Loco-Manipulation for Quadruped Robot: Deploying RL Policies on Unitree Go2

Steph Li, Zhizhen Zhang & Supervisors: Dr Yadan Luo, Prof Helen Huang

UQMM Lab, School of Electrical Engineering and Computer Science

Scan me for details →

Background & Motivation

- ❑ Most systems **separate walking and arm** use on quadrupeds.
- ❑ **RL** can learn a single **full-body loco-manipulation** controller.
- ❑ Sim-trained policies often fail on real robots (latency, noise, dynamics).
- ❑ Our goal is **robust embodied AI** that works reliably in real, messy environments.

Figure 1: Loco-manipulation in practice: VR-based teleoperation of the Go2 with SO-100/101 arm in UQMM lab.

Project Aim & Contributions

- ➢ **Aim**: Deploy and test a unified RL loco-manipulation controller on Unitree Go2 with VR control.
- ➢ Adapt **PPO policies** from simulation to real hardware via Unitree SDK2 on Jetson Orin.
- ➢ Build the **VR/WebRTC pipeline** and benchmark walking + manipulation tasks in sim vs real.

System Architecture & Control Stack

End-to-end VR-RL control pipeline

- ❖ **Meta Quest VR** streams head pose, joystick and hand input via **WebRTC/WebSocket**.
- ❖ **PhosphoBot (Jetson Orin)** fuses VR input with RL policy output at ~50 Hz for gait + arm control.
- ❖ **Hybrid PPO + safety guards** (limits, contact checks, E-stop) keep behaviour stable.
- ❖ **Context + wrist cameras** give real-time visual feedback for precise manipulation.

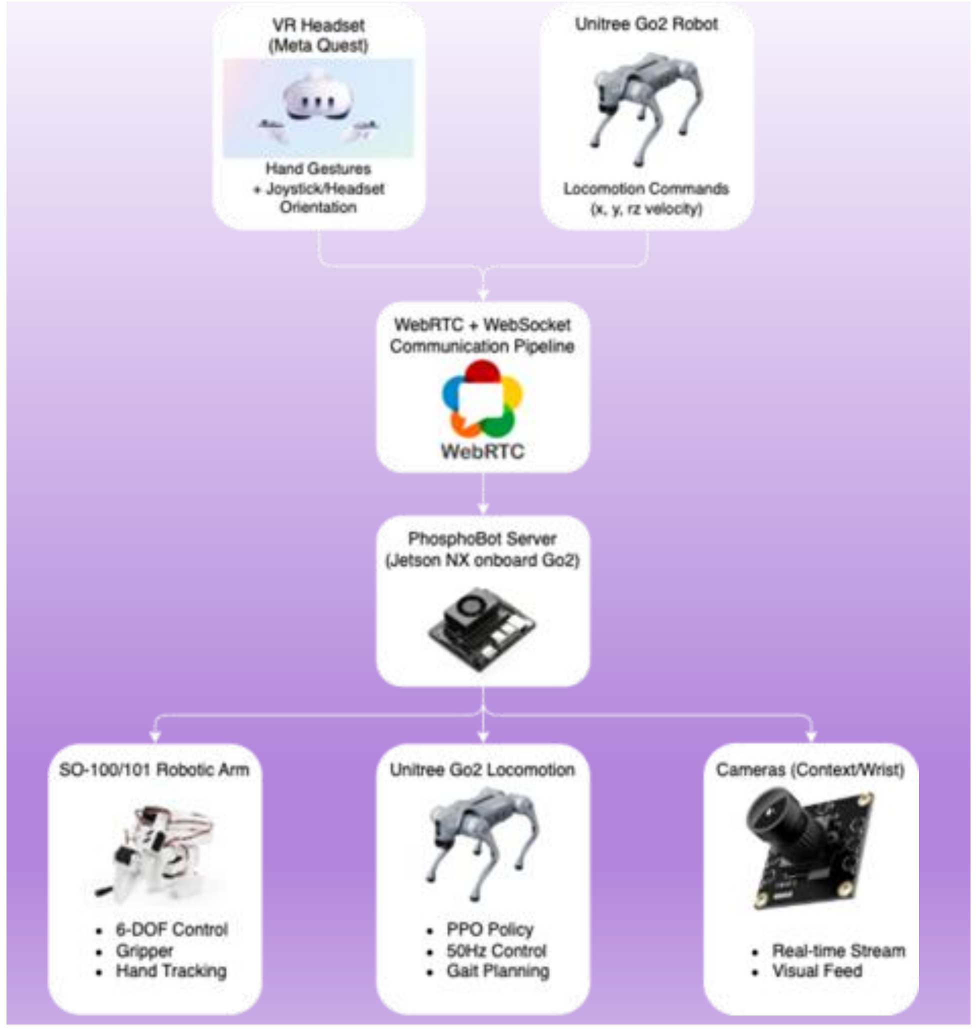

Figure 2: VR/WebRTC loco-manipulation pipeline from Meta Quest to the Jetson-based PhosphoBot server and Unitree Go2 locomotion + SO-100/101 arm + cameras.

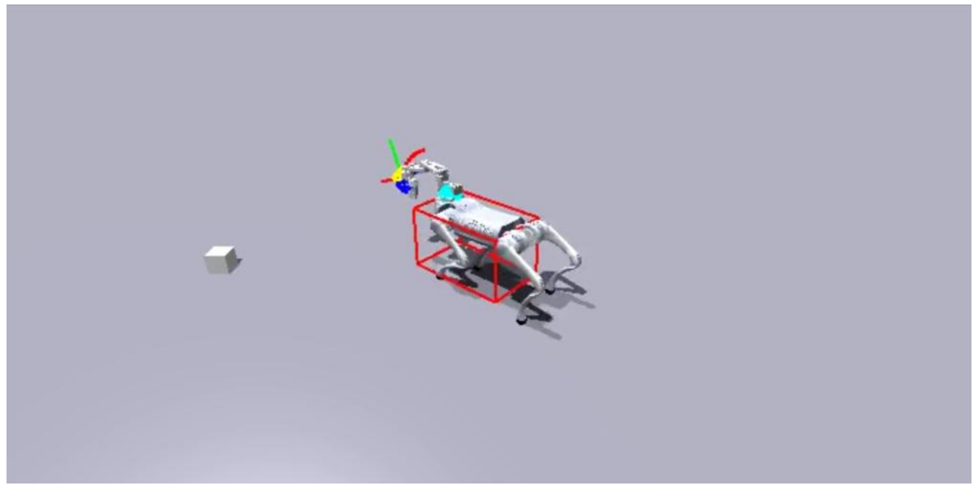

Figure 3: Loco-manipulation in practice: PPO loco-manipulation policy running in simulation.

Key Technical Challenges

Synchronisation, sensing & Sim2Real gap

- ○ **Action synchronisation**: arm and legs must stay tightly timed - unsynchronised actions cause wobble or instability.
- ○ **Observation complexity**: high-dimensional proprioception + vision + VR commands, all latency-sensitive.
- ○ **Generalisability**: sim-trained policies can overfit to ideal dynamics and struggle with hardware delays, noise and friction.

Results, Limitations & Future Work

- ✔ **End-to-end stack**: VR → WebRTC → Jetson Orin → Unitree SDK2 → Go2 + arm.
- ✔ Policies are **stable in sim**, but less stable on hardware (Sim2Real gap).
- ✔ **Limits**: sensitive to delay, calibration errors and contact modelling.
- ✔ **Next**: more domain randomisation, better arm–leg coordination, latency-aware control, harder test scenarios.

Follow us on Instagram
↓

steph.li@uq.edu.au
linkedin.com/in/steph-li-9779b6162/
instagram.com/uqmm_lab/
luoyadan.github.io/uqmm/

School of Electrical Engineering and Computer Science

Acknowledgements

I would like to thank:
- Dr Yadan Luo - academic supervision and feedback
- Mr Zhizhen Zhang - technical mentoring on RL deployment
- UQMM Lab - hardware support with Go2 and robotic arm
- UQ Employability - Winter Research Program coordination

UQMM LAB

THE UNIVERSITY OF QUEENSLAND
AUSTRALIA
CREATE CHANGE