作端的位姿数据转换为虚拟执行端可识别的标准化格式。再通过 Socket 通信协议将转换后的数据实时传输至 Unity3D 客户端。Unity3D 客户端接收到数据后, 利用逆运动学求解算法计算虚拟执行端机器人的关节角度, 并实时更新虚拟模型的状态, 实现操作端对虚拟执行端的精确同步控

制。为增强系统的交互性和沉浸感，虚拟执行端的末端位姿以及力数据通过 Visual Studio 服务器反馈至操作端，实现操作者的触觉感知。同时，虚拟执行端的位姿数据通过 Visual Studio 服务器传输至机器人控制器服务器，驱动实体机器人执行相应动作，确保虚拟环境与实体机器人保持同步运行。

操作端与实体执行端之间的交互是遥操作的前提。操作端设备支持开发者使用 OpenHaptics 或 QuickHaptics 开发触觉应用程序，通过在 Visual Studio 中创建控制台应用并配置操作端开发环境，利用 OpenHaptics SDK 实时读取操作端设备的各关节转角值。结合空间映射算法，将操作端的位姿数据转换为机器人可识别的位姿信息，并通过 Socket 通信协议将数据实时传输至客户端。客户端接收到数据后，利用逆运动学求解函数计算出执行端机器人的关节角度，并将这些数据传输至实体机器人控制器，驱动机器人执行相应动作。同时，实体执行端机器人通过力传感器采集力反馈数据，并将其发送至服务器。服务器将这些力数据实时反馈至操作端，从而实现操作端对实体机器人的精确控制以及力觉交互功能。操作端设备上的按钮用于控制机器人的运动，如图 4-9 所示，按下按钮发送数据 1，松开按钮发送数据 0，通过服务器端发送给客户端控制机器人跟随运动。深色按钮用于到达指定位置后开始跟随运动，浅色按钮用于开始或暂停运动。



图 4-9 操作端按钮

4.6 本章小结

本章围绕虚拟现实中的交互与通信技术展开研究，探讨了碰撞检测、虚拟现实交互、坐标系转换、运动学算法验证及通信技术等内容。通过层次包围盒算法和力反馈交互技术，增强了虚拟环境的交互性和沉浸感。基于 Unity3D 平台验证了打磨机器人的正逆运动学算法，确保了运动学模型的准确性。同时，结合 VR

交互技术,进一步提升了虚拟现实系统的交互性和用户友好性。设计了分布式通信系统,实现了操作端、虚拟执行端与实体机器人的协同控制,为虚实同步运行提供了技术支持。确保操作者的动作和意图能够准确地传输到虚实执行端机器人,同时机器人的状态信息以及打磨过程的力也能实时反馈给操作者。

第5章 虚拟现实打磨机器人交互系统测试实验

为验证基于虚拟现实的打磨机器人交互系统,本章通过虚拟位置映射实验和实体机器人位置映射实验、力觉交互实验和虚实打磨实验来验证系统的位置映射精度,以评估系统的力反馈性能和实际打磨效果。最后,对本章内容进行总结,为系统的进一步优化提供依据。

5.1 实验平台搭建

虚拟现实打磨机器人交互系统测试实验是在特定的系统环境与软件版本条件下开展的,选用 64 位的 Win11 系统、Unity3D 2021.3.32f1c1 版本以及 Visual Studio 2022 版本,硬件环境如表 5-1 所示,系统开发软件环境如表 5-2 所示。

表 5-1 硬件环境

类别	参数配置
CPU	AMD Ryzen 7 5800H with Radeon Graphics 3.20 GHz
内存	16G
显卡	NVIDIA GeForce RTX 3070 Laptop GPU

表 5-2 系统开发软件环境

类别	工具
操作系统	Win11
建模及优化软件	SolidWorks、Blender
虚拟交互软件	Unity3D 2021.3.32f1c1
编程语言	C#
编程平台	Visual Studio2022

如图 5-1 所示,虚拟现实打磨机器人交互系统由操作端、虚拟执行端与实体执行端三部分构成。操作端包含 Geomagic Touch 力反馈设备、KH-01 步进电机控制器、24 V 直流开关电源、步进电机驱动器、滚珠丝杆直线滑台模组及光学升降平台 Z 轴工作台,形成精密运动控制单元。虚拟执行端基于计算机与 HTC VIVE Pro2 虚拟现实头显构建交互环境,集成虚拟机器人模型与工件以实现动态仿真。

实体执行端由实体打磨机器人、3D 打印的装置，将打磨机稳固安装在机器人末端。为实时监测打磨过程中的接触力，在机器人末端与打磨机之间安装了六维力传感器。为确保打磨过程的稳定性，系统采用重型角固式平口钳（承重 10 KG）固定工件，有效防止工件在打磨过程中发生位移。这种硬件配置既保证了系统的精确控制能力，又确保了打磨作业的稳定性和安全性。

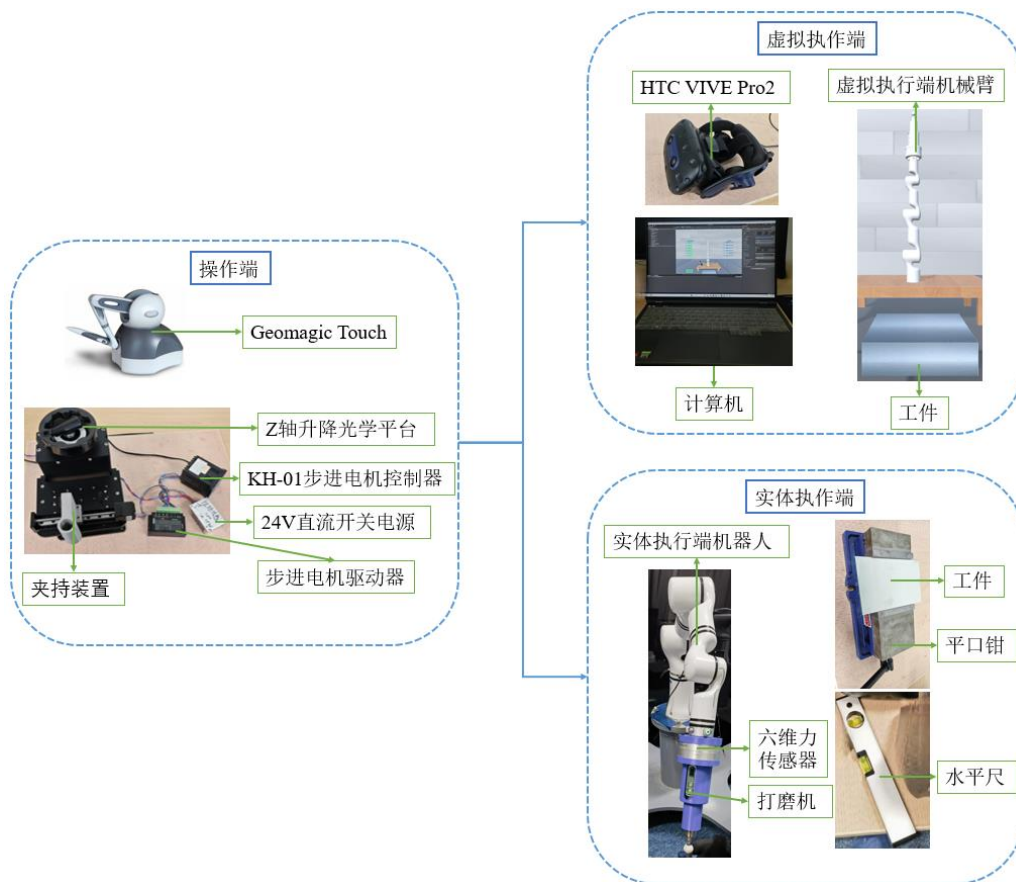


图 5-1 实验设备图

5.2 位置映射实验

5.2.1 虚拟位置映射实验

验证操作端设备与虚拟执行端机器人之间的空间映射算法是实现精准控制的关键，本文设计了空间轨迹跟随实验，如图 5-2 所示。用户手持操作端设备在 Unity3D 场景内绘制英文字母“M”，通过 Visual Studio 服务器实时获取操作端各关节数据，利用正运动学算法求解操作端设备的末端位姿。随后，通过空间映射算法对位姿数据进行处理，并将位姿数据实时传输至 Unity3D 虚拟仿真平台的 Socket 客户端。虚拟仿真平台基于接收到的末端位姿数据，结合编写的逆运动学

脚本解算出虚拟执行端机器人的各关节角度，驱动机器人进行同步运动。为确保系统实时性，将 Unity3D 的帧率设置为 100 帧，使每次调用 Update() 函数时都能及时刷新数据，从而实现虚拟执行端机器人在虚拟空间中的精确轨迹复现。通过编写的数据采集脚本，系统可同时记录操作端设备与虚拟执行端机器人的末端坐标数据，并保存为 TXT 文件。最后，利用 MATLAB 对采集数据进行处理，绘制操作端与虚拟执行端末端点的位置跟随轨迹对比图，如图 5-3 所示，直观展示了空间映射算法的准确性和系统的跟踪性能。



图 5-2 虚拟位置映射实验

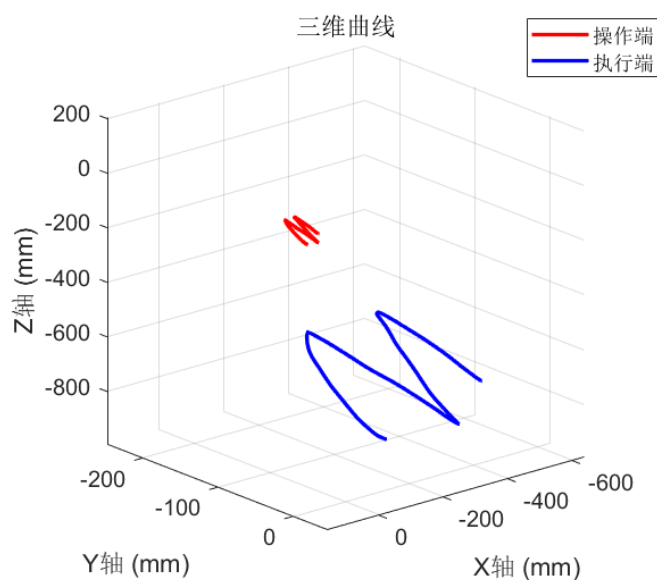


图 5-3 三维空间末端轨迹对比图

操作端与虚拟执行端末端在 X 、 Y 、 Z 三个轴向上的位置映射情况及变化趋势如图 5-4 所示。

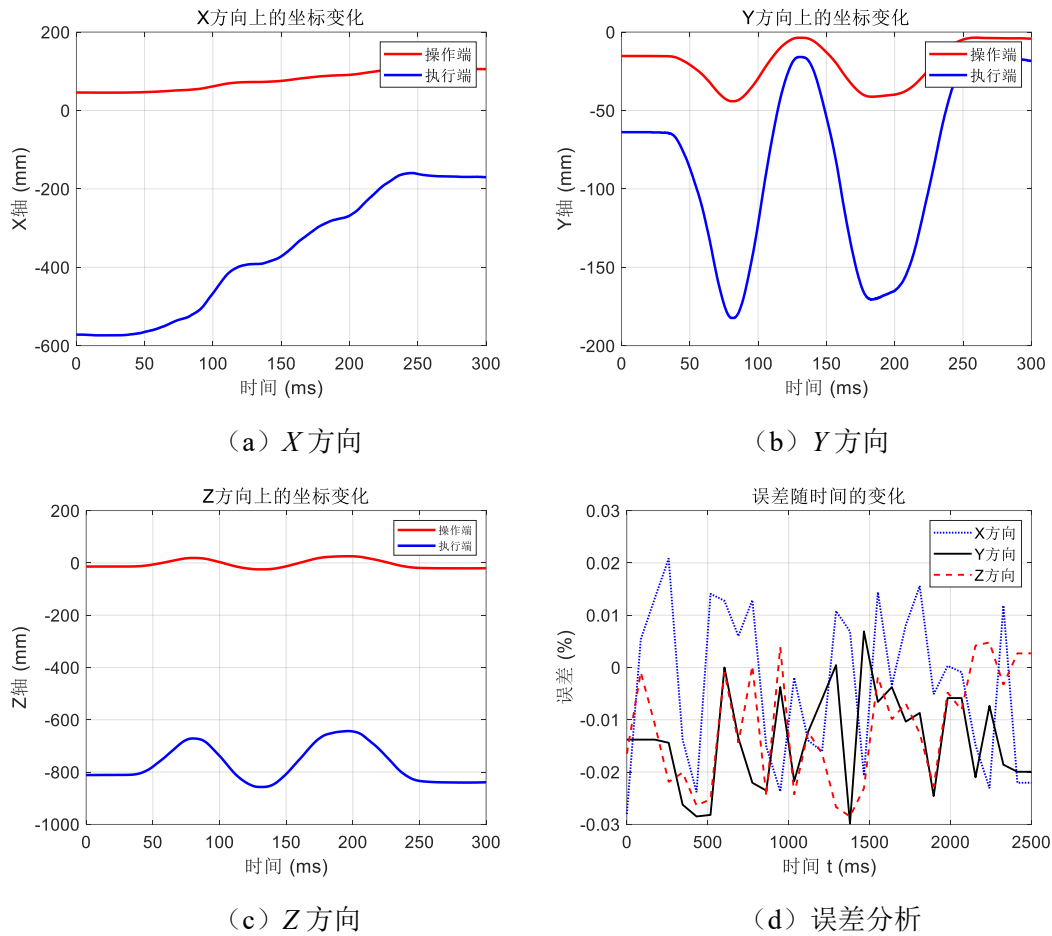


图 5-4 末端点坐标 X 、 Y 、 Z 值随时间变化图

根据图 5-3 和图 5-4 的实验数据分析可知，操作端末端点与虚拟执行端末端点的运动轨迹均表现出良好的平滑性，两者的运行趋势保持高度一致，且虚拟执行端的运动轨迹呈现出预期的比例放大效果。进一步分析图 5-4(d) 所示的误差分布可知，末端位置点在 X 、 Y 、 Z 三个方向上的误差均控制在 $\pm 0.03\%$ 的范围内，验证了执行端机器人的控制精度和可靠性。

5.2.2 实体机器人位置映射实验

在完成操作端对虚拟执行端机器人的控制验证后，进一步开展操作端对实体执行端机器人的控制效果验证实验。如图 5-5 所示，用户手持操作端设备在三维空间内绘制英文字母“N”，Visual Studio 服务器端实时读取操作端控制器各关节角度数据，通过运动学正解算法求解操作端末端坐标。随后，系统通过 Socket 服务器端以 100 Hz 的频率将坐标数据实时传输至执行端机器人。执行端机器人接

收数据后，首先通过空间映射算法计算映射后的目标位置坐标，然后利用运动学逆解算法求解各关节转角信息，最终驱动实体执行端机器人在空间中复现字母"N"的轨迹。实验过程中，系统将操作端和实体执行端的位置数据同步保存为TXT文件，并利用 MATLAB 绘制两者的末端点位置映射轨迹对比图，如图 5-6 所示，直观展示了操作端对实体执行端机器人末端点位置映射轨迹图。

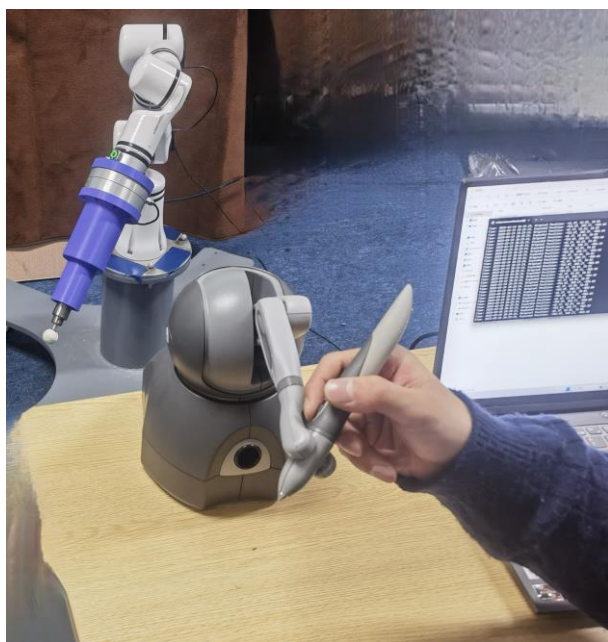


图 5-5 实体机器人位置映射实验

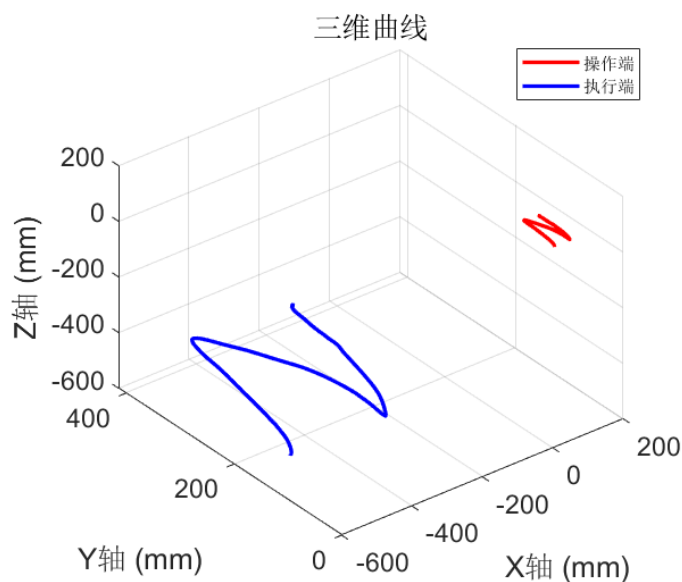


图 5-6 遥操作三维空间末端轨迹对比图

操作端与虚拟执行端末端在 X 、 Y 、 Z 三个轴向上的位置映射情况及变化趋势如图 5-7 所示。

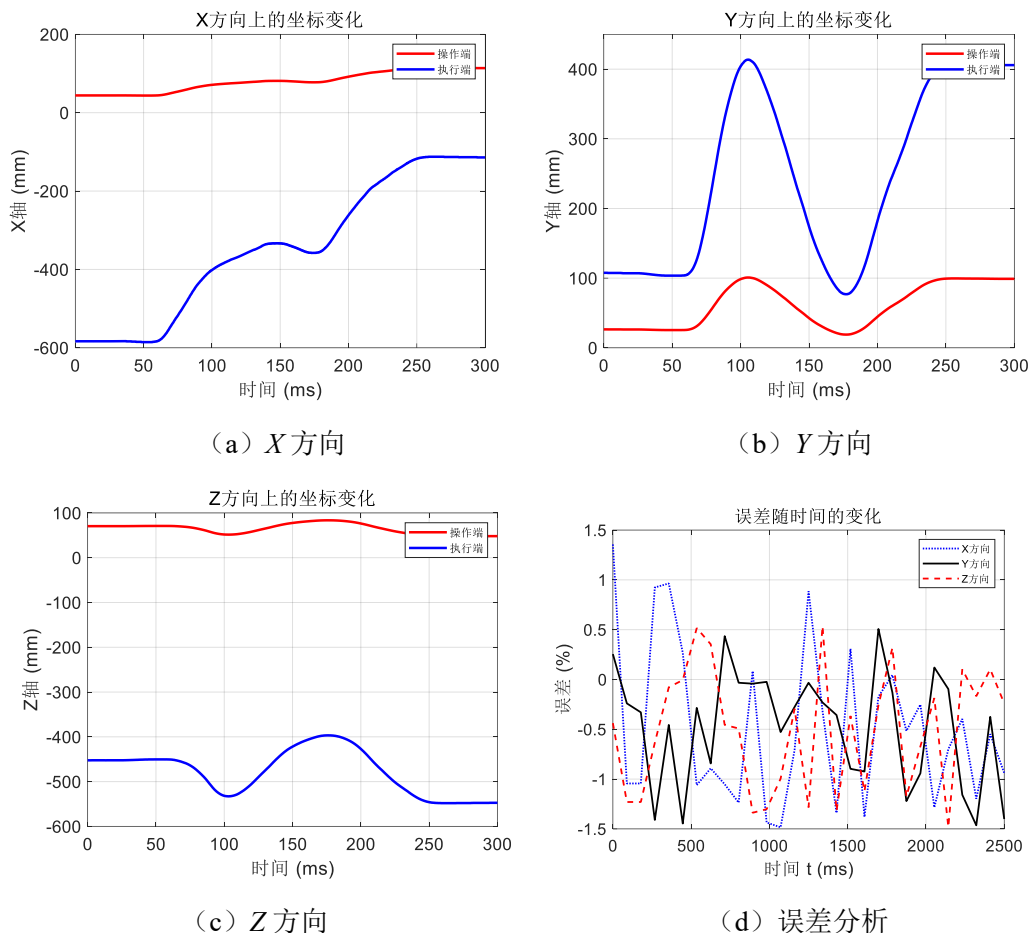


图 5-7 遥操作末端点坐标 X、Y、Z 值随时间变化图

根据图 5-6 和图 5-7 的实验数据分析可知，操作端末端点与实体执行端末端点的运动轨迹均表现出良好的平滑性，两者的运行趋势基本保持一致，且实体执行端的运动轨迹呈现出预期的比例放大效果。进一步分析图 5-7(d) 所示的误差分布可知，末端位置点在 X、Y、Z 三个方向上的误差均控制在 $\pm 1.5\%$ 的范围内。这一结果表明，实体执行端能够被操作端设备精确控制，验证了系统在实际物理环境中的控制精度和可靠性。

5.3 力觉交互实验

为进一步验证打磨机器人交互系统的效果，本文设计了力觉交互实验，选用型号为 IIM-F95-600-30-22141 的六维力传感器，X、Y、Z 三轴力量程为 600 N，力矩为 30 Nm，六维力精度 $< 0.5\% \text{ FS}$ ，打磨头的材质为白刚玉，工件材质为铝。针对实际打磨作业环境中可能存在的障碍物，在虚拟环境中为工件添加了碰撞体组件。为消除手持操作时的手部抖动对打磨精度的影响，系统采用步进电机控制

器套件进行运动控制。在硬件集成方面，通过 3D 打印技术制作专用夹持装置，将操作端与滑台机构连接。其中，滚珠丝杠机构作为核心传动部件，将电机的旋转运动转化为平稳的直线运动，确保打磨过程中的运动轨迹平滑稳定。系统采用 Z 轴光学平台作为基础支撑平台，将操作端设备固定于其上，并将滚珠丝杆直线滑台模组安装于光学升降平台的 Z 轴工作台。通过 Z 轴光学平台调节操作端高度，实现对打磨力的精准控制。在系统校准环节，将实体工件与机器人基座的相对位置与虚拟环境保持一致，并使用水平尺确保 Z 轴光学平台和工件的水平度，保证打磨机与工件表面垂直接触，虚拟力觉交互如图 5-8 所示。



图 5-8 虚拟打磨测试图

当操作者佩戴 HTC VIVE Pro2 进行作业时，步进电机带动操作端设备控制虚实执行端机器人接触工件，系统通过碰撞检测生成碰撞点信息。这些信息通过 Socket 通信传输至操作端，从而实现设备的力觉交互功能。在力控制方面，通过调节 Z 轴光学平台高度来控制打磨头与工件的接触程度，间接调节打磨力的大小。步进电机控制器的运行速度设定为 10 mm/s，带动操作端设备进行水平运动，进而驱动虚实执行端机器人执行打磨作业。为保护操作端力设备受过大冲击损坏，在实体机器人实验阶段，如图 5-9 所示。将机器人打磨力按 1:30 的比例缩小后传输至操作端设备。系统从执行端机器人受力信息中提取 Z 轴方向受力数据，同时通过操作端末端力传感器采集操作者受力信息（即操作端设备的反馈力大小），并将两者进行对比分析。实验结果如图 5-10、5-11、5-12 所示，分别展示了 30 N、60 N、90 N 三种工况下虚实力测试图。



图 5-9 实体机器人打磨测试图

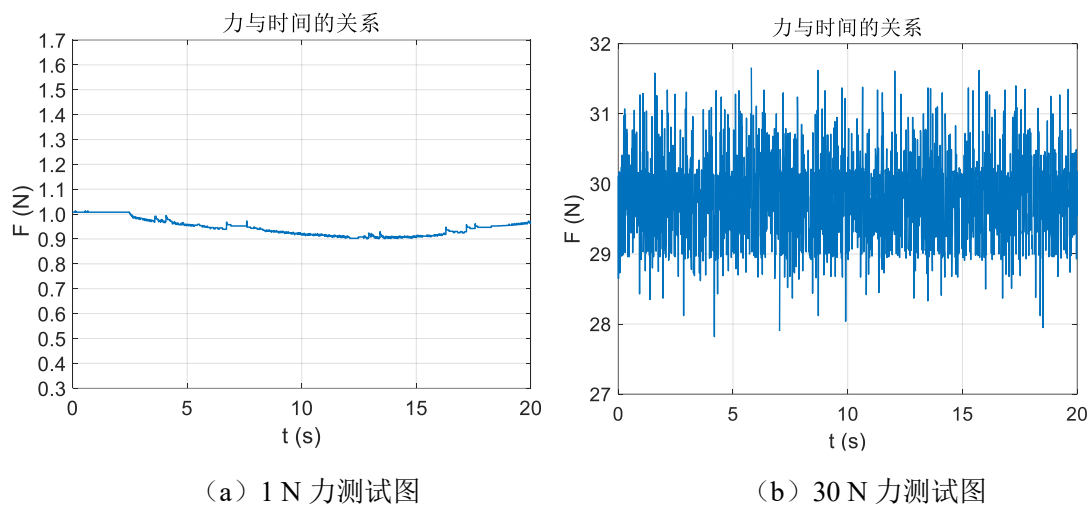


图 5-10 30 N 虚实力测试图

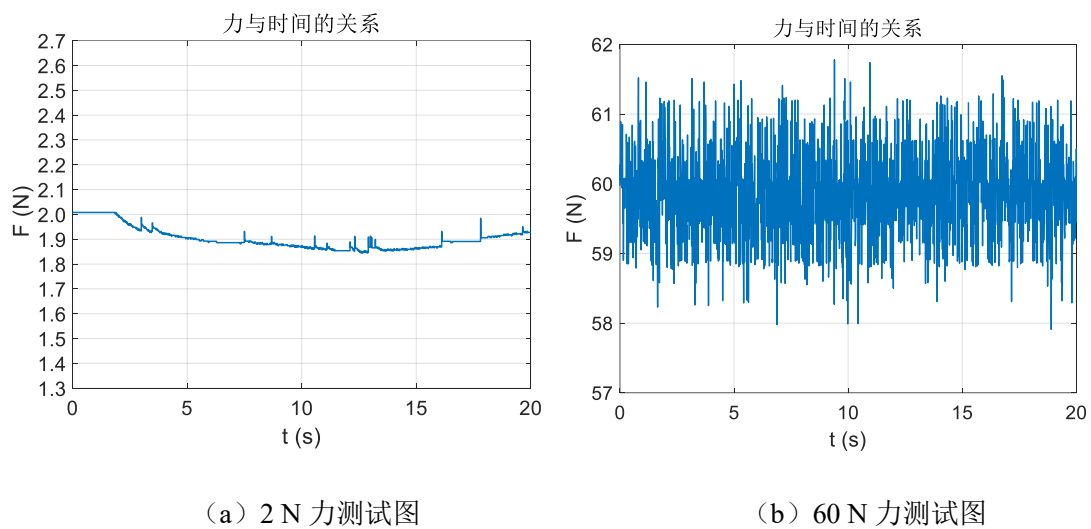


图 5-11 60 N 虚实力测试图

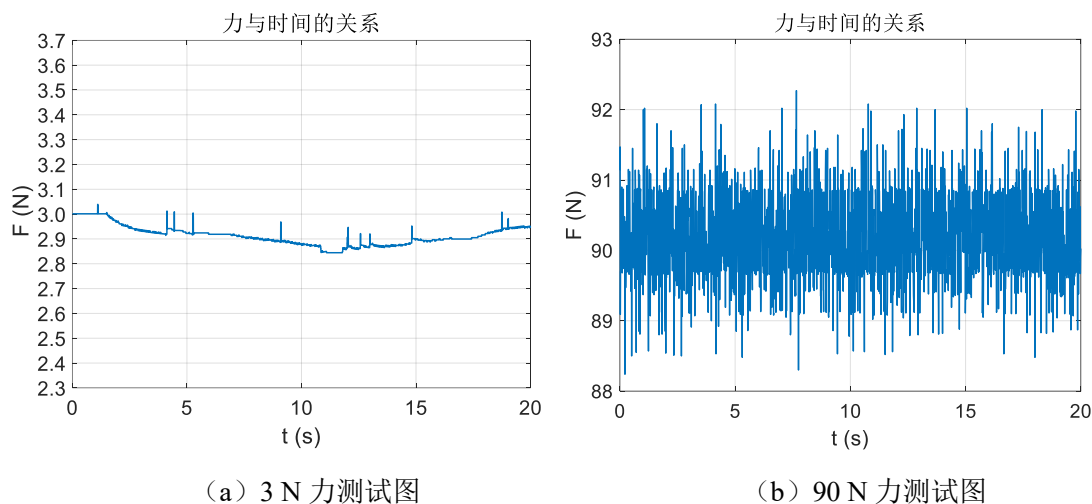


图 5-12 90 N 虚实力测试图

通过对实验数据的分析可以看出,操作端设备的输出反馈力与执行端机器人的打磨力变化呈现良好的对应关系。具体而言,当执行端机器人施加 30 N 打磨力时,执行端反馈力的误差范围为 0.1280,这与 1 N 虚拟打磨力下的误差范围 0.1114 基本吻合;当执行端机器人施加 60 N 打磨力时,操作端反馈力的变化范围为误差范围为 0.0645,与 2 N 打磨力下的变化范围 0.0818 相匹配;当执行端机器人施加 90 N 打磨力时,操作端反馈力的误差范围为 0.0487,与 3 N 打磨力下的变化范围 0.0648 相对应。这些数据表明,操作端设备在虚拟打磨过程中输出的反馈力能够准确反映执行端机器人的打磨力变化,验证了虚拟现实在打磨作业中的可靠性和准确性。

5.4 虚实打磨实验

操作者通过佩戴 HTC VIVE Pro 2,通过步进电机控制器操控操作端设备的运动速度(10mm/s),间接控制虚拟打磨机器人的打磨头运动,在 Unity3D 中编写打磨头转速脚本,设置转速为 2500 RPM,接着用户通过光学升降平台 Z 轴工作台控制打磨头逐渐接触工件,通过观察屏幕中接触力的大小,逐渐增加到 1 N、2 N、3 N,然后启动步进电机控制器进行打磨测试实验,实时的材料去除效果如图 5-13 所示,图(a)、(b)、(c)分别对应 1 N、2 N、3 N 时的打磨效果。

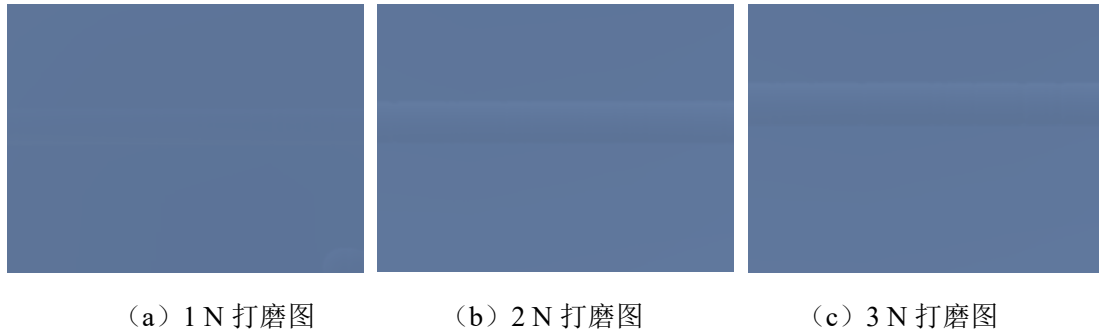


图 5-13 虚拟打磨效果图

用户通过步进电机控制器控制操作端设备的运动速度（10 mm/s），间接控制实体打磨机器人的打磨头运动，启动打磨机，并设置转速为 2500RPM，接着通过光学升降平台 Z 轴工作台控制打磨头逐渐接触工件，通过观察六维力传感器中接触力的大小，逐渐增加到 30 N、60 N、90 N，然后启动步进电机控制器控制实体机器人进行往复打磨测试实验，实验结果如图 5-14 所示，图（a）、（b）、（c）分别对应 30 N、60 N、90 N 时的打磨效果。

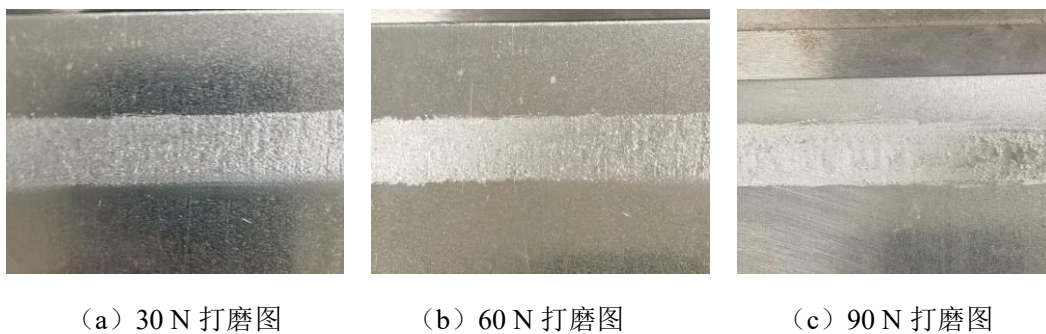


图 5-14 实际打磨效果图

5.5 本章小结

本章对虚拟现实打磨机器人交互系统进行了多维度测试，涵盖位置映射、力觉交互和虚实打磨实验。位置映射实验验证了操作端设备能精准控制虚拟与实体环境中的执行端机器人，确保了系统的高精度。力觉交互实验通过遥操作对比虚拟打磨力与实际打磨力的变化范围，二者基本一致，验证了虚拟打磨力反馈的真实性。虚实打磨实验模拟实际操作流程，进一步验证了系统的可行性和稳定性。实验结果表明，该系统能够有效实现位置映射、力觉交互和虚实结合的打磨操作，为工业应用提供了可靠支持，各项功能均达到预期目标，为后续优化和应用奠定了基础。

第6章 总结与展望

6.1 总结

本文聚焦虚拟现实（VR）打磨机器人交互系统的技术痛点，针对打磨机器人系统中异构模型差异导致的控制精度不足、打磨工艺过程中存在的不确定性与操作复杂性，以及打磨作业缺乏高保真的视觉-力觉融合反馈等关键技术问题展开研究，基于此，本研究致力于开发一套具有高保真视觉-力觉融合特性的虚拟打磨交互系统，为智能制造领域的人机协作提供创新解决方案。

针对操作端与执行端结构差异导致的控制精度不足，采用改进的 D-H 参数法建立二者的连杆坐标系，并对操作端进行正运动学求解。通过 Pieper 解法对执行端进行逆运动学求解，并利用蒙特卡洛法求解和可视化操作端与执行端的工作空间，分析二者之间的差异性，并设计空间映射算法，从而实现了操作端、虚拟执行端、实体执行端之间的精确控制。

针对打磨任务中的环境不确定性和操作复杂性，设计了一套虚拟现实打磨机器人交互系统。通过构建虚拟环境、开发用户界面、建立工作网格模型以及球形打磨头变形算法，系统能够实时渲染打磨效果。研究了碰撞检测、力觉反馈、虚拟现实交互、坐标系转换以及虚实交互等问题，显著提升了系统的操作精度和协同效率。

为验证虚拟现实打磨机器人交互系统的有效性和实用性，本文构建了系统性实验方案，从多维度进行测试。首先，通过虚拟位置映射实验，验证了空间映射算法的准确性及系统跟踪性能，位置误差控制在 $\pm 0.03\%$ ，算法精确性得到证明。其次，通过实体机器人位置映射实验，验证了实际机器人位置精确映射能力，误差范围仅为 $\pm 1.5\%$ ，进一步验证了系统可靠性。最后，通过力觉交互实验，对比虚实打磨过程中的力反馈数据，二者误差基本吻合，证明系统能满足真实力觉交互需求。实验结果从位置精度、映射能力和力觉交互等维度验证了系统的有效性和实用性。

6.2 展望

尽管本文在虚拟现实打磨机器人交互系统的设计与实现方面取得了一定的成果，但仍有许多值得进一步探索和改进的方向。未来的研究可以从以下几个方面展开：

（1）当前基于虚拟现实的机器人打磨交互系统虽然能够实现加工过程的实时材料去除渲染，但其在模拟过程中尚未充分考虑工件表面形貌特征的实时演化特性。后续研究应着力构建融合表面粗糙度动态演化模型与实时物理反馈的多物理场耦合机制，建立跨尺度表面形貌表征框架，通过视觉渲染精度与实际表面质量演变的同步映射，实现工艺仿真效果与真实物理过程的高度一致性。

（2）未来的研究将聚焦于系统的落地部署以及稳定性和安全性研究。在落地部署方面，可分三个阶段推进：首先在实际工厂环境中进行试点部署，选择典型打磨任务场景，完成系统与工厂生产管理系统的初步集成，验证系统在真实生产流程中的适用性和兼容性；接着在小规模生产线上推广，根据试点反馈优化系统功能，建立完善的运行和维护流程；最后逐步扩大系统应用范围，与工厂智能制造生态系统深度融合，实现打磨任务的智能化管理。

（3）未来的研究可以探索将 5G、Wi-Fi 6 等高速无线通信技术集成到系统中，以替代传统的有线连接方式。通过无线通信，操作端设备与执行端机器人之间的数据传输将更加灵活，减少线缆束缚，提升系统的可移动性和部署便捷性。

参考文献

- [1] 宋大伟, 郭雯. 我国服务型制造发展和“十四五”展望[J]. 中国科学院院刊, 2020, 35(11): 1358-1365.
- [2] Zhou L, Zhang L, Konz N. Computer vision techniques in manufacturing[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2022, 53(1): 105-117.
- [3] 王仕强, 于佩航, 喻建胜, 等. 放喷管线自动打磨爬行机器人设计与试验[J]. 机械设计与研究, 2023, 39(01): 22-25+30.
- [4] Qian N, Chen J, Khan A M, et al. Towards sustainable grinding of difficult-to-cut alloys—a holistic review and trends[J]. Chinese Journal of Mechanical Engineering, 2024, 37(1): 23.
- [5] 浦一子, 肖冰, 高睿, 等. 钢轨打磨用钎焊金刚石砂轮研究[J]. 金刚石与磨料磨具工程, 2022, 42(03): 325-331.
- [6] Li J, Zou L, Luo G, et al. Enhancement and evaluation in path accuracy of industrial robot for complex surface grinding[J]. Robotics and Computer-Integrated Manufacturing, 2023, 81: 102521.
- [7] Fracaro S G, Glassey J, Bernaerts K, et al. Immersive technologies for the training of operators in the process industry: A Systematic Literature Review[J]. Computers & Chemical Engineering, 2022, 160: 107691.
- [8] Asad M M, Naz A, Churi P, et al. Virtual reality as pedagogical tool to enhance experiential learning: a systematic literature review[J]. Education Research International, 2021, 2021(1): 7061623.
- [9] Chen Z, Zhu H, Song L, et al. Wireless multiplayer interactive virtual reality game systems with edge computing: Modeling and optimization[J]. IEEE transactions on wireless communications, 2022, 21(11): 9684-9699.
- [10] 黄凯歌, 惠延波, 刘永刚, 等. VR交互式三维虚拟起重装备建模与仿真[J]. 计算机工程与应用, 2024, 60(07): 204-211.

- [11] Lyu K, Brambilla A, Globa A, et al. An immersive multisensory virtual reality approach to the study of human-built environment interactions[J]. Automation in construction, 2023, 150: 104836.
- [12] 张娅婕, 陈永当, 毛勃越. 基于 VR 的工业机器人运动控制实训系统设计[J]. 智能计算机与应用, 2025, 15(02): 96-102.
- [13] Jiang Y, Yin S, Li K, et al. Industrial applications of digital twins[J]. Philosophical Transactions of the Royal Society A, 2021, 379(2207): 20200360.
- [14] 殷光淼, 朱国力, 谢哲, 等. 基于数字孪生的盾构机换刀机器人监控系统[J]. 计算机集成制造系统, 2024, 30(03): 811-824.
- [15] Javaid M, Haleem A, Suman R. Digital twin applications toward industry 4.0: A review[J]. Cognitive Robotics, 2023, 3: 71-92.
- [16] 史秋雨. 工业机器人应用系统虚拟仿真技术的研究[D]. 上海第二工业大学, 2020.
- [17] Xie H L, Wang Q H, Ong S K, et al. Adaptive human-robot collaboration for robotic grinding of complex workpieces[J]. CIRP Annals, 2022, 71(1): 285-288.
- [18] Zhu D, Feng X, Xu X, et al. Robotic grinding of complex components: a step towards efficient and intelligent machining—challenges, solutions, and applications[J]. Robotics and Computer-Integrated Manufacturing, 2020, 65: 101908.
- [19] Song X. Virtual Reality and Metaversal Literature[J]. Theoretical Studies in Literature and Art, 2023, 43(1): 1-11.
- [20] Zhang B, Wu S, Wang D, et al. A review of surface quality control technology for robotic abrasive belt grinding of aero-engine blades[J]. Measurement, 2023, 220: 113381.
- [21] KUKA 机器人可实现自动去除铸造毛刺[J]. 金属加工(热加工), 2018, (11): 11.
- [22] Song H C, Song J B. Precision robotic deburring based on force control for arbitrarily shaped workpiece using CAD model matching[J]. International Journal of Precision Engineering and Manufacturing, 2013, 14: 85-91.
- [23] Li J, Guan Y, Chen H, et al. A high-bandwidth end-effector with active force control for robotic polishing[J]. IEEE Access, 2020, 8: 169122-169135.

- [24] 孙一兰, 柳洪义, 王品, 等. 导弹弹体内壁打磨机器人及其运动轨迹规划[J]. 中国机械工程, 2009, 20(07): 838-843.
- [25] 郭兆根. 金属铸件打磨机器人轨迹规划与柔顺控制研究[D]. 中国矿业大学, 2024.
- [26] 李旭浩. 云制造环境下复杂重型装备制造资源虚拟化建模与封装研究及应用[D]. 重庆大学, 2023.
- [27] Liu S, Lu S, Li J, et al. Machining process-oriented monitoring method based on digital twin via augmented reality[J]. The International Journal of Advanced Manufacturing Technology, 2021, 113: 3491-3508.
- [28] 陈伟才, 段明峰, 张南, 等. 数字孪生驱动的车间管理系统开发[J]. 组合机床与自动化加工技术, 2023, (09): 178-181.
- [29] Tao F, Xiao B, Qi Q, et al. Digital twin modeling[J]. Journal of Manufacturing Systems, 2022, 64: 372-389.
- [30] Boschert S, Rosen R. Digital twin—the simulation aspect[J]. Mechatronic futures: Challenges and solutions for mechatronic systems and their designers, 2016: 59-74.
- [31] 刘晓峰. 基于 Unity3D 的虚拟深孔加工系统的研究与开发[D]. 中北大学, 2020.
- [32] 刘家武. 六自由度串联式力反馈机械臂研发与应用研究[D]. 华南理工大学, 2018.
- [33] 倪建龙. 基于虚拟现实的砂带研抛工艺仿真研究[D]. 华南理工大学, 2018.
- [34] 任豪豪. 基于 Unity3D 的交互式虚拟机器人焊接模拟[D]. 武汉科技大学, 2023.
- [35] 唐磊. 机器人打磨过程虚拟仿真技术研究[D]. 西南科技大学, 2024.
- [36] Ni D, Yew A W W, Ong S K, et al. Haptic and visual augmented reality interface for programming welding robots[J]. Advances in Manufacturing, 2017, 5: 191-198.
- [37] Togias T, Gkourmelos C, Angelakis P, et al. Virtual reality environment for industrial robot control and path design[J]. Procedia CIRP, 2021, 100: 133-138.

- [38] Lin M, San L, Ding Y. Construction of robotic virtual laboratory system based on Unity3D[C]//IOP Conference Series: Materials Science and Engineering. IOP Publishing, 2020, 768(7): 072084.
- [39] Xia P, Xu F, Song Z, et al. Sensory augmentation for subsea robot teleoperation[J]. Computers in Industry, 2023, 145: 103836.
- [40] Fukazu Y, Hara N, Kanamiya Y, et al. Reactionless resolved acceleration control with vibration suppression capability for JEMRMS/SFA[C]//IEEE International Conference on Robotics and Biomimetics, IEEE, 2009: 1359-1364.
- [41] Ma B, Jiang Z, Liu Y, et al. Advances in space robots for on-orbit servicing: A comprehensive review[J]. Advanced Intelligent Systems, 2023, 5(8): 2200397.
- [42] Su Y, Lloyd L, Chen X, et al. Latency mitigation using applied HMMs for mixed reality-enhanced intuitive teleoperation in intelligent robotic welding[J]. The International Journal of Advanced Manufacturing Technology, 2023, 126(5): 2233-2248.
- [43] 王友渔, 胡成威, 唐自新, 等. 我国空间站机器人系统关键技术发展[J]. 航天器工程, 2022, 31(06): 147-155.
- [44] 宋甫俊. 咽喉子机器人建模与远程共享控制研究[D]. 大连海事大学, 2022.
- [45] Singh G, Banga V K. Kinematics and trajectory planning analysis based on hybrid optimization algorithms for an industrial robotic manipulators[J]. Soft computing, 2022, 26(21): 11339-11372.
- [46] 陈菲菲, 居鹤华, 刘潇晗. 一种高精度求解多轴机器人逆运动学的方法[J]. 机械工程学报, 2023, 59(01): 50-58.
- [47] 张功. 虚拟现实辅助遥操作机器人作业系统研究[D]. 江苏科技大学, 2023.
- [48] Granja M, Chang N, Granja V, et al. Comparison between standard and modified Denavit-Hartenberg methods in robotics modelling[C]//Proceedings of the World Congress on mechanical, chemical, and material engineering. 2016, 1(1): 1-10.
- [49] 史艳琼, 李克凡, 卢荣胜, 等. 基于双目视觉的工业机器人运动学参数辨识[J]. 激光与光电子学进展, 2023, 60(14): 272-280

- [50] 李鹏飞, 李抗, 张蕾, 等. 基于 MATLAB 的双机器人系统建模与仿真[J]. 机床与液压, 2016, 44(21): 5-10.
- [51] Zhang S, Li A, Ren J, et al. Kinematics inverse solution of assembly robot based on improved particle swarm optimization[J]. Robotica, 2024, 42(3): 833-845.
- [52] 黄磊. 基于虚拟现实的水下机械臂遥操作仿真系统研究[D]. 江苏科技大学, 2022.
- [53] Wang T, Wang L, Li D, et al. Monte Carlo-based improved ant colony optimization for path planning of welding robot[J]. Journal of King Saud University-Computer and Information Sciences, 2023, 35(7): 101603.
- [54] Lu Y, Yihao L. Design of immersive and interactive application based on augmented reality and machine learning[C]//2021 IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC). IEEE, 2021, 4: 316-323.
- [55] 田锦秀. 基于虚拟现实技术的移动机器人视觉伺服控制系统设计[J]. 计算机测量与控制, 2023, 31(09): 150-156+189.
- [56] 林菲, 邹玲, 张聪. 基于混合层次包围盒的快速碰撞检测算法[J]. 计算机仿真, 2023, 40(09): 454-457.
- [57] Dinas S, Bañón J M. A literature review of bounding volumes hierarchy focused on collision detection[J]. Ingeniería y competitividad, 2015, 17(1): 49-62.
- [58] Freire D L, de Carvalho A C, Peterlevitz A J, et al. A Methodology for Automated Conversion of Axis-Aligned to Polygonal and Oriented Bounding Box Annotations in Aerial Imagery Object Detection[C]//International Conference on Intelligent Data Engineering and Automated Learning. Cham: Springer Nature Switzerland, 2024: 373-385.
- [59] 谢斌斌. 基于包围盒的碰撞检测技术的研究[D]. 哈尔滨工程大学, 2015.
- [60] Galvan A F, Ramirez J D, Deshpande A D, et al. An Extended Virtual Proxy Haptic Algorithm for Dexterous Manipulation in Virtual Environments[C]//2023 IEEE World Haptics Conference (WHC). IEEE, 2023: 176-182.

攻读硕士期间发表的学术成果

- [1] 第一作者.基于 Unity3D 的打磨工艺虚拟化研究[J]. 安徽工程大学学报. 2025.
(已录用)
- [2] 第三作者. 混合 LM-GA 算法的人-机动作映射控制研究[J]. 安徽工程大学学报. 2025. (已录用)
- [3] 第二作者.基于虚拟现实的打磨机器人遥操作交互技术研究[J]. 安徽工程大学学报. 2025. (外审)
- [4] 第三发明人. 一种基于 Unity3D 的打磨工艺虚拟化仿真系统及方法[P]. 中国专利: CN202411445302.X..2024-10-17. (实审中)
- [5] 第四发明人. 一种基于混合 LM-GA 算法的人-机动作映射控制系统及方法:202411449770.4[P].2024-10-17.(实审中)

致谢

在校的这三年时间里很感谢老师们对我的淳淳教诲,是你们教会了我们勤奋学习,诚实做人,踏实做事,以宽容之心面对生活。指引着我们沿着正确方向前进。在点滴汇聚中使我逐渐形成正确、成熟的人生观、价值观。

首先特别要感谢我的指导老师,张振老师给予我很大的帮助,从论文的选题、研究计划的制定、技术路线的选择到系统的开发研制,各个方面都离不开张老师热情耐心的帮助和教导,每逢遇到困难,总会第一时间替我想办法解决困难,在这三年里,张老师认真的工作态度,强大的逻辑能力,以及诚信宽厚的为人处世态度,都给我留下了深刻的印象,也为我今后的工作树立了优秀的榜样。

感谢课题组的葛攀、张太盛、史钊楠、赵庆、王立园、梁益存以及我的室友徐晓辉、王荣香、汪杰。给我带来的生活上的快乐和学习上的极大帮助。

感谢我的父母,他们把我养育成人,从小教育我要踏踏实实地做人做事,给予我厚望,我会继续努力,在接下去的研究生阶段,我将继续努力,不会辜负你们的厚望。

最后,我要感谢我自己。感谢自己在硕士阶段的坚持和努力,感谢自己在遇到困难时不轻言放弃,感谢自己在追求梦想的道路上一一直勇往直前。这段经历让我成长了许多,也让我更加坚定了自己的信念和追求。

尚德敏学 唯实惟新

