arXiv:2601.01196v1 [cs.RO] 3 Jan 2026

EduSim-LLM：面向初学者、融合大型语言模型与机器人仿真的教育平台 *

Shenqi Lu
杭州电子科技大学信息工程学院 杭州，中国
229500317@hziee.edu.cn

Liangwei Zhang
杭州电子科技大学信息工程学院 杭州，中国
zhanglw@hziee.edu.cn

摘要

近年来，大语言模型（LLM）的快速发展显著提升了自然语言理解和人机交互能力，为机器人技术领域创造了新的机遇。然而，在人机交互和智能自动化行业快速发展的背景下，如何将自然语言理解融入机器人控制，仍是一项重要挑战。这一挑战阻碍了人们以直观方式控制复杂机器人系统，限制了这些系统在教育和实践中的普及。为此，我们推出了EduSim-LLM，这是一个将大语言模型与机器人仿真相结合的教育平台，并构建了一种语言驱动的控制模型，可将自然语言指令转换为可在 CoppeliaSim 中执行的机器人行为序列。我们设计了两种人机交互模式：直接控制和自主控制；基于多种语言模型开展系统性仿真，并评估多机器人协作、运动规划和操作能力。实验结果表明，大语言模型能够可靠地将自然语言转换为结构化的机器人动作；应用提示工程模板后，指令解析准确率显著提高；随着任务复杂度增加，在最高复杂度测试中的总体准确率超过 88.9%。

关键词 大语言模型（LLM） · 机器人仿真 · 自然语言接口 · 人机交互 · 教育机器人技术

1 引言

近年来，人机交互和人工智能的快速发展为直观的机器人控制带来了新的机遇。与基于编程的方法相比，自然语言控制平台提供了易于使用且用户友好的解决方案，使非专业人士能够有效操作机器人 [1]。Stein 等人 [2] 开发了一种基于浏览器的机器人仿真系统，并采用基于积木的界面，使初学者学生能够更轻松地学习机器人编程。Souza 等人 [3] 报告称，带有可视化积木编程功能的乐高套件常用于初学者的课堂机器人教育。然而，尽管取得了这些进展，语言驱动的控制仍面临局限：模型可能会误解复杂指令，并且可能无法在动态任务中实时运行 [4]。

现有基于语言的机器人控制研究通常侧重于将单步指令转换为针对单个机器人的操作。例如，Liang 等人 [5] 使用大语言模型（LLM）为抓取与放置任务生成 Python 代码，Ahn 等人 [6] 则使用 LLM 将指令映射到一组固定的移动操作技能。这些系统依赖预定义技能词汇或代码模板，针对的任务也相当简单。它们通常不支持多机器人协调或精细的机械臂控制。此外，如何解读或理解这些 LLM 生成的代码，在很大程度上仍未得到解决 [7]。

*引用：**Shenqi Lu 和 Liangwei Zhang**。《**EduSim-LLM：面向初学者的融合大语言模型与机器人仿真的教育平台**》。**2026年**

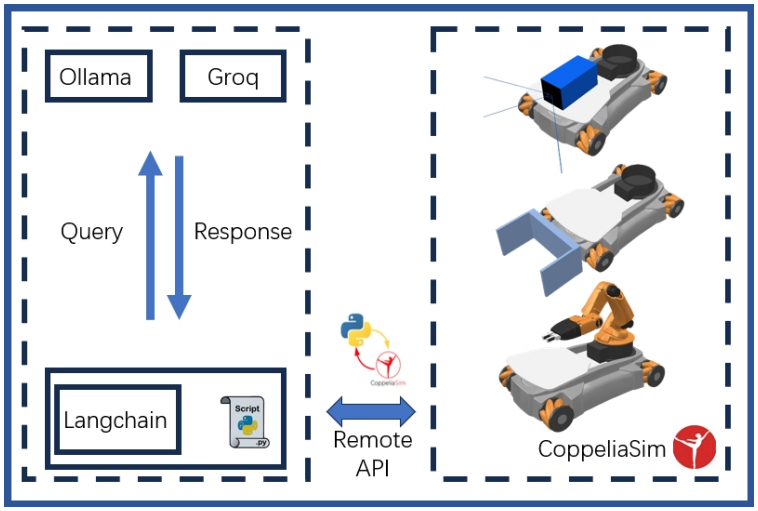

图 1：融合大语言模型与 CoppeliaSim 的 EduSim-LLM 系统架构

大多数现有教育机器人平台都没有将 LLM 与仿真环境紧密集成，其开发通常也需要投入大量编程工作。Gazebo 等高保真模拟器能够提供精确建模，但配置这些模拟器需要专业技术知识 [8]。相比之下，面向教育的模拟器通常更易于使用，但也存在实际限制，包括专有许可证和访问受限；例如，Robot Virtual Worlds [9] 和 VEX Virtual [10] 均需付费安装。因此，对于编程经验有限的学习者和教师而言，现有工具仍然难以获取。

为解决这些局限性，我们推出了 EduSim-LLM，这是一个将大语言模型与 CoppeliaSim 机器人仿真器相结合、用于基于自然语言的机器人控制的平台。EduSim-LLM 提供了便于新手使用的界面：用户只需用自然语言输入指令，即可观察仿真机器人执行相应操作，无需编写任何代码。系统支持两种交互模式：通过逐步下达指令进行直接操作，以及基于语言的自主操作。用户指令通过结构化提示词发送给大语言模型，其输出随后转换为 Python 控制代码，并在 CoppeliaSim 中执行。EduSim-LLM 支持多种机器人配置，包括由移动机器人组成的团队、机械臂，以及多个大语言模型后端。我们通过多机器人协作、移动路径跟踪和机械臂操作等具体任务，对 EduSim-LLM 进行了验证。基于数据集的评估表明，指令执行模型在简单任务、复合任务和复杂任务上的成功率分别为 100%、94.4% 和 88.9%。定量比较进一步表明，在复杂操作中，自然语言控制下的平均人工操作时间始终短于手动控制，在所有评估案例中，节省时间均超过 17.0 秒。我们的贡献如下：

(1) 我们设计了一个用于教育场景下自然语言机器人控制的零代码可视化平台。
(2) 我们构建了一个教育基准，用于评估不同指令复杂度级别下的表现。
本文结构如下：第 2 节介绍 EduSim-LLM 的系统框架和核心组件。第 3 节介绍实验设置、指令类别和评估方法。第 4 节总结全文。

2 提出的方法

2.1 设计概述

我们提出了 EduSim-LLM，这是一个集成框架，通过分层指令处理流水线连接自然语言理解与机器人仿真。其系统架构如图 1 所示，由四个模块组成：(1) 自然语言接口，(2) 基于大语言模型的指令规划器，(3) 仿真控制后端，以及 (4) 用户交互前端。

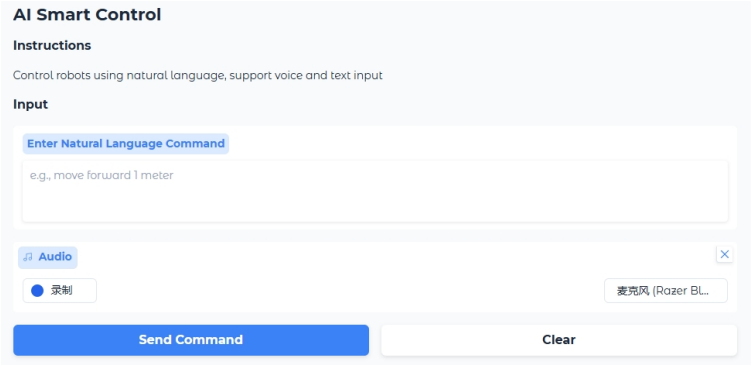

图 2：EduSim-LLM 的用户界面

2.2 自然语言界面

自然语言接口是 EduSim-LLM 系统的输入模块。如图 2 所示，它接收文本或语音形式的用户指令。输入内容包括单步、多步和多智能体指令。系统使用结构化提示模板，将自然语言转换为 Python 控制代码。该模块让用户无需编写代码即可操作机器人，并作为指令处理流水线的初始阶段。

2.3 基于大语言模型的指令规划器

该平台的核心是指令规划器，它利用 Groq 的 llama3-70b versatile 和 Ollama 的本地 llama3.1:8b 等大型语言模型。该规划器将用户指令转换为机器人可执行的动作原语，并管理多个机器人之间的条件逻辑。规划器借助 LangChain 实现及时调度和流式执行，从而支持模块化设计和后端互换。

2.4 仿真控制后端

该组件通过 CoppeliaSim 的远程 API 将高层策略与机器人执行连接起来。RobotController 类提供预定义的动作函数，用于移动（例如 moveForward、moveToXY、rotateToBeta）和操作（例如 moveArmSequential、presetFold）。executeActionCode 函数会将大语言模型生成的命令动态解析为一系列 Python 调用。后端负责初始化机器人句柄、跟踪位置和方向状态、实现包含超调检测的 7 阶段渐进减速，并维护轨迹历史以优化导航。

2.5 用户交互前端

前端使用 Gradio 实现，作为用户交互界面。如附录所示，用户可以在界面中输入命令、选择机器人实例、选择大语言模型后端，并执行控制流程。用户还可以查看解析后的代码、执行状态和机器人反馈。前端支持手动控制模块，包括移动、机械臂关节配置和摄像头预览。这些组件使用户无需编程即可提交命令并获得系统响应。

3 实验

3.1 实验设置

本实验旨在验证使用自然语言模型生成的机器人控制代码的可靠性及其效率提升。测试平台集成了 CoppeliaSim 仿真软件、基于 Python 的控制接口以及三种不同的 YouBot 机器人。YouBot1 配备视觉传感器，可实时传输图像；YouBot2 配备运输箱，用于清除障碍物；YouBot3 配备机械臂，用于抓取和取回物体。所有机器人均配备差速驱动底盘，可执行运动和操作任务。待评估的模型为 Groq 的 llama-3.3-70b-versatile。大语言模型生成的运动代码会直接在仿真环境中执行，并记录执行成功率等性能数据。

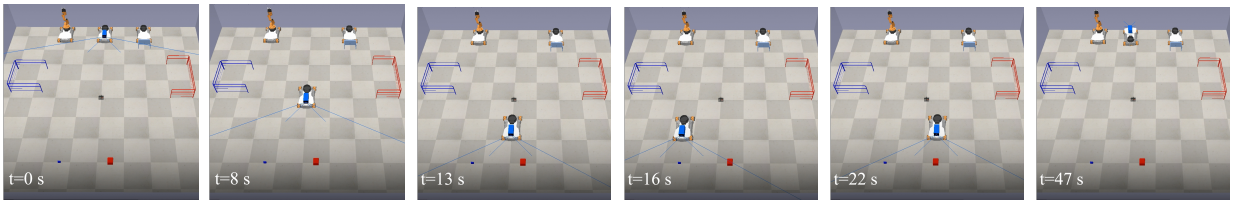

图 3：Youbot1 机器人首先在初始位置拍摄一张照片，然后朝负 y 轴方向移动，靠近目标物体后再拍摄一张照片；接着继续朝目标物体移动，然后沿负 x 轴方向横向移动，在到达待抓取物体之前拍摄一张照片；随后沿正 x 轴方向横向移动，在到达待移除的障碍物之前拍摄一张照片；最后结合 x 轴和 y 轴方向的移动，返回起点。

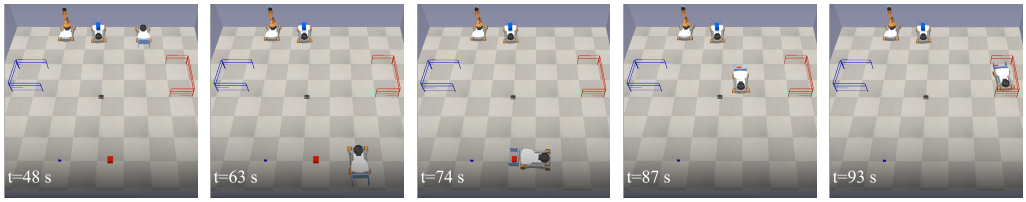

图 4：Youbot2 机器人从初始位置出发，沿负 y 轴方向移动至待移除的障碍物处。随后，它结合 x 轴和 y 轴方向的移动，将障碍物收集到机器人前方的固定框架中，并将其移至指定的停放区域。最后，它沿正 x 轴方向横向移动至停放区域，完成移除任务。

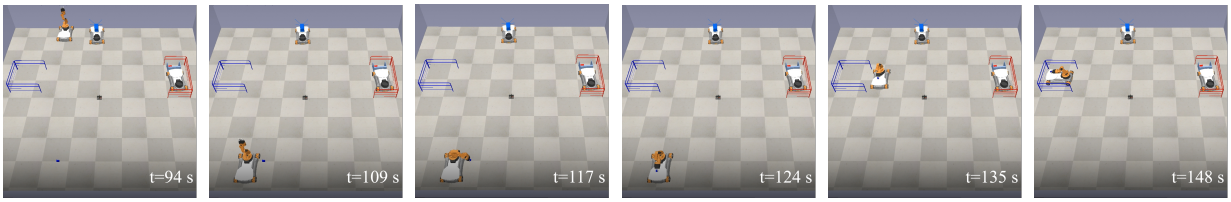

图5：Youbot3机器人从初始位置出发，沿y轴负方向移动至待夹取物体处。随后，它使用机械臂夹取并取回该物体。取回物体后，机器人沿y轴正方向移动至指定停放区域，最后沿x轴负方向进入停放区域，完成夹取任务。

3.2 基于数据集的实验

本实验评估大语言模型生成的机器人控制代码在不同指令难度等级下的表现。测试集包含 108 个实例，其中包括 36 条简单指令、36 条复合指令和 36 条复杂指令。每个实例由一条自然语言命令、大语言模型生成的 Python 控制代码以及一个难度标签组成。各难度等级的定义及实例数量见表1。

对于每条测试指令，executeActionCode函数都会在仿真器中执行生成的脚本，并记录执行是否成功以及每次试验的运行时间。对每组36个案例进行汇总后，简单、复合和复杂指令的成功率分别为100%、94.4%和88.9%，如图6（B）所示。随着指令难度从简单增加到复合和复杂，成功率逐渐下降。

在上述基于数据集的评估基础上，我们使用自然语言输入“所有机器人开始执行动作”开展了一项多机器人实验。系统通过提示模板将这段文本发送给大语言模型，模型将其分解为分别面向 YouBot1、YouBot2 和 YouBot3 的顺序动作代码。YouBot1 执行由 capturePhoto、moveToY 和 moveLateral 交替调用组成的九步动作序列，最后调用带有 xFirst 选项的 moveToXY 返回起点，从而完成四次图像采集及相应的位置调整子任务。YouBot2 执行三步动作序列：先沿 y 轴负方向通过 moveToY 移动，再通过带有 xFirst 选项的 moveToXY 运输障碍物，最后通过 moveLateral 横向移动至停放区域，从而完成沿运输路径的位置调整任务。YouBot3 执行六步动作序列，其中使用 moveToY 接近物体，使用 presetFold 和 presetExtend 配置

表1：实验分组与评估变量

难度	描述	总数	评估指标
简单	基本单步操作或包含 2 步的序列，涉及移动、机械臂预设或夹爪控制	36	成功率、失败原因
复合	结合 3-4 个步骤序列，整合导航与机械臂操作或多个方向性运动	36	成功率、失败原因
复杂	高级的 5-6 个步骤序列，协调集成多个系统，包括精确的机械臂定位、夹爪控制和导航	36	成功率、失败原因

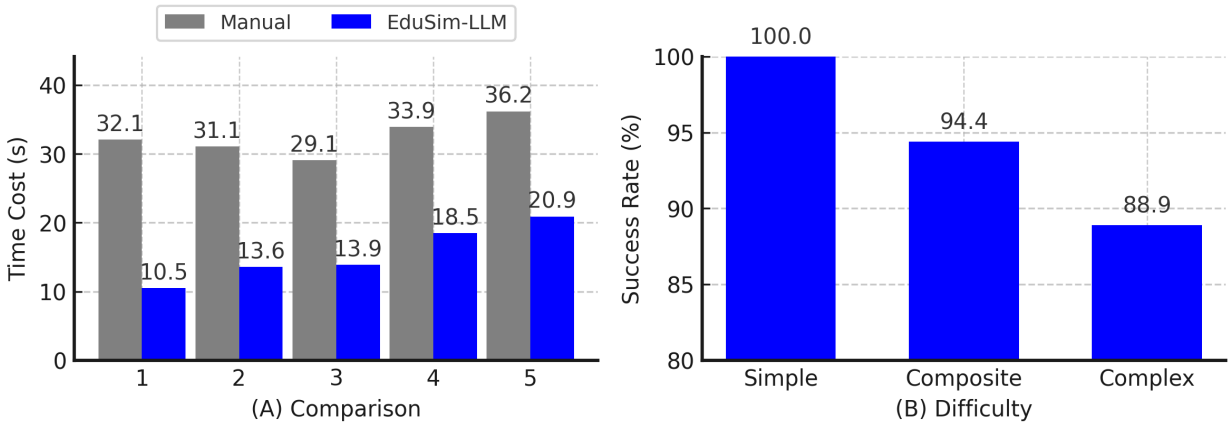

图6：（A）手动控制与自然语言控制的时间成本对比。在各项任务中，自然语言控制的平均执行延迟均低于手动控制，表明其交互效率更高。（B）不同指令难度等级下的执行成功率。每个难度类别均包含36个测试用例。尽管成功率会随着任务复杂度的增加而下降，但最低成功率仍为88.9%，体现了系统在不同指令类型下的稳健性能。

机械臂姿态，使用 closeGripper 执行抓取操作，通过 moveToY 沿 y 轴正方向运输物体，并在旋转后沿 x 轴通过 moveToXWithRotation 移动至停放区域，从而完成抓取物体并将其运输至目标位置的任务。图 3、图4 和图5 分别展示了 YouBot1、YouBot2 和 YouBot3 在此指令下的执行过程。

3.3 手动控制与自然语言控制的对比实验

本小节比较了在复杂任务中使用手动控制和自然语言控制的情况。从复杂任务组中选取五项任务，并在两种条件下各执行一次：手动控制，以及基于大语言模型生成的代码进行自然语言控制。每次试验的计时从用户开始输入命令时开始，到命令发送至仿真器时结束。记录的时间对应命令执行前的人工操作时间。

在手动控制条件下，操作员通过键盘和鼠标输入发出一系列低级控制命令，以指定移动底座、机械臂和相机采集的动作。在自然语言控制条件下，用户输入自然语言指令，系统通过提示模板将该指令发送给大语言模型，并将模型生成的 Python 控制代码作为一个整体发送到仿真器执行。对于这五项复杂任务，手动控制条件下的人工操作时间在所有任务中均超过 29 秒，而自然语言控制条件下的人工操作时间在所有任务中均少于 21 秒，如图 6（A）所示。对于全部五项任务，自然语言控制所需的人工操作时间均少于手动控制。这一对比结果表明，在预先构建控制函数库和提示模板的情况下，自然语言控制能够缩短用户为复杂机器人操作准备和发送命令所花费的时间。

4 结论

本研究提出了 EduSim-LLM，这是一种将大语言模型与机器人仿真相结合、以实现基于自然语言的机器人控制的教育平台。其系统架构由结构化指令界面、大语言模型规划器、仿真控制后端和面向用户的前端组成。通过对三类机器人以及 108 个复杂程度各异的指令案例进行实验评估，结果表明，大语言模型能够有效地将自然语言命令转换为可执行的控制代码。系统在简单任务、复合任务和复杂任务中的成功率分别达到 100%、94.4% 和 88.9%。多机器人协作实验进一步验证了模型为多个智能体生成序列动作的能力。与手动控制的对比实验表明，在复杂任务中，自然语言控制显著缩短了用户的操作时间。结构化提示词与预先构建的函数库相结合，提高了解析的可靠性和系统响应速度。这些发现证实了在仿真环境中使用大语言模型进行教育机器人控制的可行性和效率。

参考文献

[1] Yate Ge、Yi Dai、Run Shan、Kechun Li、Yuanda Hu 和 Xiaohua Sun。Cocobo：探索将大语言模型用作面向终端用户的机器人编程引擎。载于*2024*年 *IEEE*视觉语言与以人为中心的计算研讨会（*VL/HCC*），第89–95页。IEEE，2024年。[2] Gordon Stein、Devin Jean、Saman Kittani、Menton Deweese 和 Ákos Lédeczi。面向新手且易于使用的网络化教育机器人仿真平台。教育科学，15(2):198，2025年。[3] Isabelle ML Souza、Wilkerson L Andrade、Lívia MR Sampaio 和 Ana Liz Souto O Araujo。乐高®机器人技术在教育中的应用：系统综述。载于*2018*年 *IEEE* 教育前沿会议（*FIE*），第1–9页。IEEE，2018年。[4] Zhao Yu、Peize Zhang 和 Jing Shi。自然语言模型推动工业机器人技术转型：最新进展与未来展望。机器人与计算机集成制造，97:103113，2026年。[5] Jacky Liang、Wenlong Huang、Fei Xia、Peng Xu、Karol Hausman、Brian Ichter、Pete Florence 和 Andy Zeng。代码即策略：用于具身控制的语言模型程序。*arXiv*预印本 *arXiv:2209.07753*，2022年。[6] Michael Ahn、Anthony Brohan、Noah Brown、Yevgen Chebotar、Omar Cortes、Byron David、Chelsea Finn、Chuyuan Fu、Keerthana Gopalakrishnan、Karol Hausman 等。按我能做的去做，而非按我说的去做：将语言落地于机器人可供性。*arXiv*预印本 *arXiv:2204.01691*，2022年。[7] Peng Shu、Huaqin Zhao、Hanqi Jiang、Yiwei Li、Shaochen Xu、Yi Pan、Zihao Wu、Zhengliang Liu、Guoyu Lu、Le Guan 等。用于编程与机器人教育的大语言模型。*arXiv*预印本 *arXiv:2402.06116*，2024年。[8] Owen Gervais 和 Therese Patrosio。通过乐高SPIKE Prime开展ROS和Gazebo入门教学。载于国际机器人教育会议（*RiE*），第201–209页。Springer，2021年。[9] Sharon Mistretta。混合式教学与学习中的虚拟机器人。电子学习新进展，145，2022年。[10] VEX Robotics。VEXcode虚拟机器人（VR），2022年。

用户交互前端图片附录

图 7–10 展示了 EduSim-LLM 的前端界面，用于命令输入和执行监控，以及对移动底座、机械臂/夹爪进行手动控制和基于视觉的图像采集。

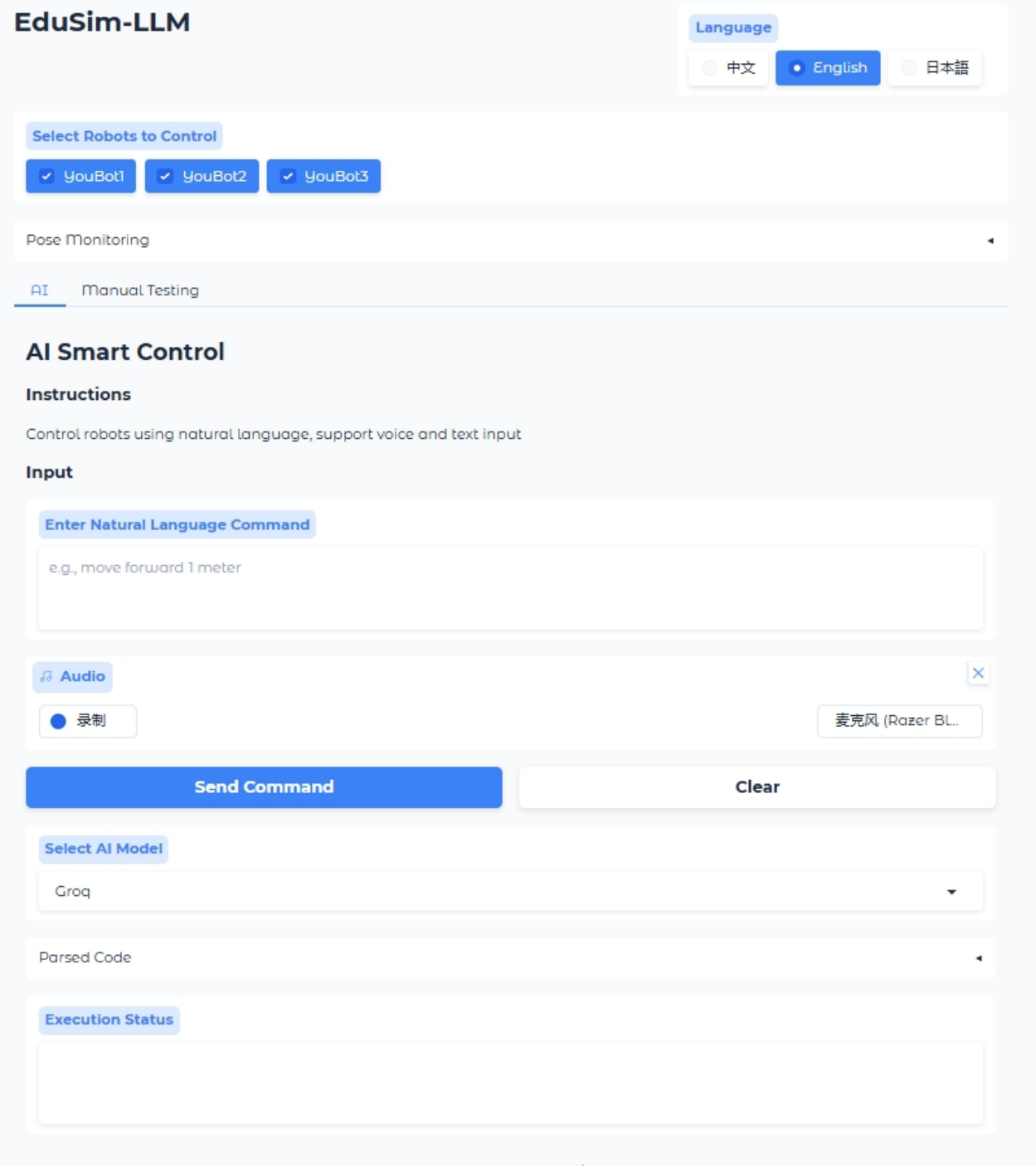

图 7：自然语言智能控制仪表板

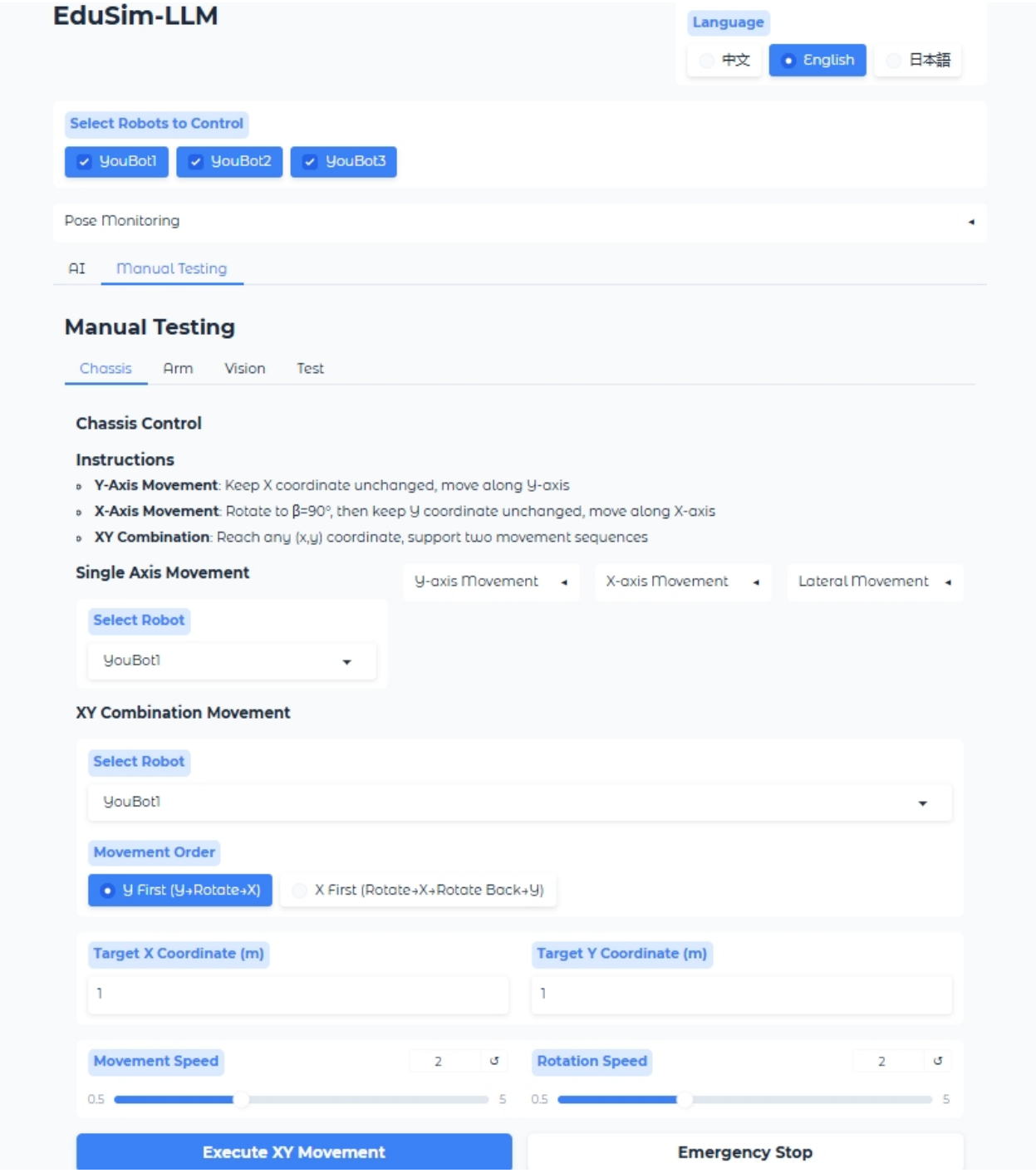

图8：移动底座手动运动测试界面

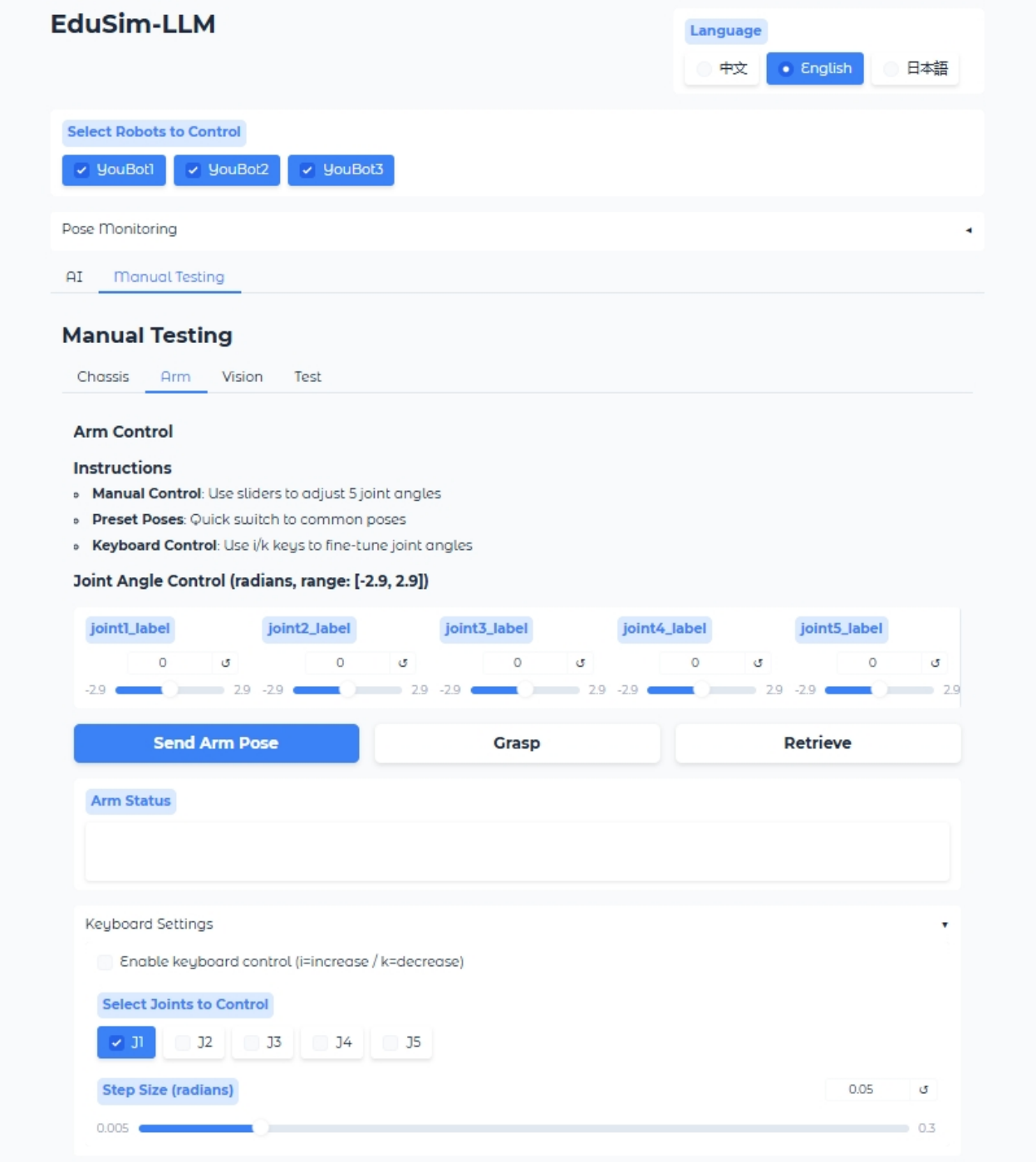

图 9：机械臂和夹爪手动控制界面

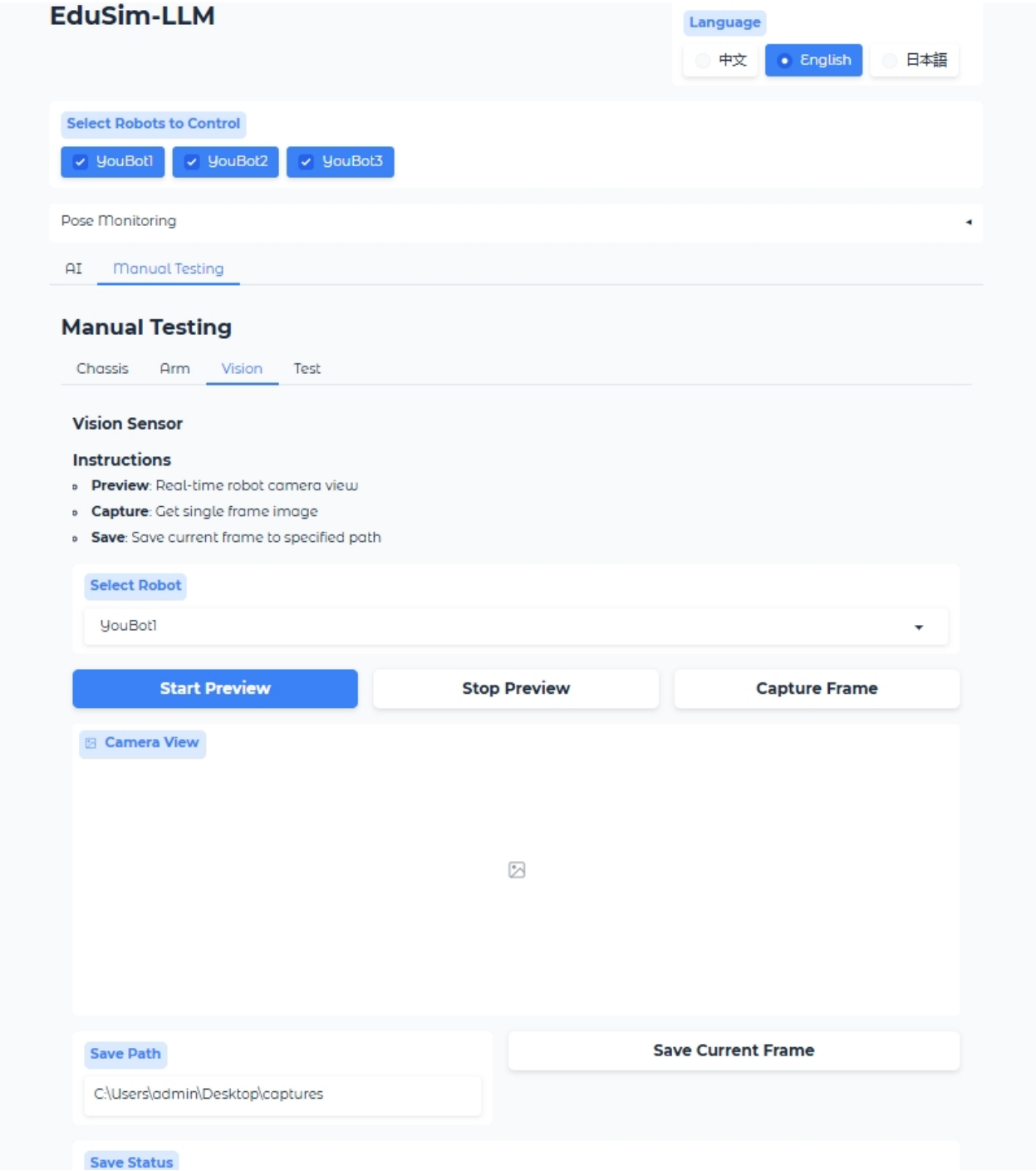

图 10：视觉传感器与图像采集界面