This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Arcadia: Toward a Full-Lifecycle Framework for Embodied Lifelong Learning

Minghe Gao[1,2*†], Juncheng Li[1‡], Yuze Lin[1*], Xuqi Liu[1*], Jiaming Ji[3*]
Xiaoran Pan[1], Zihan Xu[1], Xian Li[1], Mingjie Li[2], Wei Ji[4], Rong Wei[5], Rui Tang[5],
Qizhou Wang[2], Kai Shen[6], Jun Xiao[1], Qi Wu[7], Siliang Tang[1], Yueting Zhuang[1]
[1]Zhejiang University [2]Unitree Tech [3]Peking University [4]Nanjing University
[5]Manycore Tech [6]Bytedance Seed [7]University of Adelaide

Abstract

We contend that embodied learning is fundamentally a lifecycle problem rather than a single-stage optimization. Systems that optimize only one link (data collection, simulation, learning, or deployment) rarely sustain improvement or generalize beyond narrow settings. We introduce **Arcadia**, *a closed-loop framework that operationalizes embodied lifelong learning by tightly coupling four stages: (1) Self-evolving exploration and grounding for autonomous data acquisition in physical environments, (2) Generative scene reconstruction and augmentation for realistic and extensible scene creation, (3) a Shared embodied representation architecture that unifies navigation and manipulation within a single multimodal backbone, and (4) Sim-from-real evaluation and evolution that closes the feedback loop through simulation-based adaptation. This coupling is non-decomposable: removing any stage breaks the improvement loop and reverts to one-shot training.* **Arcadia** *delivers consistent gains on navigation and manipulation benchmarks and transfers robustly to physical robots, indicating that a tightly coupled lifecycle: continuous real-world data acquisition, generative simulation update, and shared-representation learning, supports lifelong improvement and end-to-end generalization. We release standardized interfaces enabling reproducible evaluation and cross-model comparison in reusable environments, positioning* **Arcadia** *as a scalable foundation for general-purpose embodied agents.*

1. Introduction

Humans learn across an entire lifespan, accumulating and reorganizing experience to enable broad generalization and skill transfer beyond the training context [5, 6, 13, 19, 22]. Cognitive and embodied theories further argue that competence is shaped by continuous perception–action coupling over time, not by short-lived skills in isolation [10, 33, 40, 46]. This perspective inspires implications for robotics: optimizing only one link of the chain (*e.g.*, training in static simulators or deploying without feedback) rarely sustains improvement or robustness [35, 47]. Recent studies show that coordinating multiple stages of the embodied pipeline (real-world data collection, generative simulation, unified representation learning, and deployment-time feedback) yields stronger generalization and data efficiency than siloed designs [7, 8, 26, 31, 41, 49]. We adopt this view and refer it as **embodied lifelong learning**: designing and evaluating embodied intelligence as closed loops that continually couple real experience with updates to representations, policies, and the simulators in which they are trained.

Recent efforts have begun to integrate previously isolated stages. GRUtopia [45] unifies diverse scenarios, agents, and benchmarks within a single simulation system, while NaVILA [11] connects high-level language instructions to low-level motor control and validates them on physical robots to mitigate the sim-to-real gap. These works expand the coverage of the pipeline, yet they still fall short of a full lifecycle. GRUtopia primarily extends the simulation side of the pipeline, while NaVILA primarily extends the execution span toward real robots. Both broaden coverage but stop short of closing the lifecycle: they do not establish a persistent, editable path from deployment experience back into simulation assets, nor a persistent mechanism for routing in-deployment signals to supervise subsequent training. As a result, they connect individual edges without maintaining the full real-to-sim-to-real loop across stages.

On the real-to-sim side, two key limitations dominate: **(1) Exogenous Data Dependence.** From a lifecycle perspective, training data are most effective when sourced from, and remain grounded in, deployment contexts. Reliance on external, off-distribution sources weakens this coupling: YouTube videos are misaligned with embodied control objectives, while data from quadruped robots re-

*Equal Contribution
†Work done when interning at Unitree
‡Corresponding author.

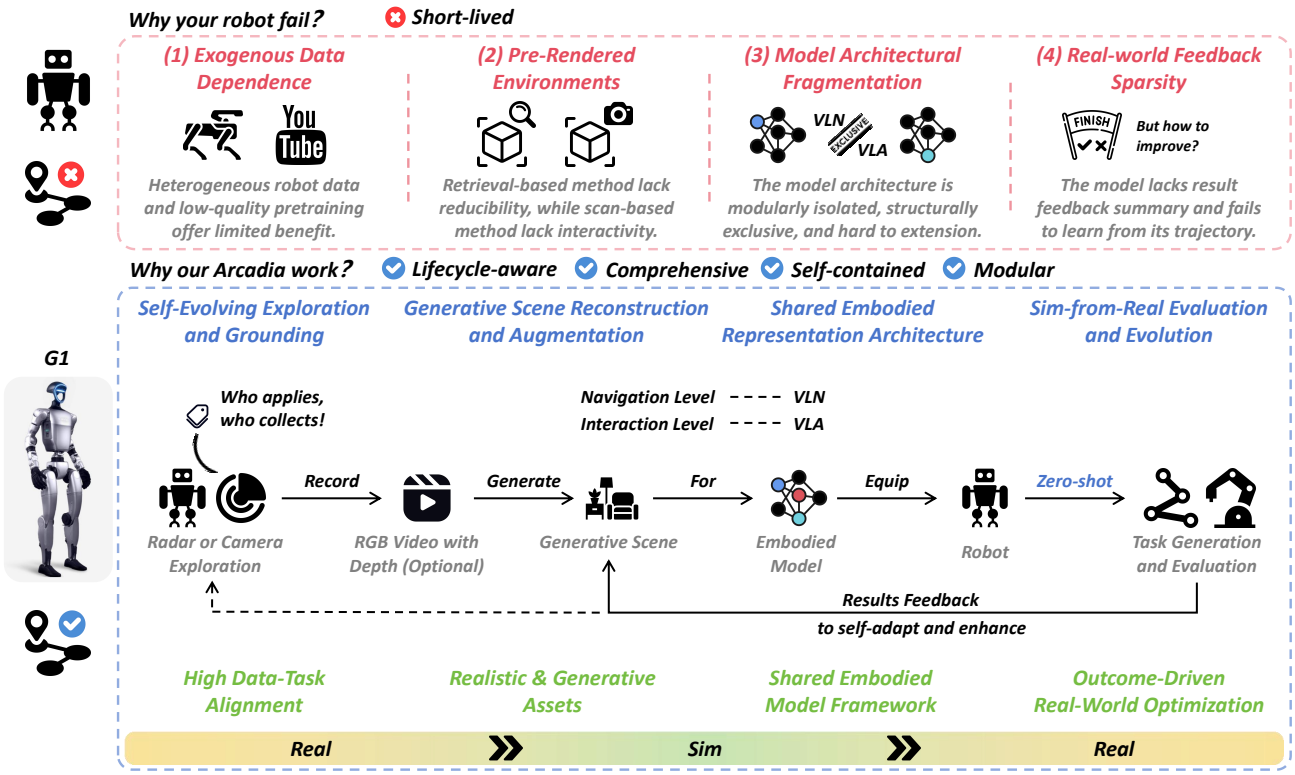


Figure 1. **Arcadia** provides an overview of a full real-to-sim-to-real lifecycle for embodied learning, illustrating how the framework closes the loop between real-world experience, simulation, and redeployment while addressing four core limitations in contemporary embodied AI: exogenous data dependence, static pre-rendered environments, fragmented model architectures, and sparse real-world feedback.

flects a distinct morphology and viewpoint relative to a humanoid. Such exogenous and noisy data introduce distribution mismatches and are costly to curate or modify in simulation, let alone to adapt based on feedback. It hinders execution signals from driving asset or policy updates and stalling lifelong improvement. This limitation is evident in NaVILA [11], which compensates for data scarcity using mismatched external corpora, yielding limited gains.

(2) Pre-Rendered Environments. Under a lifecycle view, simulators should ingest real observations and remain editable so that deployment experience can be incorporated into new scenes. Pre-rendered or static environments (e.g., Matterport3D [9] and Habitat v1 [37]) with limited physical properties, restrict controllability and prevent the faithful insertion of deployment-time variations, slowing or even blocking the data–asset–policy loop. In practice, the common workflow, which trains on static sets and then validates within Isaac Sim [30]/Lab [29], adds engineering overhead without restoring lifecycle coupling. Even large, high-fidelity suites such as GRUtopia [45] rely on finite asset libraries and retrieval-based scene variations, limiting generative adaptation and hindering sustained improvement.

On the sim-to-real side, two further limitations appear:

(3) Model Architectural Fragmentation. A lifecycle system benefits from a shared embodied representation so that data, supervision, and deployment feedback can propagate across task families and embodiments [14, 32, 38]. In practice, pipelines often build separate task-specific stacks with incompatible assumptions (e.g., navigation agents modeled as oriented bounding boxes in indoor simulators [1, 21] and tabletop manipulation under fixed camera and end-effector control [16, 39]), which block cross-task credit assignment and prevent coherent policy updates from real deployments. The result is siloed improvement for long-horizon, multi-skill objectives, which is the opposite of lifecycle coupling.

(4) Real-world Feedback Sparsity. In a lifecycle system, deployment should provide dense, structured signals that update both simulator assets and policies. Yet many pipelines treat deployment as one-shot, capturing at most coarse success/failure labels [25, 36]. This sparsity stalls the cycle: long-horizon errors cannot be localized, partial progress and environmental shifts are not fed back, and execution confidence cannot guide refinement. Effective lifecycle coupling requires routing multi-level feedback (task progress, scene dynamics, hardware constraints) back into asset updates and policy learning, rather than treating deployment as the endpoint.

We summarize these four limitations in Figure 1. Taken together, they point not only to isolated algorithmic gaps but to a breakdown of lifecycle coupling: from data acqui-

sition through simulation, representation, and deployment-time supervision. A platform oriented to embodied lifecycle learning should therefore (i) maintain strong alignment between collected experience and target tasks, (ii) translate real observations into editable, generative simulation assets, (iii) learn with a shared, extensible embodied representation across tasks, and (iv) route outcome-driven, in-deployment feedback back into both assets and policies.

To address these challenges, we present **Arcadia, a full-lifecycle framework for embodied lifelong learning in a closed loop**. Specifically, it comprises four components: **(1) Self-Evolving Exploration and Grounding**, in which robots autonomously collect task-aligned data within their physical environments; **(2) Generative Asset Reconstruction and Augmentation**, which transforms real-world sensory data into realistic and extensible simulation assets; **(3) Shared Embodied Representation Architecture**, enabling visual-language-navigation (VLN) and visual-language-action (VLA) model to share a common multimodal backbone; and **(4) Sim-from-Real Evaluation and Evolution**, which integrates real-world feedback into simulation as adaptive supervision to continuously refine embodied behavior.

Crucially, the four components function both independently and as a coupled closed-loop system. Each addresses a distinct bottleneck in the embodied intelligence lifecycle while collectively driving continual self-improvement. This design yields substantial gains across both simulated and physical domains. In simulator evaluations, **Arcadia** achieves an average improvement of 7.07% on navigation and 11.08% on manipulation tasks compared with other baseline methods. Furthermore, in real-world experiments conducted on a Unitree G1 robot equipped with a Dex-3 manipulator, **Arcadia** successfully completed 46% of navigation and 27% of manipulation trials, substantially outperforming NaVILA (13%) and OpenVLA (9%). These improvements are particularly evident in multi-destination navigation and multi-object manipulation scenarios. Collectively, these results highlight **Arcadia**'s promising generalization in real-world embodied operation. We also open-source standardized interfaces for benchmarking and cross-model evaluation within reusable environments.

In summary, our contributions are as follows:

- We propose **Arcadia**, a full-lifecycle embodied intelligence framework that unifies data, simulation, learning, and feedback into a closed self-improving loop.
- We develop a real-to-sim generative pipeline that reconstructs real assets for scalable, high-fidelity simulation.
- We introduce a novel *Sim-from-Real* feedback mechanism that transfers structured real-world signals back into simulation for continual policy and asset refinement, effectively closing the real-to-sim-to-real loop.

2. Related Works

2.1. Embodied Lifelong Learning

Recent advances in embodied intelligence extend lifelong learning beyond static datasets toward continuously evolving perception–action loops. Rather than optimizing fixed policies in isolated tasks, embodied lifelong learning seeks persistent cycles that link data acquisition, model adaptation, and deployment feedback. This perspective has inspired systems combining autonomous exploration, self-supervised refinement, and multi-task generalization across diverse embodiments [12, 14, 17, 32, 33, 38]. Several frameworks have explored continual adaptation within real or simulated settings such as Octo, RoboCat, and RoboAgent which enable robots to expand skills across domains through ongoing experience [7, 8, 41]. These efforts mark a shift from episodic to adaptive embodied learning.

However, most pipelines remain fragmented: data, training, and deployment are loosely coupled, and deployment feedback rarely supervises further learning. **Arcadia** adopts a closed-lifecycle formulation, in which real experience continuously informs simulation, representation, and policy updates, enabling sustained self-improvement across the full perception–action loop.

2.2. Bridging Simulation and Reality

Because large-scale real-world experimentation is costly and risky, embodied learning often maps physical environments into simulation, where agents can explore efficiently. This results in a reality gap: the mismatch between simulated and physical domains [15, 34, 42]. Domain randomization over textures, lighting, and dynamics improves robustness and has powered scalable systems like OpenAI's Dactyl [2], yet handcrafted simulators capture only limited real-world complexity, such as deformable objects, sensor noise, or dynamic scenes, leaving trained policies brittle after deployment.

Recent work has begun to narrow this gap through generative reconstruction and adaptive simulation frameworks [11, 45], which translate real-world observations into editable, physics-consistent virtual assets. **Arcadia** extends this direction toward a bidirectional, closed-loop paradigm: real-world data regenerates simulation assets and supervision, while feedback from simulation refines shared representations and policies before redeployment. This continual real-to-sim-to-real cycle transforms simulation from a static proxy into an active driver of embodied adaptation.

3. Method

Given a natural-language instruction such as "bring me the cup from the table", **Arcadia** executes a complete real-to-sim-to-real cycle (Figure 2 and 3). The process starts

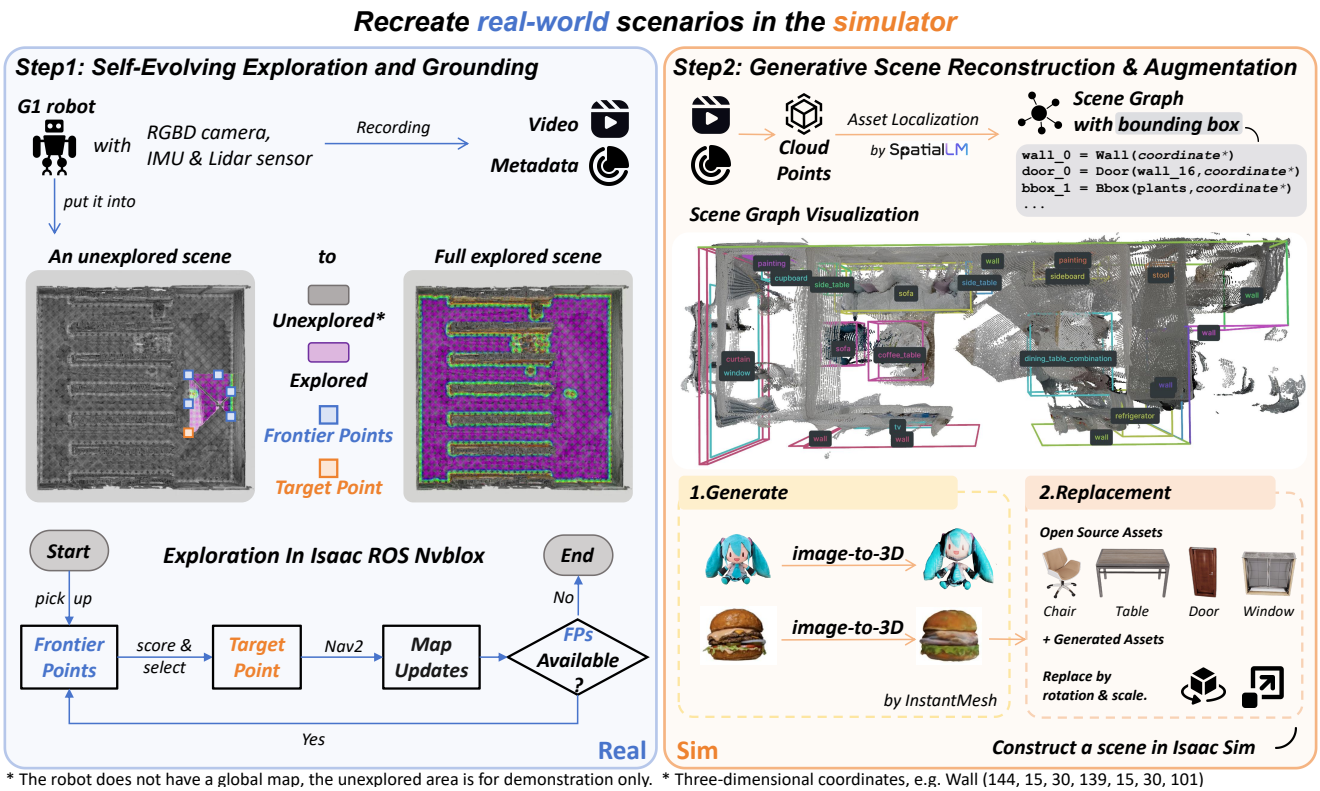


Figure 2. Overview of **Arcadia**'s real-to-sim pipeline: robots autonomously explore real environments to collect multimodal data (Step 1), which are then reconstructed and augmented into editable 3D scenes for simulation (Step 2).

with autonomous exploration and data collection in real environments (Sec. 3.1), followed by generative reconstruction of realistic, editable simulation assets (Sec. 3.2). The simulator then trains unified navigation and interaction policies under a shared embodied model (Sec. 3.3). Finally, real-world deployment provides feedback that is incorporated back into simulation to refine both assets and policies (Sec. 3.4). This closed loop enables continual, self-improving embodied learning across the robot's lifecycle.

3.1. Self-Evolving Exploration and Grounding

Conventional embodied datasets, collected from exogenous sources or passive mapping, often exhibit weak task alignment and domain gaps between training and deployment. To bridge this, **Arcadia** autonomously acquires task-relevant data in the same physical environments used for deployment, ensuring that perception and control models learn under realistic conditions.

Built on Isaac ROS [27] and Nvblox [28] for SLAM and 3D reconstruction, our method employs a frontier-based exploration policy that maximizes information gain. Frontier points, the boundaries between explored and unexplored regions, are scored by expected entropy reduction, and the robot visits the highest-scoring ones using low-level motion control APIs. The map and frontier sets are continuously updated, producing adaptive trajectories that balance coverage, efficiency, and semantic relevance. Compared with grid or scripted exploration, this self-evolving policy emphasizes regions critical to downstream tasks, improving both sample efficiency and task-grounded coverage.

After exploration, **Arcadia** outputs synchronized multimodal data (RGB-D, LiDAR, IMU, odometry, poses). Retaining complete observation histories instead of discarding intermediate frames yields dense, temporally grounded supervision for reconstruction and policy learning, maintaining lifecycle consistency and reducing the real-to-sim gap.

3.2. Generative Scene Reconstruction

Traditional simulators rely on static scans or retrieval-based scene assembly, which require manual curation and offer limited domain coverage. To overcome these constraints, **Arcadia** employs a generative reconstruction pipeline that converts real environments into editable, task-aligned simulation assets, combining the realism of physical data with the scalability of synthesis.

From the multimodal inputs collected in Section 3.1, videos and point clouds are parsed into a structured 3D scene graph $G = (V, E)$, with objects/architectural elements as nodes and spatial relations as edges, implemented by a scene parsing module (*e.g.*, SpatialLM [24]). Rather than retrieving meshes from databases, **Arcadia** synthesizes assets directly from multi-view observations using a Gaussian-splat-based reconstructor [48], producing simulator-compatible USD objects with consistent geom-

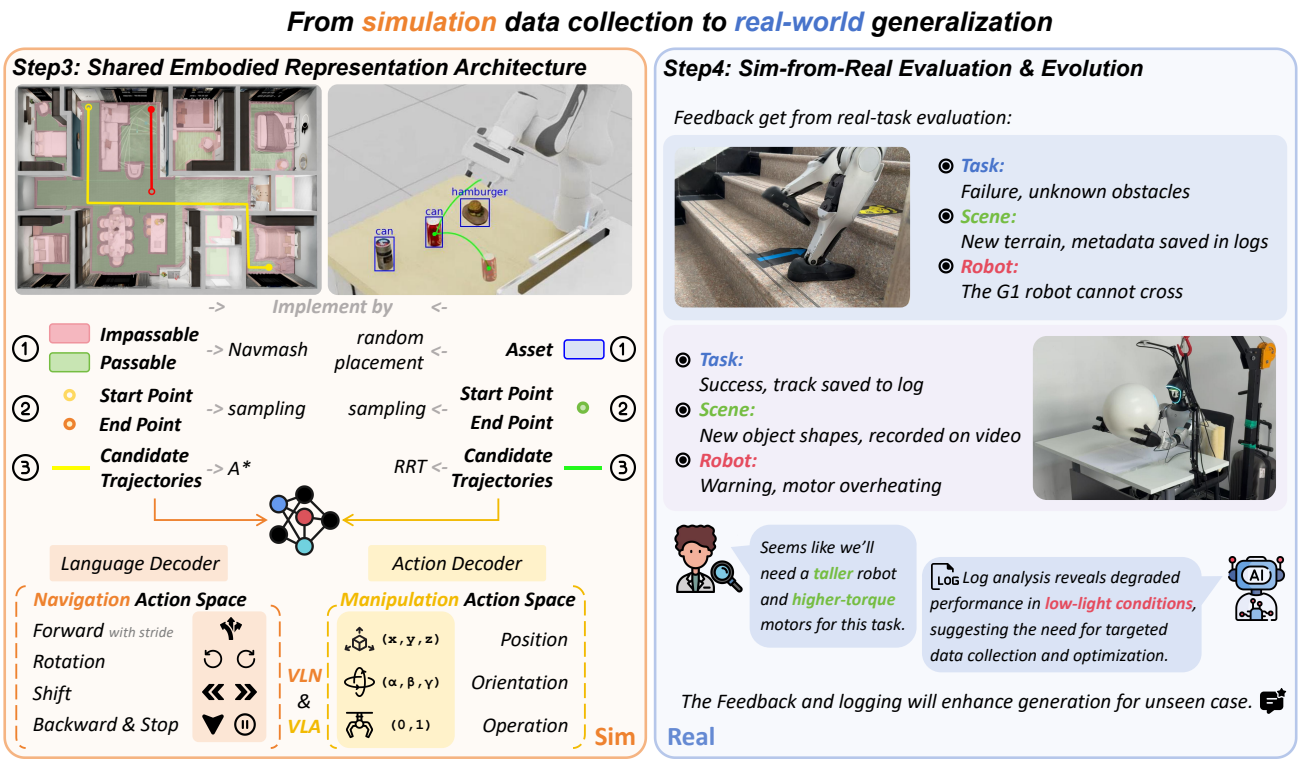


Figure 3. Overview of **Arcadia**'s sim-to-real pipeline: navigation and manipulation trajectories are collected in simulation (Step 3) using A* and RRT planning within a shared embodied architecture, then real-world feedback is integrated for continual refinement (Step 4).

etry, texture, and semantics. This enables broad domain expansion without manual intervention, reduces asset bias, and preserves task semantics observed in the real world.

All assets are imported into Isaac Sim [30] via an automated management interface, yielding high-fidelity, extensible environments. By replacing manual retrieval with generative synthesis, **Arcadia** delivers realistic yet diverse simulations that support scalable, lifelong embodied learning while maintaining lifecycle consistency.

3.3. Shared Embodied Representation Architecture

Most embodied systems train different locomotion and manipulation tasks as separate models with independent perception stacks, action spaces, and objectives. This separation imposes a hard boundary between spatial reasoning and fine-grained interaction, resulting in two issues: (i) no shared grounding across language, vision, and action, requiring each model to relearn objects, relations, and affordances; and (ii) brittle cross-stage coupling, where navigation can reach the vicinity of a goal but exposes no structured state for manipulation, while manipulation assumes an arm-centric viewpoint. This fragmentation increases data requirements, limits transfer, and degrades performance on long-horizon language-conditioned instructions.

Arcadia instead employs a unified multimodal backbone jointly trained across navigation and manipulation tasks, with lightweight task-specific decoders for action generation. Supervision is generated in simulation: for navigation, start–goal pairs are sampled and A* produces collision-free paths expressed in a 7-primitive discrete control space, which generalizes across robot morphologies; for manipulation, RRT generates physically feasible trajectories. All trajectories are language-conditioned at the input, formatted as in VLN-CE [20] and BridgeData V2 [44], and passed through the shared perception/state encoder into their respective decoders.

Joint training over both tasks induces a shared embodied state representation, global layout, reachable goals, and approach strategies from navigation are encoded in the same latent space as local affordances and contact behaviors from manipulation. This shared grounding reduces modality drift and facilitates representation transfer between tasks. While each task is evaluated independently under its corresponding embodiment, this unified backbone establishes a common semantic and spatial understanding, enabling coherent reasoning over long-horizon instructions.

3.4. Sim-from-Real Evaluation and Evolution

Most embodied pipelines terminate after inference and treat deployment as purely evaluative, discarding dense execution traces. **Arcadia** instead treats deployment as an additional supervision stage: real-world rollouts are logged, decomposed into structured feedback signals, and reintegrated into simulation to update both policy and environment. This

Table 1. Comparison of various methods on different benchmarks.

	VLN-CE-Isaac				R2R Val-Unseen				RxR Val-Unseen				ScanQA
	NE↓	OS↑	SR↑	SPL↑	NE↓	OS↑	SR↑	SPL↑	NE↓	OS↑	SR↑	SPL↑	Meteor↑
Tuning	6.12	50.7	44.9	38.5	5.44	56.0	47.1	41.3	7.12	45.2	40.9	53.7	13.4
NaVILA	5.94	53.2	45.1	40.1	5.30	61.0	51.6	47.5	6.78	47.8	42.6	57.8	16.3
Arcadia w/o feedback	5.51	56.9	48.7	43.6	5.18	62.0	54.2	49.4	6.71	49.4	44.5	58.3	19.0
Arcadia w/ feedback	5.32	58.6	50.1	45.0	5.03	64.5	55.9	49.8	6.50	50.3	45.7	60.1	19.1

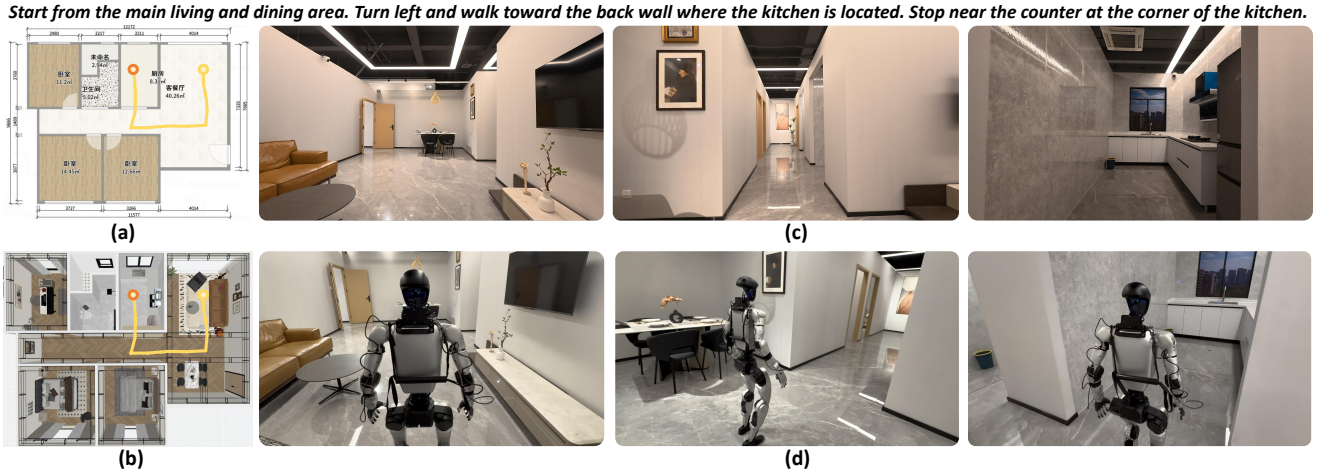


Figure 4. Navigation route from the living and dining area to the kitchen by Unitree G1 robot. (a) Floor plan showing the designated path. (b) 3D rendering of the environment with the same route. (c) First-person visual sequence along the path. (d) Third-person view.

closes the loop between real and simulated data and enables continual adaptation rather than static training.

We extract feedback along three channels. **Task feedback.** Each task is decomposed into step-level actions, and feedback at time t is defined as

$$F_t^T = \lambda_1 R_t + \lambda_2 \|s_{t+1} - s_t\| + \lambda_3 \mathcal{L}_{\text{conf}}(o_t, \hat{o}_t) + \lambda_4 \mathcal{L}_{\text{goal}}(s_t, s_g),$$

where R_t is the scalar reward, $\|s_{t+1} - s_t\|$ measures state transition magnitude, $\mathcal{L}_{\text{conf}}$ measures perceptual consistency between predicted and observed observations, and $\mathcal{L}_{\text{goal}}$ measures distance to the goal state; λ_i weight these terms. This converts raw trajectories into a supervisory signal that jointly encodes reward, dynamics, perception, and goal alignment, allowing both global rollout scoring and localized error attribution.

Scene feedback. Multimodal sensory streams (RGB, depth, LiDAR, IMU) are used to characterize environment dynamics and perception quality. Failures such as degraded mapping in low light or the appearance of previously unseen objects are logged and used to update the simulator: new assets are instantiated or perturbations are injected, so that future simulation reflects deployment conditions rather than a fixed, pre-rendered scene.

Robot feedback. Hardware telemetry (joint states, actuator load, communication stability) is monitored during execution. Constraint violations (*e.g.*, a robot exceeding allowable step height or a manipulator exceeding payload limits) are recorded as a robot-level signal F^R, which is used both for safety gating and for adapting motion policies to platform limits.

These feedback channels are fed back into simulation to update assets, dynamics, and supervision targets, producing a bidirectional real-to-sim-to-real loop. In **Arcadia**, policy learning, environment generation, and evaluation are no longer disjoint stages but part of the same iterative update process, reducing the sim-to-real gap at training time rather than compensating for it at deployment.

4. Experiments

We evaluate **Arcadia** across both simulation and real-world settings to assess its effectiveness throughout the embodied intelligence lifecycle. Our experiments aim to answer four key questions: **(Q1)** Does **Arcadia** improve performance on VLN tasks? **(Q2)** Does it enhance VLA tasks involving object manipulation? **(Q3)** How well does it transfer to physical robots in real-world environments? **(Q4)** Which components contribute most to its performance, as shown by ablation studies? We address these questions through standardized benchmarks, controlled robotic deployments, and systematic analyses.

4.1. VLN Task Performance

Settings. We evaluate the impact of **Arcadia** on VLN performance under the following four training configurations:

- *Tuning.* Fine-tuning the backbone model solely on VLN-CE [20] data in a single-stage setup.

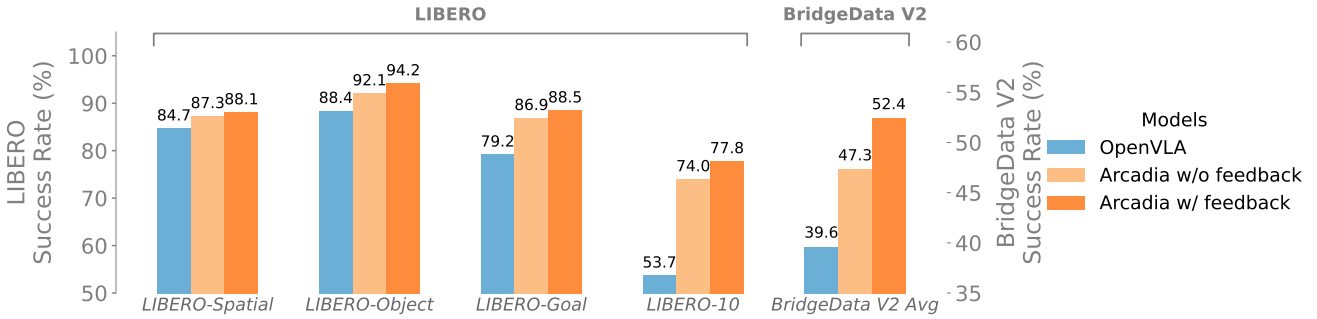


Figure 5. Comparison of various models and methods on different benchmarks.

Task: Put the red cube on top of the blue cube.

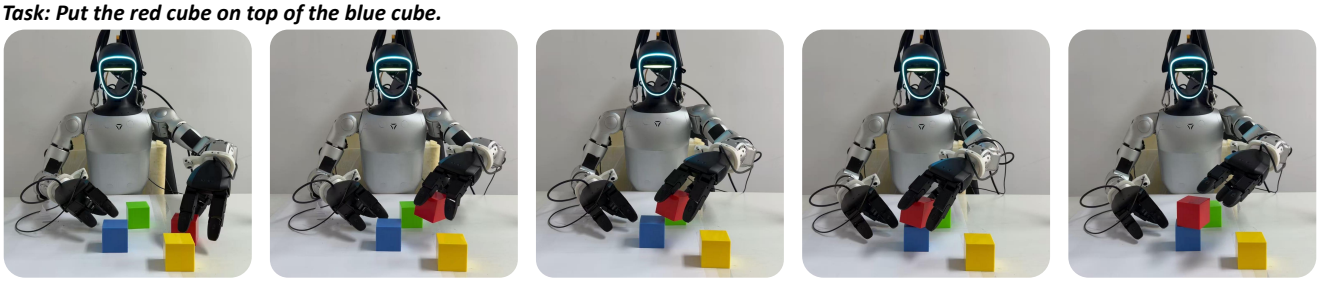


Figure 6. A representative case to illustrate Arcadia's manipulation behavior.

- *NaVILA* [11]. Extends *Tuning* by adding an auxiliary QA fine-tuning stage to improve language grounding.
- ***Arcadia*** *w/o feedback.* Uses the same multi-stage setup as *NaVILA* but replaces first-stage trajectories with task-aligned data autonomously collected by **Arcadia**.
- ***Arcadia*** *w/ feedback.* The full **Arcadia** framework, where real-world feedback further refines navigation data and policies, completing the real-to-sim-to-real loop.

Backbone and Simulator. **Arcadia** adopts the hierarchical architecture of NaVILA, composed of a high-level vision–language model (VLM) for perception and reasoning, and low-level controllers interfaced with robot APIs for execution in both simulation and the real world. We use *Qwen2.5-VL* [4] as the high-level VLM and the open-source Unitree G1 [43] platform for embodied validation. All simulated navigation data are organized in the VLN-CE [20] format for cross-benchmark compatibility within Isaac Sim [30].

Benchmarks and Evaluation. We benchmark **Arcadia** on VLN-CE-Isaac [11], R2R [1], and RxR [21], covering diverse instruction-following settings and visual domains. We also include ScanQA [3] for navigation-oriented question answering. Consolidated results are shown in Table 1.

Results. Integrating QA-based reasoning into navigation enhances instruction comprehension, while **Arcadia**'s self-collected data provide greater environmental diversity and scene realism. Under identical architectures and training budgets, ***Arcadia*** *w/o feedback* outperforms *NaVILA* by an average success rate (SR) of 2.7%, demonstrating the benefit of autonomous, task-aligned data generation. Incorporating real-world feedback (***Arcadia*** *w/ feedback*) further improves generalization and trajectory diversity, achieving the best performance across all benchmarks. Results in Table 1 confirm that these gains stem primarily from improved data quality and closed-loop refinement rather than dataset size alone.

4.2. VLA Task Performance

Settings. To assess **Arcadia**'s effectiveness on manipulation tasks, we compare three configurations:

- *OpenVLA* [18]. The baseline is trained in a single-stage setup using only the training dataset corresponding to the evaluation environment.
- ***Arcadia*** *w/o feedback.* Uses the same data scale as *OpenVLA* but replaces all trajectories with manipulation data generated by **Arcadia**'s simulation pipeline.
- ***Arcadia*** *w/ feedback.* The full closed-loop framework, where real-world feedback further refines policies and data, completing the real-to-sim-to-real cycle.

Backbone and Simulator. **Arcadia** employs the same high-level vision–language model, *Qwen2.5-VL* [4], augmented with an OpenVLA-style 7D action de-tokenizer. Simulated training is conducted in Isaac Sim [30], and the GRUtopia [45] robotic arm serves as the embodied platform for interaction and data collection within physically consistent scenes.

Benchmarks and Evaluation. We evaluate **Arcadia** across two major manipulation benchmarks: LIBERO [23] and BridgeData (V2) [44]. To ensure compatibility, all datasets are standardized to the RLDS format. Evaluation metrics and scripts are aligned with their official implementations for consistency and fairness.

Results. **Arcadia** outperforms *OpenVLA* on all benchmarks. The improvements are particularly pronounced in BridgeData (V2), where feedback enhances object grounding and long-horizon stability. These results highlight **Arcadia**'s strong cross-task generalization and validate its scalability beyond language-conditioned navigation.

4.3. Real-World Evaluation

To evaluate **Arcadia**'s real-world performance, we established a physical testing environment for human evaluation, comprising 100 navigation and 100 manipulation tasks. Figures 4 and 6 illustrate representative configurations of this setup. It is noteworthy that all navigation tasks were conducted under a zero-shot setting, with no task-specific fine-tuning. In contrast, for manipulation tasks, we fine-tuned a dedicated dual-arm model to handle operations involving four blocks placed on a tabletop. Experiments were performed using a Unitree G1 robot equipped with a Dex-3 manipulator.

As baselines, they achieved 13 navigation and 9 manipulation successful completions, respectively, while **Arcadia** completed 46 and 27 tasks. Notably, in multi-destination navigation and multi-object manipulation scenarios, where both baselines failed, **Arcadia** maintained a 17% success rate, often completing initial subtasks but struggling with extended or compositional instructions. These results indicate that although there is still room for improvement in physical deployment, **Arcadia** demonstrates markedly stronger generalization and reliability, underscoring its potential as a robust foundation for real-world embodied intelligence.

4.4. Ablation and Analysis

Component Ablations. In Table 2, we conduct a detailed ablation study to analyze the contribution of each individual component within Arcadia. Specifically, we systematically replace one of its four key modules with a suboptimal alternative, allowing us to isolate and quantify the influence of each design choice on the overall system performance. The four components under examination are as follows: (1) the static training set, in which the model is trained using the ScaleVLN and RLBench datasets; (2) the retrieval-based scene reconstruction, where scenes are reconstructed using assets drawn from a limited asset library rather than a fully dynamic environment; (3) the joint training strategy, in which the VLM backbone is optimized through multi-task learning to support both VLN and VLA tasks; and (4) the sparse feedback mechanism, where the model receives only a binary success or failure signal as task feedback.

The results demonstrate that removing any single component significantly degrades overall performance, highlighting the importance of designing embodied intelligence systems that realistically capture the full lifecycle of interaction. Notably, the performance drop in case (3) is minimal, suggesting that the VLN and VLA tasks can share a unified VLM backbone. This finding motivates further exploration toward integrating VLN and VLA into a single framework, enabling unified execution of tasks that involve both navigation and manipulation.

		Success Rate (%)	
		VLN-CE-Isaac	LIBERO
0	Backbone	44.9	76.5
1	Static training set	43.0	72.9
2	Retrieval scene construction	46.1	81.4
3	Joint training	49.8	87.0
4	Sparse feedback	48.8	85.3
5	**Arcadia**	50.1	87.2

Table 2. Ablation study on each component.

5. Conclusion

We introduced **Arcadia**, a full-lifecycle framework that unifies real-world data collection, generative simulation, shared representation learning, and feedback-based adaptation into a closed loop for embodied lifelong learning. By tightly coupling these stages, **Arcadia** enables robots to continuously refine both their environments and policies, achieving improvements across navigation and manipulation tasks in simulation and real-world deployments. The framework demonstrates strong generalization, robustness, and scalability, highlighting its potential as a practical foundation for developing self-improving embodied agents and advancing the study of real-to-sim-to-real learning.

6. Limitations and Future Work

While **Arcadia** provides a comprehensive full-lifecycle framework for embodied intelligence, the current implementation is confined to the Unitree G1 platform and the Isaac Sim simulator. Hardware availability and cost constraints further limit empirical validation to a 7B-scale vision–language model, thereby narrowing the scope of large-scale evaluation. Future work will extend **Arcadia** to a broader range of embodiments and simulation environments, incorporating emerging open frameworks such as InternRobot to strengthen the coupling between robots, tasks, and environments. In practice, a substantial portion of our development effort has been devoted to unifying interfaces and configuring diverse environments. Accordingly, our primary motivation is to enable researchers to focus on model innovation rather than engineering complexity. To this end, we maintain open-source releases to promote reproducibility and foster community collaboration, allowing researchers to concentrate on advancing foundational models instead of managing implementation overhead.

7. Acknowledgement

This work was supported by the National Key Research and Development Program of China (2025ZD0123100), the Zhejiang NSF (LQK26F020001), Key R&D Program of Zhejiang (2026SDXT005), Fundamental Research Funds for the Central Universities (226-2025-00057), Ningbo Yongjiang Talent Introduction Programme (2024A-401-G), Zhejiang University Education Foundation Qizhen Scholar Foundation.

References

[1] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton Van Den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3674–3683, 2018. 2, 7

[2] OpenAI: Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Jozefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, et al. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020. 3

[3] Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. Scanqa: 3d question answering for spatial scene understanding. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19129–19139, 2022. 7

[4] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. 7

[5] Paul B Baltes. Theoretical propositions of life-span developmental psychology: On the dynamics between growth and decline. *Developmental psychology*, 23(5):611, 1987. 1

[6] Paul B Baltes. On the incomplete architecture of human ontogeny: Selection, optimization, and compensation as foundation of developmental theory. *American psychologist*, 52 (4):366, 1997. 1

[7] Homanga Bharadhwaj, Jay Vakil, Mohit Sharma, Abhinav Gupta, Shubham Tulsiani, and Vikash Kumar. Roboagent: Generalization and efficiency in robot manipulation via semantic augmentations and action chunking. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4788–4795. IEEE, 2024. 1, 3

[8] Konstantinos Bousmalis, Giulia Vezzani, Dushyant Rao, Coline Devin, Alex X Lee, Maria Bauza, Todor Davchev, Yuxiang Zhou, Agrim Gupta, Akhil Raju, et al. Robocat: A self-improving foundation agent for robotic manipulation. *arXiv preprint arXiv:2306.11706*, 1(8), 2023. 1, 3

[9] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3d: Learning from rgb-d data in indoor environments. *arXiv preprint arXiv:1709.06158*, 2017. 2

[10] Zhiyuan Chen and Bing Liu. *Lifelong machine learning*. Morgan & Claypool Publishers, 2018. 1

[11] An-Chieh Cheng, Yandong Ji, Zhaojing Yang, Zaitian Gongye, Xueyan Zou, Jan Kautz, Erdem Bıyık, Hongxu Yin, Sifei Liu, and Xiaolong Wang. Navila: Legged robot vision-language-action model for navigation. *arXiv preprint arXiv:2412.04453*, 2024. 1, 2, 3, 7

[12] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3366–3385, 2021. 3

[13] Donald Olding Hebb. *The organization of behavior: A neuropsychological theory*. Psychology press, 2005. 1

[14] Ahmed Jaafar, Shreyas Sundara Raman, Yichen Wei, Sofia Juliani, Anneke Wernerfelt, Benedict Quartey, Ifrah Idrees, Jason Xinyu Liu, and Stefanie Tellex. Lanmp: A language-conditioned mobile manipulation benchmark for autonomous robots. *arXiv preprint arXiv:2412.05313*, 2024. 2, 3

[15] Nick Jakobi, Phil Husbands, and Inman Harvey. Noise and the reality gap: The use of simulation in evolutionary robotics. In *European conference on artificial life*, pages 704–720. Springer, 1995. 3

[16] Stephen James, Zicong Ma, David Rovick Arrojo, and Andrew J Davison. Rlbench: The robot learning benchmark & learning environment. *IEEE Robotics and Automation Letters*, 5(2):3019–3026, 2020. 2

[17] Khimya Khetarpal, Matthew Riemer, Irina Rish, and Doina Precup. Towards continual reinforcement learning: A review and perspectives. *Journal of Artificial Intelligence Research*, 75:1401–1476, 2022. 3

[18] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024. 7

[19] David A Kolb. *Experiential learning: Experience as the source of learning and development*. FT press, 2014. 1

[20] Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *European Conference on Computer Vision*, pages 104–120. Springer, 2020. 5, 6, 7

[21] Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. *arXiv preprint arXiv:2010.07954*, 2020. 2, 7

[22] Ulman Lindenberger and Martin Lövdén. Brain plasticity in human lifespan development: the exploration–selection–refinement model. *Annual Review of Developmental Psychology*, 1(1):197–222, 2019. 1

[23] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023. 7

[24] Yongsen Mao, Junhao Zhong, Chuan Fang, Jia Zheng, Rui Tang, Hao Zhu, Ping Tan, and Zihan Zhou. Spatiallm: Training large language models for structured indoor modeling. *arXiv preprint arXiv:2506.07491*, 2025. 4

[25] Yuan Meng, Zhenshan Bing, Xiangtong Yao, Kejia Chen, Kai Huang, Yang Gao, Fuchun Sun, and Alois Knoll. Preserving and combining knowledge in robotic lifelong reinforcement learning. *Nature Machine Intelligence*, pages 1–14, 2025. 2

[26] Ruaridh Mon-Williams, Gen Li, Ran Long, Wenqian Du, and Christopher G Lucas. Embodied large language models enable robots to complete complex tasks in unpredictable environments. *Nature Machine Intelligence*, pages 1–10, 2025. 1

[27] NVIDIA Corporation. Nvidia isaac ros: A collection of gpu-accelerated ros2 packages for autonomous robots. https://github.com/NVIDIA-ISAAC-ROS, 2025. 4

[28] NVIDIA Corporation. Isaac ros nvblox: Gpu-accelerated 3d reconstruction for ros 2. https://github.com/NVIDIA-ISAAC-ROS/isaac_ros_nvblox, 2025. 4

[29] NVIDIA Corporation. Nvidia isaac lab, 2025. Official product page. 2

[30] NVIDIA Corporation. Nvidia isaac sim, 2025. Official product page. 2, 5, 7

[31] Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024. 1

[32] Aishwarya Padmakumar, Jesse Thomason, Ayush Shrivastava, Patrick Lange, Anjali Narayan-Chen, Spandana Gella, Robinson Piramuthu, Gokhan Tur, and Dilek Hakkani-Tur. Teach: Task-driven embodied agents that chat. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2017–2025, 2022. 2, 3

[33] German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural networks*, 113:54–71, 2019. 1, 3

[34] Xue Bin Peng, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Sim-to-real transfer of robotic control with dynamics randomization. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 3803–3810. IEEE, 2018. 3

[35] Rolf Pfeifer and Josh Bongard. *How the body shapes the way we think: a new view of intelligence*. MIT press, 2006. 1

[36] Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635. JMLR Workshop and Conference Proceedings, 2011. 2

[37] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9339–9347, 2019. 2

[38] Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10740–10749, 2020. 2, 3

[39] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Cliport: What and where pathways for robotic manipulation. In *Conference on robot learning*, pages 894–906. PMLR, 2022. 2

[40] Jaclynn V Sullivan. Learning and embodied cognition: A review and proposal. *Psychology Learning & Teaching*, 17 (2):128–143, 2018. 1

[41] Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024. 1, 3

[42] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 23–30. IEEE, 2017. 3

[43] Unitree Robotics. Unitree g1 humanoid robot, 2025. 7

[44] Homer Rich Walke, Kevin Black, Tony Z Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning*, pages 1723–1736. PMLR, 2023. 5, 7

[45] Hanqing Wang, Jiahe Chen, Wensi Huang, Qingwei Ben, Tai Wang, Boyu Mi, Tao Huang, Siheng Zhao, Yilun Chen, Sizhe Yang, et al. Grutopia: Dream general robots in a city at scale. *arXiv preprint arXiv:2407.10943*, 2024. 1, 2, 3, 7

[46] Andrew D Wilson and Sabrina Golonka. Embodied cognition is not what you think it is. *Frontiers in psychology*, 4: 58, 2013. 1

[47] Xuan Xiao, Jiahang Liu, Zhipeng Wang, Yanmin Zhou, Yong Qi, Shuo Jiang, Bin He, and Qian Cheng. Robot learning in the era of foundation models: A survey. *Neurocomputing*, page 129963, 2025. 1

[48] Jiale Xu, Weihao Cheng, Yiming Gao, Xintao Wang, Shenghua Gao, and Ying Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024. 4

[49] Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR, 2023. 1