

# From Remote Control to Vision-Language-Action: Initial Deployment of the Humanoid Robot Unitree G1 with Tactile Hands

**Branislav Zigo**

Faculty of Mathematics, Physics and Informatics, Comenius University in Bratislava, Slovakia  
branislav.zigo@fmph.uniba.sk

## Abstract

This paper reports initial deployment experience with two Unitree G1 EDU U6 humanoid robots equipped with Inspire RH56DFTP tactile five-finger hands in a university research-and-education lab. We document the path from unboxing to repeatable research use: commissioning, remote control and teleoperation, locomotion, SDK/ROS 2 configuration, integration of third-party dexterous hands, simulation-based de-risking, and cybersecurity hardening. Despite rapidly growing research demonstrations, detailed operator-facing notes on real G1 manipulation deployments remain limited; we aim to narrow this gap with a practical, system-level report. From a hardware perspective the platform largely fulfils expectations and flat-ground locomotion is convincing; nevertheless, the complete system is still fragile for everyday experimental work, especially once the dexterous hands are installed. The main friction points are software version drift and weak configuration management across vendor components, mechanical vulnerability of the external hand cabling during loco-manipulation, and the need to build a unified sensorimotor pipeline that incorporates tactile perception beyond vendor-provided hand motion primitives. We also discuss practical constraints of the onboard sensing stack (camera viewpoint and the current maturity of SLAM/navigation tooling) and summarize defensive measures required when operating a network-connected humanoid on institutional infrastructure. The paper concludes with lessons learned and a roadmap toward early Vision-Language-Action experiments with UniFoLM and GR00T.

## 1 Introduction

The recent availability of comparatively accessible humanoid platforms has shifted a recurring question in embodied AI from "when will labs get a humanoid?" to "what happens after the robot arrives?". Official material around the Unitree G1 emphasizes agility, optional secondary development, force-controlled dexterous hands, and the broader UnifoLM ecosystem [1], [13], [19]. At the same time, the research community is rapidly moving toward integrated Vision-Language-Action (VLA) and humanoid foundation models, including vendor-specific efforts such as Unitree UnifoLM-VLA-0 and more general systems such as NVIDIA GR00T N1 [14], [15], [23]. Recent G1 results already span perceptive locomotion, reduced-order MPC (Model Predictive Control) walking, single-demonstration loco-manipulation, unified whole-body control, visual sim-to-real transfer, and teleoperated medical manipulation [34]-[39]. This makes the G1 especially interesting as a bridge platform: it is accessible enough to be seriously considered by university laboratories, yet advanced enough to expose the real systems-integration burden behind contemporary humanoid-robotics research.

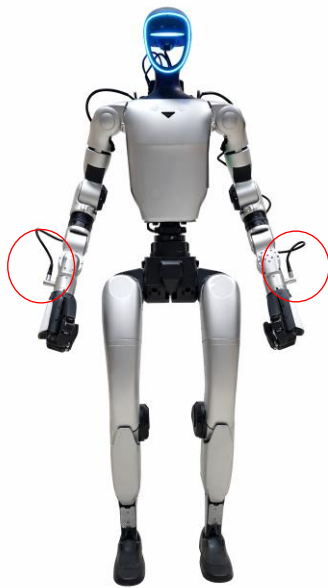
This paper is intentionally written as an early deployment note. It does not claim state-of-the-art task performance; instead, it documents first practical experience from real-life testing of two Unitree G1 EDU U6 robots equipped with Inspire RH56DFTP tactile hands in a research-and-education setting. Our goal is to summarize the transition

from remote control toward software-defined autonomy and, ultimately, VLA-style control. The main contribution is therefore not a new control algorithm, but an honest account of where the platform already works well, where it remains brittle, and which engineering gaps appear when one moves from promotional videos to repeatable laboratory use. While the academic literature is rapidly growing, detailed and reproducible platform-engineering reports for real manipulation deployments (especially with dexterous hands) are still relatively scarce. In contrast to many published G1 demonstrations, which rely on custom research stacks, we focus deliberately on developer workflow, configuration management, and the promise-versus-reality gap encountered during first deployment [34]-[39].

## 2 Platform overview and official parameters

Officially, the G1 EDU has standing dimensions of 1320 x 450 x 200 mm, a folded size of 690 x 450 x 300 mm, a weight of about 35 kg+, and a total degree-of-freedom range of 23-43 depending on configuration [1]. The official parameter list also states up to 120 N·m knee torque, about 3 kg arm load, depth camera plus 3D LiDAR sensing, Wi-Fi 6, Bluetooth 5.2, about 2 h endurance, a 5 W speaker, and support for secondary development in the EDU edition [1], [19]. Unitree's store documentation further makes explicit that secondary development is reserved for the EDU edition rather than the base consumer configuration [19].

For the hands, the RH56DFTP user manual specifies 12 joints, 6 DoF, 6 force sensors with 0.1 N resolution, 5-17 tactile sensors distributed across fingertips, finger pads, and the palm, repeated fingertip positioning accuracy of  $\pm 0.2$  mm, and five-finger grip force of 30 N [2]. The hand operates at 24 V and exposes Ethernet, RS485, and CAN2.0 interfaces [2]. In our case, this creates a composite research platform: a Unitree humanoid body combined with third-party tactile hands, rather than a single vertically integrated end-to-end system.



**Fig. 1.** Humanoid Unitree G1 EDU U6 with dexterous tactile hands RH56DFTP. Problematic external cable connection to hand containing power supply and control wires is marked by red ellipses.

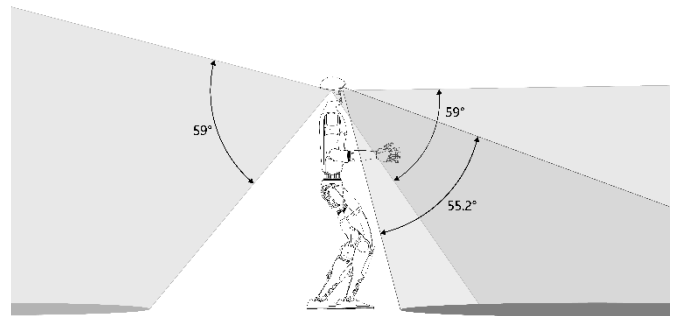
### 3 Early deployment experience

#### 3.1 Commissioning, remote control, and first motion

In our early use, unpacking and basic installation were not the main problem. The robot can be brought to first motion relatively quickly if one stays close to the vendor procedure and treats the first sessions as acceptance testing rather than free exploration. The dedicated handheld controller was the most dependable interface for first commissioning. The mobile app is useful for calibration, status inspection, and service access [3], but pairing was not consistently frictionless in our tests and should be treated as an auxiliary interface rather than the primary

operational one. A practical lesson is that the shortest path to safe initial use is controller first, app second, SDK third. In our case, both units were delivered within about two months from order, which is positive for academic project planning.

A second lesson concerns stair locomotion. Official demonstrations understandably highlight challenging motions, but stand-alone stair traversal should not be used as an early acceptance test in a normal laboratory environment. In our view, it remains a high-risk activity during initial deployment and should be attempted only under controlled conditions with close supervision. By contrast, ordinary flat-ground walking and running were immediately among the strongest parts of the platform. This local impression is consistent with the broader G1 literature, which already reports robust terrain locomotion and strong hardware walking performance under specialized controllers [34], [35]. In our experience, stair traversal remained unreliable and is generally not recommended during early deployment.



**Fig. 2.** Vertical range of view of RGB-D camera is limited and robot cannot see faces of humans in front of him. Source: [40]

#### 3.2 Locomotion is stronger than manipulation

Walking and running were the most convincing robot skills in our first tests. This should, however, be interpreted carefully when manipulation is discussed. A large fraction of online promotional media features terminal covers or passive, rubber-like prosthetic hands that tolerate incidental floor contact and reduce distal mass. That setup is not equivalent to a G1 equipped with research-grade tactile five-finger hands. RH56DFTP hands are heavier, significantly more expensive, and more fragile despite their robust appearance. In addition, the external power/data cable routed from the neck switch to the hand connector is mechanically fragile and can be stressed during manipulation and locomotion; reliable strain relief and careful cable routing are mandatory (Fig. 1). Once installed, they immediately change the practical balance envelope, acceptable collisions, floor-contact assumptions, and repair anxiety of the whole system. In

short, locomotion marketing should not be naively extrapolated to locomanipulation research.

This distinction also clarifies why the same robot can look simultaneously impressive and limited. As a locomotion platform, G1 already feels mature enough for meaningful experimentation. As a manipulation platform with tactile five-finger hands, it is promising but still clearly in the systems-integration phase. The published literature supports this mixed picture: impressive real-robot locomanipulation, visual sim-to-real, and teleoperation results do exist, but they typically rely on custom whole-body controllers, retargeting pipelines, specialized teleoperation setups, or highly tuned learning stacks rather than a stock out-of-the-box workflow [36]-[39]. Cameras were among the less problematic subsystems in our early experiments, whereas the speaker was notably weaker than one would prefer for demonstrations in a noisier environment and required additional software attention.

### 3.3 Software stack and configuration management

Here the platform reveals its largest weakness: software configuration management and version control across the vendor stack. The official development path combines SDK2, SDK2 Python, ROS 2 Foxy, CycloneDDS, and multiple Ubuntu/ROS environments [4]-[6]. In practice this is not a plug-and-play experience. Parts of the stack must be built from source, environment variables such as `CYCLONEDDS_HOME` and `CMAKE_PREFIX_PATH` must be aligned, and behavior can change depending on controller/app modes and firmware versions (e.g., entering debug mode can disable higher-level motion services) [7], [8]. A recurring problem is documentation and sample code that refer to different (sometimes implicit) versions of firmware, middleware, and libraries, which creates unnecessary churn, hard-to-reproduce bugs, and slows down development.

From a research perspective the platform is therefore only partially open. In our deployment, the auxiliary PC2 side provided an accessible Linux development target with Jetson Orin-class compute [9], while the main vendor-managed PC1 side remained closed. This hybrid openness is useful but incomplete: the robot is not a single, clean, fully transparent software artifact. Without a robust producer-side versioning and release process (source, binaries, firmware, and documentation), the integration burden tends to increase as new hardware variants and software branches accumulate. In our experience, even minor upstream changes can break portability of developed software, datasets, and trained policies between otherwise similar robots, turning configuration reproducibility into a first-order engineering task.

### 3.4 Power management and development workflow

The supplied Li-ion batteries provide roughly two hours of operation per pack in our typical lab sessions, and we observed fast charging on the order of tens of minutes (approximately 25 minutes with the supplied charger). We procured two batteries per robot and recommend this as a baseline for uninterrupted development and demonstrations. Importantly, the robot consumes non-trivial power even while standing, therefore hanging the robot on its rack during software development and testing is strongly recommended not only for safety, but also for maximizing effective uptime.

### 3.5 Sensors, perception, and navigation tooling

The LiDAR coverage is satisfactory for basic obstacle perception, but the default RGB-D camera viewpoint is constrained by the robot's head geometry and mounting angle. As illustrated in Fig. 2, the camera primarily observes the manipulation workspace and the ground in front of the robot and cannot reliably capture faces of standing humans at conversational distance. For human-robot interaction or safety monitoring, mounting an additional camera is therefore strongly recommended. In our units, free USB-C ports are available for such extensions. Finally, the SLAM/navigation components provided in the SDK are currently rudimentary for real locomotion tasks; more mature mapping and navigation solutions should be integrated for repeatable autonomous operation in non-trivial environments.

### 3.6 Hand subsystem integration in practice

The RH56DFTP hands are supported for basic motion generation through the vendor stack, but tactile perception is not integrated into the whole-body control pipeline out of the box. Closing the loop from tactile/force sensing to grasp stabilization and bimanual coordination therefore requires additional middleware and application-layer development.



Fig. 3. Dexterous hands RH56DFTP

## 4 From remote control to VLA

RH56DFTP is attractive precisely because it combines actuation, force sensing, and tactile sensing in one commercial hand [2]. Yet on G1 this does not automatically translate into a unified sensorimotor API. In

our workflow, the body and the hands had to be treated as partially separate subsystems. This matters for contact-rich tasks: grasp closure, grasp force, tactile-state acquisition, and arm-posture synchronization do not automatically arrive as one coherent research stack. As a result, early locomanipulation is less limited by kinematics than by interface fragmentation, middleware engineering, and safety constraints. Published G1 manipulation studies point in the same direction: strong performance is possible, but it is typically achieved with custom teleoperation interfaces, grasp parameterizations, or whole-body control stacks rather than with a single unified vendor API [36], [39].

This is also the point where tactile sensing becomes scientifically interesting. The hands promise richer feedback for grasp stability, slip, and contact events, but the practical pipeline from raw tactile/force data to a closed-loop whole-body behavior remains to be built. In our view, the body and the hands are not yet a seamless single developer platform; they are a promising multi-vendor assembly that requires explicit fusion work. That is precisely why they are interesting for academic research.

#### 4.1 Simulation with Isaac Sim and Isaac Lab

To reduce risk, we used Isaac Sim and Isaac Lab as the main simulation route. NVIDIA describes Isaac Sim as a USD-centered (USD – Universal Scene Description), GPU-accelerated robotics simulation platform for designing, tuning, training, and deploying control agents [10]. Isaac Lab builds on top of Isaac Sim as an open and modular framework for reinforcement learning, learning from demonstrations, and motion planning [12]. Current NVIDIA guidance lists at least 32 GB RAM and an RTX 3070-class GPU for Isaac Sim, with higher RAM and VRAM recommended for heavier workloads and Isaac Lab training [11]. This matches our practical impression: simulation is viable and valuable, but it is not lightweight. The main benefit is not visual polish; it is the ability to de-risk locomotion, arm motion, and data collection before touching fragile hardware.



Fig. 4. Unitree G1 in Isaac Lab. Source: [41]

#### 4.2 VLA integration paths: UnifoLM, GR00T, and external tooling

From the perspective of embodied AI, the most interesting shift is from remote control to VLA. Unitree now publicly exposes an embodied-AI track through its official open-source portal [13]. Its current UniFoLM track includes UnifoLM-VLA-0, released with training code, inference code, checkpoints, and associated G1 manipulation datasets [14], [21]. Unitree describes the model as a humanoid-manipulation VLA that extends a vision-language backbone through continued pre-training on robot-manipulation data, aiming to turn a VLM into an "embodied brain" with stronger spatial semantics and physical common sense [13], [14]. The public repository further states that the model can complete 12 categories of complex manipulation tasks in real-robot validation using a single policy [14]. For our purposes, the key point is not only the headline claim, but the existence of a vendor-supported path from language-conditioned perception to executable humanoid manipulation policy.

NVIDIA's GR00T should be interpreted differently. GR00T N1 is presented as an open cross-embodiment humanoid foundation model rather than a Unitree-specific stack [15], [22], [23]. Its published architecture explicitly separates a slower vision-language reasoning module (System 2) from a faster action-generation module (System 1) based on a diffusion transformer [15]. NVIDIA further positions Isaac GR00T as a broader development platform that combines foundation models, simulation, and synthetic-data pipelines for humanoid research [23], [24]. This is attractive for humanoid research because it frames adaptation not as training from scratch, but as post-training a broad prior with embodiment- and task-specific data.

Hugging Face LeRobot occupies a third role. It is neither a vendor stack nor a single VLA model, but a practical open-source layer that standardizes datasets, policies, simulation, and hardware interfaces [16], [25], [26]. Current documentation explicitly supports Unitree G1, including teleoperation, simulation, locomanipulation, and policy training, while exposing a common framework for community policies and deployment workflows [16], [25]. The LeRobotDataset v3.0 specification is especially relevant because it unifies multimodal time-series data, multi-camera video, and rich metadata in a format that can be streamed directly from the Hugging Face Hub [26]. In practical terms, LeRobot looks less like a final scientific claim and more like an integration substrate on which a G1 plus RH56DFTP research workflow can realistically be built.

For G1 EDU U6 equipped with RH56DFTP, however, none of these paths is plug-and-play. The core obstacle is embodiment mismatch. Public G1 examples often assume

default hands, simplified end-effectors, or a different action space than a third-party tactile five-finger hand. Any serious VLA deployment will therefore require explicit redesign of observation vectors, finger-level action parameterization, control frequency, middleware bridges, and synchronized logging. In that sense, the path from remote control to VLA is real, but it passes through systems integration rather than model download alone.

### 4.3 Candidate datasets for Unitree dexterous manipulation training

Usable training data for Unitree dexterous manipulation can be divided into three tiers. The first tier is embodiment-near data. Unitree now publicizes G1 dexterity-hand manipulation datasets, G1 gripper manipulation datasets, Z1 dual-arm datasets, and an imitation-learning framework based on LeRobot [13], [20]. In addition, its current UnifoLM whole-body teleoperation collection already includes G1 whole-body tasks with Inspire hands, such as collecting clothes, picking up a pillow, and placing clothes into a washing machine or basket [21]. These datasets are the most directly relevant starting point because robot morphology, control interfaces, and task kinematics are close to the target platform.

The second tier is cross-embodiment robot-learning data useful for pre-training and transfer, even if it is not directly executable on G1 plus RH56DFTP. Open X-Embodiment aggregates data from 22 robots and 527 skills [27]. DROID adds a large in-the-wild corpus of 76k manipulation trajectories collected across 564 scenes [28]. RH20T is especially interesting for contact-rich manipulation because it provides more than 110k real-world robot sequences with visual, force, audio, and action signals, together with paired human demonstrations and language descriptions [29]. For a humanoid dexterous-hand project, such datasets are best viewed as sources of visual-language-manipulation priors, subtask structure, and robustness, not as drop-in finger-control supervision.

The third tier comprises hand-object interaction datasets that are most useful for perception, retargeting, grasp priors, and contact modeling. DexYCB focuses on hand grasping of objects and benchmarks 6D object pose, 3D hand pose, and handover-relevant grasp generation [30]. HOI4D provides 2.4 million RGB-D egocentric frames with hand pose, object pose, action, and scene annotations [31]. ARCTIC concentrates on dexterous bimanual hand-object manipulation with articulated objects and dynamic contact over 2.1 million frames [32]. GRAB adds whole-body grasping sequences with contact information, hand pose, object pose, and full-body motion [33]. These are not end-to-end Unitree policy datasets, but they are highly relevant for building the perception and retargeting layers that dexterous humanoid manipulation still lacks.

Consequently, the most credible data strategy for our platform is layered: start from vendor- and embodiment-near Unitree datasets, add laboratory teleoperation data collected on the actual RH56DFTP-equipped robots, and then use large cross-embodiment corpora plus hand-object datasets for pre-training, perception, and synthetic-data augmentation. This hybrid strategy is more realistic than expecting one open dataset to solve whole-body humanoid dexterous manipulation end-to-end.

## 5 Cybersecurity and operational deployment implications

Humanoid deployment is not only a robotics problem but also a networked-systems problem. In a university environment, the robot is routinely connected to developer laptops, institutional networks, external cameras, and storage. This enlarges the attack surface and turns the humanoid into a high-value cyber-physical endpoint that can influence both information security and physical safety. A recent SoK (Systematization of Knowledge) on humanoid cybersecurity evaluates G1 EDU among real platforms and argues that humanoids require a layered security model spanning hardware, operating system, middleware, cloud services, and human-robot interaction [17]. In addition, a dedicated G1 security assessment highlights weaknesses in provisioning, cryptographic material reuse, and telemetry behavior that motivate a conservative operational stance for laboratory deployments [18].

### 5.1 Reported attack surfaces and data flows

The G1 exposes multiple concurrent communication surfaces: local Ethernet/Wi-Fi networking used for SDK and ROS 2 operation, Bluetooth Low Energy (BLE) used for provisioning and mobile-app interaction, and additional runtime services for media streaming and cloud connectivity. On the local network, DDS/RTPS traffic enables high-rate publish/subscribe messaging between components and external development machines; without additional protections, this traffic can be observed or interfered with by any device on the same segment. In parallel, cloud-facing channels may be used for telemetry, over-the-air updates, and remote services. For laboratories, the practical implication is that physical isolation alone is insufficient: the robot must be treated like a managed endpoint with explicit network policy and monitoring.

### 5.2 Key issues highlighted in G1 security analyses

Mayoral-Vilches et al. report that the BLE-based provisioning workflow can be abused to obtain privileged code execution when an attacker is in Bluetooth range, and that the corresponding cryptographic material is reused across devices, reducing effective security [18]. The same assessment documents multiple active communication channels (e.g., local DDS/RTPS buses, cloud

telemetry/OTA coordination, and media streaming services) and reports weaknesses in the proprietary configuration protection scheme [18]. In particular, it describes periodic transmission of detailed robot state and system telemetry to external servers and notes implementation choices that may weaken transport-layer protections in some clients [18]. While the exact impact depends on firmware version and deployment configuration, these results justify a conservative operational stance: assume that (i) provisioning interfaces can be a compromise vector, (ii) default network services can leak sensitive sensor and state information unless explicitly constrained, and (iii) security posture can change across software/firmware updates and should be revalidated after each upgrade.

### 5.3 Practical mitigations for academic deployments

Based on our operational needs and the above threat model, we recommend a defensive baseline for laboratory use: (1) place the robot on a dedicated VLAN or behind a small router with explicit egress filtering; (2) disable Bluetooth and unnecessary services when not required for experiments; (3) implement a default-deny outbound firewall policy and allow only explicitly needed endpoints (e.g., time synchronization, approved model APIs) while blocking telemetry/OTA paths when data sovereignty is required; (4) monitor traffic (packet captures or flow logs) during initial commissioning and after any firmware update; and (5) keep a documented update governance process (who updates, when, and how changes are validated) to avoid accidental regression of both safety and security settings. These practices are particularly important for VLA experiments, where high-level autonomy must be paired with strict safety interlocks and strong network hygiene.

For our deployments, the key lesson is that security controls must be designed into the daily workflow: networking, logging, and access control are as important as kinematics and perception when the robot is connected to institutional infrastructure. Treating cybersecurity as part of the acceptance test (together with locomotion and manipulation) reduces the risk of silent data leakage and improves reproducibility of subsequent experiments.

## 6 Conclusion

These first experiments suggest that Unitree G1 EDU with RH56DFTP hands is not a turnkey humanoid worker, but it is already a valuable research platform. Its strongest asset is not that every subsystem is mature; it is that the platform makes current embodied-AI questions concrete and experimentally tractable. Walking and running are already convincing, and the broader literature confirms that the platform can support considerably more advanced

locomotion, loco-manipulation, and teleoperation under custom research stacks [34]-[39]. The main research friction lies elsewhere: software integration, middleware reproducibility, safe use of fragile dexterous hands, tactile-data fusion, and the passage from remote control to unified autonomous control.

In that sense the platform is best understood as a transition technology. It brings humanoid research out of the purely aspirational phase and into everyday laboratory engineering. We therefore view it as one of the first comparatively accessible commercially available humanoid setup that can realistically enter university research and education, while also making it impossible to ignore how much work remains between impressive demos and dependable autonomous locomanipulation.

## 7 Future work

Our near-term work focuses on making the platform more reproducible and less brittle for manipulation experiments. This includes hardening the software stack with explicit version pinning and configuration snapshots, improving mechanical strain relief for the external hand cabling, and extending the sensing setup with additional cameras for human-facing perception. On the autonomy side, we plan controlled experiments with UniFoLM-VLA and NVIDIA GR00T, starting with offline evaluation and teleoperation dataset collection, followed by safety-constrained on-robot rollouts. A key technical goal is to integrate RH56DFTP tactile and force signals into the observation and control loop (e.g., slip and contact-event feedback for grasp stabilization), and to replace the rudimentary SDK navigation with a more capable SLAM and locomotion planning pipeline suitable for repeatable locomanipulation in real environments.

## Acknowledgement

This project has been funded by the EU NextGenerationEU through the Recovery and Resilience Plan for Slovakia under the project No. 09I01-03-V04-00048.

## References

- [1] Unitree Robotics: Unitree G1. Product page. Available online: <https://www.unitree.com/g1/>
- [2] Beijing Inspire-Robots Technology Co., Ltd.: The Dexterous Hand RH56DFTP User Manual, V1.0.0, 2025. Available online: <https://en.inspire-robots.com/wp-content/uploads/2025/01/INSPIRE-ROBOTS-The-Dexterous-Hand-RH56DFTP-User-Manual-V1.0.0.pdf>
- [3] Unitree Robotics: Unitree Explore APP Download – G1 tutorial videos and manuals. Available online: <https://www.unitree.com/mobile/app/g1>

- [4] unitreerobotics: unitree\_sdk2. GitHub repository. Available online: [https://github.com/unitreerobotics/unitree\\_sdk2](https://github.com/unitreerobotics/unitree_sdk2)
- [5] unitreerobotics: unitree\_sdk2\_python. GitHub repository. Available online: [https://github.com/unitreerobotics/unitree\\_sdk2\\_python](https://github.com/unitreerobotics/unitree_sdk2_python)
- [6] unitreerobotics: unitree\_ros2. GitHub repository. Available online: [https://github.com/unitreerobotics/unitree\\_ros2](https://github.com/unitreerobotics/unitree_ros2)
- [7] unitreerobotics/unitree\_sdk2\_python Issue #53: Problems with cyclonedds while using NVIDIA Jetson, 2025. Available online: [https://github.com/unitreerobotics/unitree\\_sdk2\\_python/issues/53](https://github.com/unitreerobotics/unitree_sdk2_python/issues/53)
- [8] unitreerobotics/unitree\_sdk2\_python Issue #43: Instructions to control G1 movement via SDK - do not enter debug mode, 2025. Available online: [https://github.com/unitreerobotics/unitree\\_sdk2\\_python/issues/43](https://github.com/unitreerobotics/unitree_sdk2_python/issues/43)
- [9] NVIDIA: Jetson Orin technical specifications. Available online: <https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-orin/>
- [10] NVIDIA: What Is Isaac Sim? Available online: <https://docs.omniverse.nvidia.com/isaacsim/latest/overview.html>
- [11] NVIDIA: Isaac Sim technical requirements and minimum system requirements. Available online: <https://docs.omniverse.nvidia.com/isaacsim/latest/common/technical-requirements.html>
- [12] Isaac Lab Documentation: Overview. Available online: <https://isaac-sim.github.io/IsaacLab/v1.4.1/>
- [13] Unitree Robotics: Official Open Source portal. Available online: <https://www.unitree.com/opensource/>
- [14] unitreerobotics/unifolm-vla: UnifoLM-VLA-0: A Vision-Language-Action (VLA) Framework under UnifoLM Family. GitHub repository, 2026. Available online: <https://github.com/unitreerobotics/unifolm-vla>
- [15] J. Bjorck et al.: GR00T N1: An Open Foundation Model for Generalist Humanoid Robots. arXiv:2503.14734, 2025.
- [16] Hugging Face LeRobot Documentation: Unitree G1. Available online: [https://huggingface.co/docs/lerobot/main/en/unitree\\_g1](https://huggingface.co/docs/lerobot/main/en/unitree_g1)
- [17] P. P. Surve, A. Shabtai, Y. Elovici: SoK: Cybersecurity Assessment of Humanoid Ecosystem. arXiv:2508.17481, 2025.
- [18] V. Mayoral-Vilches, A. Makris, K. Finisterre: Cybersecurity AI: Humanoid Robots as Attack Vectors. arXiv:2509.14139, 2025.
- [19] Unitree Robotics Store: Unitree G1 store page and EDU note on secondary development. Available online: <https://shop.unitree.com/en-ca/products/unitree-g1>
- [20] unitreerobotics: unitree\_IL\_lerobot. GitHub repository. Available online: [https://github.com/unitreerobotics/unitree\\_IL\\_lerobot](https://github.com/unitreerobotics/unitree_IL_lerobot)
- [21] unitreerobotics: UnifoLM\_WBT\_Dataset. Hugging Face collection. Available online: <https://huggingface.co/collections/unitreerobotics/unifolm-wbt-dataset>
- [22] NVIDIA: NVIDIA Announces Isaac GR00T N1 - the world's first open humanoid robot foundation model - and simulation frameworks to speed robot development, 2025. Available online: <https://investor.nvidia.com/news/press-release-details/2025/NVIDIA-Announces-Isaac-GR00T-N1---the-Worlds-First-Open-Humanoid-Robot-Foundation-Model--and-Simulation-Frameworks-to-Speed-Robot-Development/default.aspx>
- [23] NVIDIA Developer: Isaac GR00T. Available online: <https://developer.nvidia.com/isaac/gr00t>
- [24] C. Smith, P. Du: Building a Synthetic Motion Generation Pipeline for Humanoid Robot Learning. NVIDIA Technical Blog, 2025. Available online: <https://developer.nvidia.com/blog/building-a-synthetic-motion-generation-pipeline-for-humanoid-robot-learning/>
- [25] Hugging Face LeRobot Documentation: Policy Deployment (lerobot-rollout). Available online: <https://huggingface.co/docs/lerobot/main/inference>
- [26] Hugging Face LeRobot Documentation: LeRobotDataset v3.0. Available online: <https://huggingface.co/docs/lerobot/lerobot-dataset-v3>
- [27] Open X-Embodiment Collaboration: Open X-Embodiment: Robotic Learning Datasets and RT-X Models. arXiv:2310.08864, 2023.
- [28] A. Khazatsky et al.: DROID: A Large-Scale In-the-Wild Robot Manipulation Dataset. Robotics: Science and Systems (RSS), 2024.
- [29] H.-S. Fang et al.: RH20T: A Comprehensive Robotic Dataset for Learning Diverse Skills in One-Shot. arXiv:2307.00595, 2023.
- [30] Y.-W. Chao et al.: DexYCB: A Benchmark for Capturing Hand Grasping of Objects. CVPR 2021. Available online: [https://research.nvidia.com/publication/2021-06\\_dexycb-benchmark-capturing-hand-grasping-objects](https://research.nvidia.com/publication/2021-06_dexycb-benchmark-capturing-hand-grasping-objects)
- [31] Y. Liu et al.: HOI4D: A 4D Egocentric Dataset for Category-Level Human-Object Interaction. CVPR 2022. Project page: <https://hoi4d.github.io/>
- [32] Z. Fan et al.: ARCTIC: A Dataset for Dexterous Bimanual Hand-Object Manipulation. CVPR 2023. DOI: 10.1109/CVPR52729.2023.01244.
- [33] O. Taheri et al.: GRAB: A Dataset of Whole-Body Human Grasping of Objects. In: Computer Vision - ECCV 2020. DOI: 10.1007/978-3-030-58548-8\_34.

- [34] W. Sun et al.: Learning Perceptive Humanoid Locomotion over Challenging Terrain. arXiv:2503.00692, 2025.
- [35] A. B. Ghansah, S. A. Esteban, A. D. Ames: Hierarchical Reduced-Order Model Predictive Control for Robust Locomotion on Humanoid Robots. arXiv:2509.04722, 2025.
- [36] Y. Fu et al.: DemoHLM: From One Demonstration to Generalizable Humanoid Loco-Manipulation. arXiv:2510.11258, 2025.
- [37] W. Sun et al.: ULC: A Unified and Fine-Grained Controller for Humanoid Loco-Manipulation. arXiv:2507.06905, 2025.
- [38] T. He et al.: VIRAL: Visual Sim-to-Real at Scale for Humanoid Loco-Manipulation. arXiv:2511.15200, 2025.
- [39] S. Atar et al.: Humanoids in Hospitals: A Technical Study of Humanoid Surrogates for Dexterous Medical Interventions. arXiv:2503.12725, 2025.
- [40] Unitree Robotics: G1 SDK Development Guide. Available online:  
[https://support.unitree.com/home/en/G1\\_developer](https://support.unitree.com/home/en/G1_developer)
- [41] unitreerobotics: unitree\_sim\_isaacclab. GitHub repository. Available online:  
[https://github.com/unitreerobotics/unitree\\_sim\\_isaacclab](https://github.com/unitreerobotics/unitree_sim_isaacclab)