

HIGenNTO: Scalable Humanoid Interaction Generation via Noise-Space Trajectory Optimization

Lalit Jayanti¹, Kashu Yamazaki¹, Yuto Shibata^{2,3}, Kotaro Amaya^{2,3}, Katerina Fragkiadaki¹

¹Carnegie Mellon University ²Keio AI Research Center ³Keio University

Humanoid robots can acquire complex skills by imitating kinematic humanoid motion references, yet reliable references for contact-rich interactions remain difficult to obtain: motion capture deteriorates under occlusion and close physical contact, while retargeting introduces additional contact and geometric inconsistencies. We present **HIGenNTO**, a framework that synthesizes humanoid-scene interaction motion references by optimizing the initial noise of a pretrained text-conditioned motion model under sparse spatiotemporal and scene constraints. The same formulation satisfies desired contacts, avoids collisions, and maintains stable support while retaining the prior’s realism and temporal coherence, generating interaction motions from scratch and composing long-horizon behaviors stage-wise. Across robot-environment and robot-object tasks, **HIGenNTO** produces motions that can be executed by tracking policies in simulation and used to train depth-conditioned visuomotor policies operating solely from onboard sensing. We deploy these policies on a Unitree G1 across four contact-rich tasks. Finally, the task specifications themselves can be written by a coding agent, which proposes interaction tasks and compiles them into prompt, constraint, and scene programs, authoring three of our eight evaluated tasks and four further behaviors. Together, these results establish a scalable path from high-level task descriptions to physically executable humanoid interactions.

Website: <https://higennto.github.io>

1 Introduction

Humanoid robots have acquired remarkably complex skills by imitating kinematic 3D trajectories recovered from motion capture or monocular video [6, 23, 28], and inherit the limitations of that reference data. Most large-scale datasets record the body in isolation, omitting the scene geometry, object motion, and contacts that constitute interaction [12, 24]. Interaction is also the hardest case to capture: occlusion and close contact degrade pose estimation where precision matters most, so even datasets built for it [19, 47] contain missed contacts, interpenetrations, and temporal inconsistencies, which retargeting compounds.

We introduce HIGenNTO (Humanoid Interaction Generation via Noise-Space Trajectory Optimization), which synthesizes contact-rich references without interaction motion capture (Figure 1). **We cast interaction synthesis as constrained optimization over a generative motion prior:** HIGenNTO optimizes the initial noise of a pretrained text-conditioned motion model subject to sparse spatiotemporal constraints and differentiable scene objectives that specify what text and keyframes cannot: which body part contacts which object, where and when, and which regions to keep clear. Gradients traverse the whole sampling chain, so the trajectory adapts to contact, collision, support, and task geometry while staying on the prior’s learned manifold. A single formulation yields motions from a task specification and scene geometry alone, and can extend past the pretrained motion prior’s prediction horizon.

Optimizing diffusion noise has previously enabled motion editing and control of human characters, including navigation in a static scene [18, 53]. We extend this mechanism to humanoid interaction synthesis, where contacts must not only be avoided but also deliberately established and maintained with support surfaces, manipulated objects, and other agents.

The specification for generating motions itself can be automated: from a short brief and the corpus annotations, a coding agent sizes the scene and writes the program to generate the spatiotemporal constraints. Further, these generated references can be executed by tracking policies and used to train depth-conditioned visuomotor policies operating solely from onboard sensing.

Contributions. (1) Interaction synthesis as optimization in noise space. We make a pretrained, scene-unaware motion prior contact- and geometry-aware without retraining it, achieving lower scene penetration across all three interaction



Figure 1. HIGenNTO synthesizes humanoid–scene interaction motion references through constrained optimization over a generative motion prior. Stair traversal, sitting and standing, and box pickup and placement across randomized scenes, without any motion capture of joint human–scene interaction.

categories and lower object-task hand-target error than both kinematic-constraint conditioning and classifier guidance. **(2) References that physical policies execute.** Across eight tasks, scene-aware teachers track the generated motion and distill into depth-only students that complete 57.4–98.3% of motions, where methods based on scene-unaware tracking formulations [23] largely fail. **(3) Task specification by a coding agent.** Agent-written programs supply three of our eight evaluated tasks and four further behaviors, with no per-task demonstration and no hand-tuned constraints.

2 Related Work

Human Motion Data and Human-to-Humanoid Retargeting. Large-scale motion datasets [12, 13, 24] underpin humanoid motion imitation and control [7, 28, 29], with collection scaled by video reconstruction [1, 20]. Most model humans in isolation, and those that do model interactions with the scene [14, 25, 41–43, 47, 52] still yield penetrations, contact inconsistencies, and implausible artifacts. Retargeting these motions to robots is an effective route to agile behaviors [3, 28, 34], via inverse kinematics, trajectory optimization, and learned retargeting [3, 9–11], including interaction-aware objectives [8, 15, 37, 45, 49, 50], but all depend on high-quality interaction motion that remains hard to obtain at scale. We instead synthesize such references from generative motion priors under sparse, semantically grounded constraints.

Generative Motion Priors for Humanoid Control. Diffusion-based motion models generate realistic, controllable motion conditioned on text, goals, or sparse constraints [32, 35, 38], and several works pair them with RL tracking controllers for physical realism at execution [34, 39, 48]. Adversarial motion priors such as AMP [30] and ASE [31] instead regularize policies toward realistic motion distributions without explicit trajectory tracking.

Controllable and Constraint-Guided Motion Generation. Diffusion models achieve spatial and semantic control through training-time conditioning on paired control signals such as trajectories or sparse constraints [17, 32, 35, 38, 46], through inference-time guidance by gradients of rewards, constraints, or value functions during denoising [16, 44, 54], or by optimizing the initial noise itself, backpropagating a differentiable objective through the entire sampling chain. We adopt the last: DNO [18] establishes it for editing, refinement, and obstacle avoidance, and DartControl [53] applies it to human–scene contact. Both operate on kinematic human characters against static geometry; we target humanoid robots, add objectives for contact held with movable objects and other agents, and train physical policies on the resulting references.

3 Method

Training a scene-aware interaction generator directly would require physically consistent interaction paired with time-varying scene geometry, which is hard to collect at scale. We instead inject scene-awareness at optimization time, optimizing the latent noise of a frozen text-conditioned motion diffusion model under differentiable contact, and scene

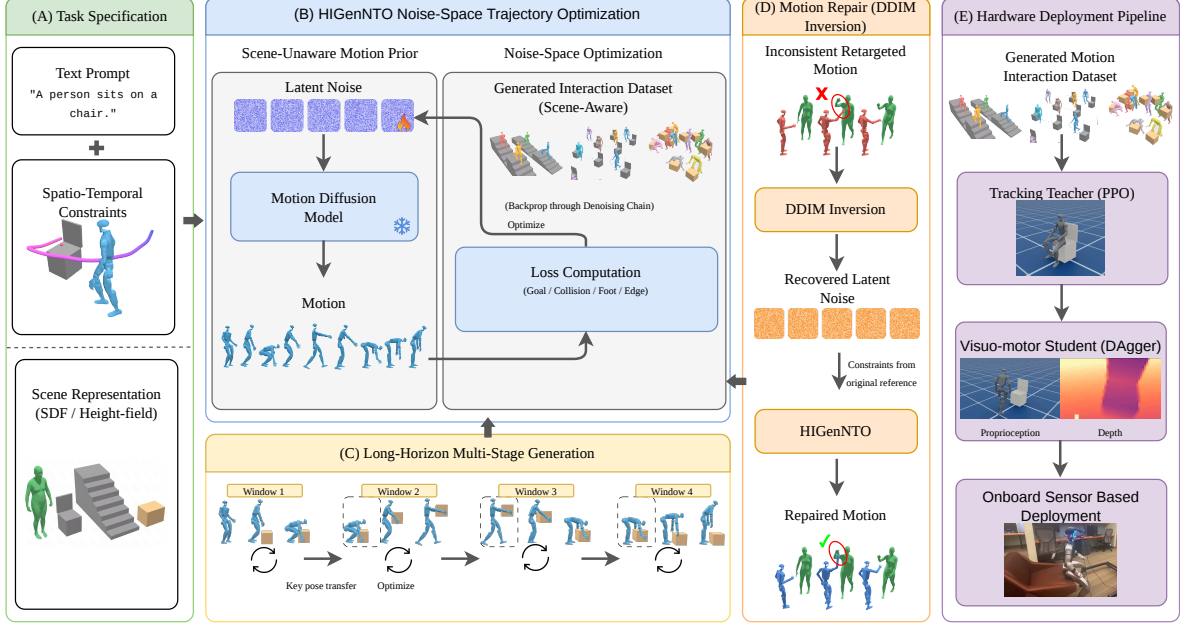


Figure 2. System overview. (A) **Task specification:** a text prompt, sparse spatiotemporal constraints on end-effector and root goals, and a scene representation (SDF / height-field). (B) **Noise-space optimization:** differentiable scene and contact losses (goal, collision, foot, edge) are backpropagated through the denoising chain of a frozen motion prior into its input latent noise, yielding a scene-aware interaction dataset. (C) **Long-horizon generation:** temporal windows optimized in sequence, with key poses transferred across boundaries. (D) **Motion repair:** an inconsistent retargeted motion sequence is inverted to latent noise and re-optimized. (E) **Deployment:** the dataset trains a PPO tracking teacher, distilled into a depth- and proprioception-conditioned student.

objectives (Figure 2). The synthesized motions stand in for motion-capture interaction data [40, 43], supervising a privileged tracking teacher that is distilled into a policy running from onboard sensing alone.

3.1 Scene-Aware Interaction Synthesis via Constrained Latent Optimization

We consider an articulated humanoid acting in a scene. A task is specified by: (i) a text prompt y ; (ii) a set of sparse spatiotemporal constraints $\mathcal{C} = \{(j_i, t_i, p_i)\}_{i=1}^{N_c}$ that constrain keypoint j_i to a target 3D position $p_i \in \mathbb{R}^3$ at time t_i ; and (iii) a scene representation $\mathcal{S}$ given by differentiable signed-distance functions (SDFs) for static and dynamic geometry, manipulated objects, and terrain height fields. Constraints are authored per task (by hand or, as we show in Sec. 4.3, by a coding agent), except for repair, where contact candidates proposed from the reference are filtered by a vision-language model (Appendix D). The goal is a motion trajectory $\mathbf{x} \in \mathbb{R}^{T \times J \times 3}$ that is semantically consistent with y , satisfies $\mathcal{C}$, respects scene geometry, and remains temporally smooth.

Scene-Unaware Humanoid Motion Prior. We use a pretrained text-conditioned humanoid motion diffusion model [32] as a differentiable motion prior, $M_\theta(\mathbf{z} | y, \mathcal{C}) = \mathbf{x}$, mapping latent noise $\mathbf{z}$, a text prompt y , and optional sparse kinematic constraints $\mathcal{C}$ to a trajectory $\mathbf{x}$. It is trained on over 700 hours of human motion retargeted to humanoid embodiments, without explicit scene or interaction input. The prior can be conditioned on kinematic constraints (full-body keyframes, sparse joint positions/rotations, and dense root paths) passed to the model as inputs. But interaction requirements such as non-penetration, stable contact, and collision avoidance are *implicit*: they depend on scene geometry and whole-body pose, cannot be written as keypoint targets in advance, and conditioning on these constraints alone does not yield contact-consistent motion (Sec. 4).

Noise-Space Optimization. We optimize over the prior’s latent variables rather than the 3D joint-position trajectory $\mathbf{x}$ directly, so every candidate remains a sample the prior can produce, retaining its learned realism and temporal coherence. Given latent noise $\mathbf{z} \in \mathbb{R}^{T \times D}$ unrolled through a differentiable DDIM sampling process, we solve

$$\min_{\mathbf{z}} \mathcal{L}(M_\theta(\mathbf{z} | y, \mathcal{C}), \mathcal{C}, \mathcal{S}) = w_g \mathcal{L}_{\text{goal}} + w_c \mathcal{L}_{\text{coll}} + w_f \mathcal{L}_{\text{foot}} + w_h \mathcal{L}_{\text{hand}} + w_e \mathcal{L}_{\text{edge}}, \quad (1)$$

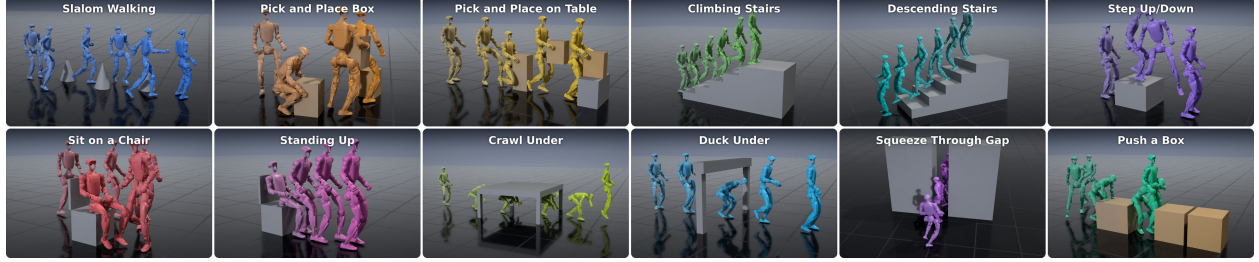


Figure 3. Generated interaction motions across the task suite. Each panel overlays frames from a single generated motion, labeled in-panel: the eight tasks we evaluate, together with four further behaviors whose task programs were written by the coding agent of Sec. 4.3.

The goal term $\mathcal{L}_{\text{goal}}$ is a masked loss on the constrained keypoints; the remaining terms are the scene-dependent collision, foot-contact, hand-contact, and edge-safety objectives; the last two are active only on tasks with grasps or terrain. We backpropagate through the entire denoising trajectory and optimize $\mathbf{z}$ with Adam; loss definitions and other details are in Appendix G.

Behavior Diversity through Augmentations. For a fixed task specification, we generate diverse motions by randomizing initial poses, approach trajectories, and latent noise seeds (Figure 3).

3.2 Long-Horizon Multi-Stage Interaction Generation

Single-window optimization is limited by the prior’s temporal horizon and by GPU memory. We therefore decompose long behaviors into temporal windows $\{(y_w, \mathcal{C}_w)\}_{w=1}^W$ optimized in sequence: the decoded suffix of window w is injected as a hard pose constraint into the prefix of window $w+1$ after re-canonicalization to the new heading and origin, composing long-horizon behavior without global optimization.

3.3 Teacher-Student Visuomotor Learning

Motion-Tracking Teacher. For each task we train a privileged teacher on the 29-DoF Unitree G1 with PPO [33], through the training stack of [2], tracking the full batch of motions our framework generates for that task (Figure 2E). It observes proprioceptive state together with privileged reference and scene information and outputs 29-D joint-angle residuals executed through PD control.

Depth-Conditioned Flow Student. We distill the teacher into a student operating only from onboard observations: a history of head-mounted depth images and proprioception, encoded into a sensor context $\mathbf{c}$. Because the same context can be consistent with several valid interaction behaviors, we model the teacher’s action distribution rather than its conditional mean, regressing a conditional velocity field with flow matching [21, 22]. Distillation is on-policy and at test time the field is integrated with a few fixed Euler steps; objective, architecture, and training details for both policies are in Appendix H.

4 Experiments

We evaluate whether HIGenNTO can generate humanoid–scene interaction motion references and whether those references provide effective supervision for humanoid control. Our experiments address four questions: **(1)** Does noise-space optimization produce more contact- and scene-consistent motions than alternative methods of controlling the same prior (Sec. 4.1)? **(2)** Can the generated references be executed by physical policies and distilled to policies that use onboard sensing (Sec. 4.2)? **(3)** Does tracking the generated references supervise better than task-based rewards, hierarchical control, or adversarial imitation (Sec. 4.2)? **(4)** Can a coding agent reduce the engineering of extending HIGenNTO to new interactions (Sec. 4.3)? We additionally present qualitative real-world deployments in videos available on the project page.

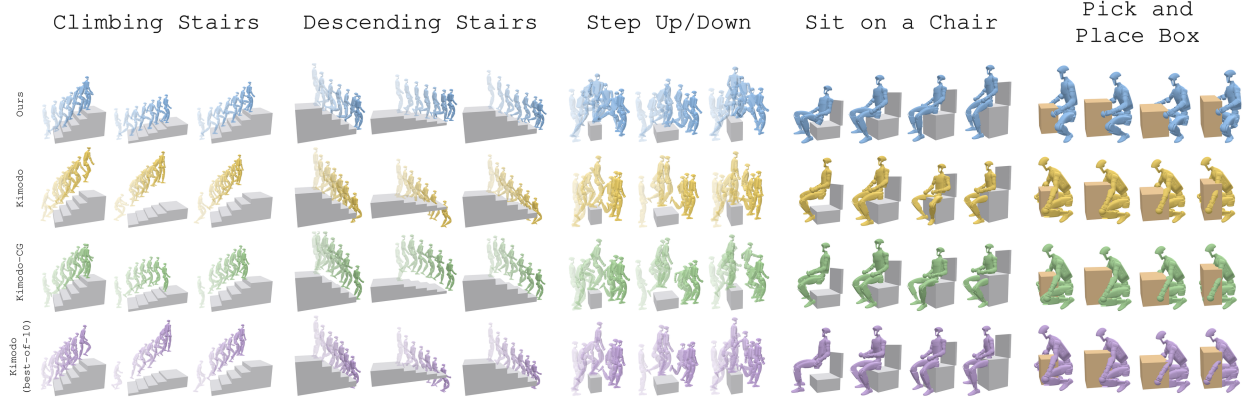


Figure 4. Humanoid–scene interaction synthesis with HIGenNTO and baseline methods. Rows, top to bottom: HIGenNTO (ours, blue), kinematic-constraint conditioning (Kimodo, yellow), classifier guidance (Kimodo-CG, green), and best-of-ten sampling (Kimodo best-of-10, purple); each column shows the identical scene, spawn, and constraint set. Stair and step panels overlay time samples of one motion sequence (opacity increases with time); chair panels show the final seated frame, and box panels the frame after grasping.

4.1 Humanoid–Scene Interaction Generation

Tasks. We evaluate motion generation on eight tasks spanning three forms of interaction (Table 1, Appendix C): **free-space locomotion** (slalom walking among obstacles), **object interaction** (picking up, carrying, and placing boxes on the floor or a table), and **terrain interaction** (climbing and descending stairs, stepping onto and off blocks, sitting on and standing from chairs), randomizing scene geometry, object placement, interaction timing, and initial conditions throughout.

Baselines. We compare HIGenNTO with three alternative mechanisms for controlling the same pretrained motion prior. All methods share prompts, random seeds, initial conditions, scene representation $\mathcal{S}$, and kinematic constraints $\mathcal{C}$. **Kimodo** samples the pretrained text-conditioned prior [32] conditioned on the kinematic constraints $\mathcal{C}$, without access to the scene-dependent objectives defined by $\mathcal{S}$. **Kimodo-CG** receives the same constraints and scene information as HIGenNTO, but applies the complete interaction objective as classifier guidance at each denoising step. **HIGenNTO** instead optimizes the initial noise by backpropagating the objective through the complete denoising trajectory. **Kimodo best-of-10** draws ten independent samples per task specification and retains the one with the lowest interaction objective.

Metrics. We measure target satisfaction (**root path error**, **hand target error** at constrained frames) and geometric consistency (**scene penetration** over body query points, **foot–support gap** at the lower foot sole). The last two measure separate failure modes: a motion can close the gap by penetrating, or avoid penetration by floating. We report mean $\pm$ standard deviation over motion sequences.

Results. Per-task results are in Table 1, with full metric definitions in Appendix C. HIGenNTO achieves lower scene penetration across all three interaction categories, and lowers object-task hand-target error compared to the baselines.

Kimodo-CG is an informative comparison because it receives the same constraints, scene representation, and interaction objective as HIGenNTO. Classifier guidance improves collision avoidance in free-space and terrain tasks but barely helps on object interaction, where dense root and hand conditioning leaves little room for the local corrections guidance can make.

Best-of-ten selection barely narrows the gap: matched sampling compute does not help, because the prior rarely samples a candidate satisfying every interaction requirement at once, which noise-space optimization instead constructs.

The box-pickup task shows this most clearly (Fig. 4). Across repeated samples and varying box sizes, the pretrained prior produces nearly identical wrist configurations and grasp widths, while HIGenNTO adapts the arm trajectory, wrist configuration, and hand separation to each object’s geometry.

Long-Horizon Multi-Stage Generation. We apply the multi-stage procedure from Sec. 3 to generate a four-stage pick-and-place sequence (Figure 2C): each stage is optimized with its own prompt and constraint set, with terminal poses transferred between consecutive windows.

Motion Repair. The same optimization can repair imperfect retargeted human–humanoid interaction references when initialized from DDIM-inverted noise (Figure 2D); setup and qualitative results are in Appendix D.

4.2 Learning Interaction Humanoid Policies from Generated References

Do generated references supervise humanoid control better than task-level reinforcement-learning objectives?

HIGenNTO learns humanoid control by training privileged, scene-aware teacher policies to track its generated interaction motion references, which are then distilled into depth-conditioned students (Sec. 4.2). Prior approaches instead optimize humanoid actions directly by reinforcement learning under task rewards, regularizing exploration with adversarial scene-unaware motion priors [30] or building on pretrained low-level controllers [4]. We compare these paradigms on *Sit on a Chair* using a shared set of 20 HIGenNTO-generated motions.

Baselines. We compare HIGenNTO-Teacher policies that track the complete generated trajectories, with four approaches:

TaskReward optimizes task rewards adapted from the *Sit Down* task of [40], with no motion prior, reference supervision, or contact and temporal supervision. **HR-TaskReward** trains a hierarchical policy under the same task rewards while using a frozen Homie controller [4] as its low-level policy. **AMP** [30] augments the task reward with an adversarial motion objective. Its discriminator is trained on the 20 generated motion references *excluding the scene*. **HIGenNTO-AMPScene** additionally conditions the discriminator on chair-relative state from the generated references, providing interaction-aware adversarial supervision without trajectory tracking.

All methods use task goals and initial states derived from the same generated motions; AMP and HIGenNTO-AMPScene additionally train their discriminators on them, and HIGenNTO-Teacher tracks them directly.

Results. Figure 5 shows that the HIGenNTO-Teacher policy reaches high task success in approximately one third of the training iterations required by HIGenNTO-AMPScene, the strongest baseline. TaskReward fails to reliably discover the approach–contact–stabilization sequence from the task objective alone; HR-TaskReward and AMP add a pretrained controller and motion realism, respectively, but neither receives explicit supervision for the humanoid–chair interaction. Conditioning the discriminator on chair-relative state substantially improves HIGenNTO-AMPScene, confirming that the generated references are also useful as a scene-aware motion prior; it still learns more slowly than HIGenNTO-Teacher, whose tracking objective supplies a time-aligned target throughout the interaction.

The generated references thus carry temporal and geometric interaction structure that shaped task rewards and motion-only regularization do not recover. Fig. 5 reports one run per method, and shows evidence of improved learning efficiency in this setting.

Can the generated interaction motions be executed without privileged information at test time? We compare HIGenNTO privileged teacher and depth-based student policies in Table 2 (Appendix E). For context, we also evaluate SONIC [23], a general-purpose motion tracker, zero-shot from its released weights: SONIC receives the motion reference but cannot observe the surrounding geometry, whereas the HIGenNTO student observes the scene through onboard sensing but receives no motion reference, testing whether reference tracking alone suffices for interaction. All policies face the same eight tasks, scene configurations, and task-success criteria; the student, having no reference, is scored on task success only. Per-task results and complete metric definitions are provided there.

Results. The privileged teachers execute the generated references reliably, and after distillation the students retain high success across all three interaction categories (Table 2) while operating without privileged state or motion references; that interaction structure transfers to policies acting from onboard sensing alone. The student’s long observation history matters on longer-horizon tasks: on *Pick* and *Place on Table*, the table is often occluded once the box is lifted, and the history lets the student retain the table position observed earlier in the rollout.

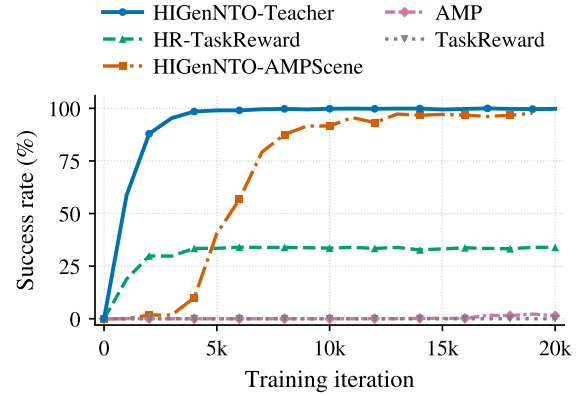


Figure 5. Policy learning on *Sit on a Chair*. Task success over reinforcement-learning iterations for generated reference-based motion tracking by HIGenNTO-Teacher policies and four alternative training objectives. Each curve represents one training run.

SONIC occasionally attains lower joint or root-orientation error, but its high root-position error and low task success show that scene-blind tracking reproduces plausible pose sequences without aligning them to the relevant object or support surface.

4.3 Scaling HIGenNTO by Automating Task Specifications with a Coding Agent

Each task in Sec. 3.1 is defined by a prompt y , constraints $\mathcal{C}$, and a scene representation $\mathcal{S}$; authoring and tuning these for every new interaction is a bottleneck. We investigate whether a coding agent can compile a natural-language task description into an executable HIGenNTO specification.

Setup. Our implementation exposes a single task-level interface: every task is a function that takes a few randomization parameters (e.g. chair height, box size) and returns $(y, \mathcal{C}, \mathcal{S})$, while the optimizer and the losses of Eq. 1 stay fixed. Given a short brief and the prior’s training-corpus annotations, the agent reads the corpus and writes a prompt y framed in a similar way to the text used to train the motion prior, probes the frozen prior to size the scene $\mathcal{S}$ and constraints $\mathcal{C}$, drafts the task function, and renders a small batch to verify and revise it; a human only reviews the final motions with free-form feedback. The full procedure is detailed in Appendix F.2.

Results. Three of the eight tasks evaluated in Sec. 4.1 (Slalom Walking, Step Up/Down, and Pick and Place on Table) were specified through this procedure. In a more open-ended setting with no target task, the agent identifies capabilities represented in the corpus annotations and proposes compatible scene interactions; Figure 3 shows four resulting tasks (Crawl Under, Duck Under, Squeeze Through Gap, Push a Box), spanning whole-body and non-prehensile object interaction with neither task-specific demonstrations nor manually authored constraints. Additional task definitions and qualitative results are provided in Appendix F.2.

4.4 Real-World Deployment

We deploy depth-conditioned student policies for Sit on a Chair, Standing Up, Pick and Place on Table, and Pick and Place Box on a Unitree G1 using onboard depth and proprioception. Videos of the deployments and qualitative simulation results are available on the project page; Appendix A shows the rollouts.

5 Conclusion and Limitations

We presented HIGenNTO, a framework for scalable humanoid interaction synthesis through constraint-guided latent optimization of pretrained motion priors. Rather than relying on large-scale interaction motion capture, HIGenNTO generates interaction-rich humanoid motions from sparse differentiable objectives encoding contacts, geometry, and temporal interaction structure, and the resulting motions serve both as interaction motion references for humanoid control and as supervision for deployable visuomotor policies. Our results suggest that interaction-rich behaviors need not be explicitly represented in the training distribution of a motion prior; they can emerge through optimization-time interaction constraints.

Our limitations suggest directions for future work. First, the generation objectives enforce kinematic and geometric consistency rather than dynamic feasibility; incorporating dynamics could directly produce motions satisfying balance, contact-force, and actuation constraints. Second, noise-space optimization remains computationally expensive and thus suited to offline data generation; flow-map-based optimization or amortized generation could reduce this cost. Finally, the framework addresses whole-body motion with simplified hand-object contact; dexterous manipulation will require richer hand representations, contact models, and object-dynamics objectives.

References

- [1] Arthur Allshire, Hongsuk Choi, Junyi Zhang, David McAllister, Anthony Zhang, Chung Min Kim, Trevor Darrell, Pieter Abbeel, Jitendra Malik, and Angjoo Kanazawa. Visual imitation enables contextual humanoid control, 2025. URL <https://arxiv.org/abs/2505.03729>.
- [2] Amazon FAR, Pieter Abbeel, Juyue Chen, Rocky Duan, Alejandro Escontrela, Manan Gandhi, Samuel Gundry, Xiaoyu Huang, Angjoo Kanazawa, Tomasz Lewicki, Jiaman Li, Karen Liu, Clay Rosenthal, Younggyo Seo, Carlo

- Sferrazza, Guanya Shi, Linda Shih, Jonathan Tseng, Zhen Wu, Lujie Yang, Brent Yi, and Yuanhang Zhang. Holosoma. <https://github.com/amazon-far/holosoma>, 2025.
- [3] Joao Pedro Araujo, Yanjie Ze, Pei Xu, Jiajun Wu, and C. Karen Liu. Retargeting matters: General motion retargeting for humanoid motion tracking, 2025. URL <https://arxiv.org/abs/2510.02252>.
 - [4] Qingwei Ben, Feiyu Jia, Jia Zeng, Juntong Dong, Dahua Lin, and Jiangmiao Pang. Homie: Humanoid locomanipulation with isomorphic exoskeleton cockpit, 2025. URL <https://arxiv.org/abs/2502.13013>.
 - [5] Bones Studio. BONES-SEED: Skeletal everyday embodiment dataset. <https://huggingface.co/datasets/bones-studio/seed>, 2026.
 - [6] Zixuan Chen, Mazeyu Ji, Xuxin Cheng, Xuanbin Peng, Xue Bin Peng, and Xiaolong Wang. Gmt: General motion tracking for humanoid whole-body control, 2025. URL <https://arxiv.org/abs/2506.14770>.
 - [7] Xuxin Cheng, Yandong Ji, Junming Chen, Ruihan Yang, Ge Yang, and Xiaolong Wang. Expressive whole-body control for humanoid robots, 2024. URL <https://arxiv.org/abs/2402.16796>.
 - [8] Théo Cheynel, Thomas Rossi, Baptiste Bellot-Gurlet, Damien Rohmer, and Marie-Paule Cani. Reconform : Real-time contact-aware motion retargeting for more diverse character morphologies, 2025. URL <https://arxiv.org/abs/2502.21207>.
 - [9] Sungjoon Choi and Joohyung Kim. Towards a natural motion generator: a pipeline to control a humanoid based on motion data. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4373–4380, 2019. doi: 10.1109/IROS40897.2019.8967941.
 - [10] Sungjoon Choi, Min Jae Song, Hyemin Ahn, and Joohyung Kim. Self-supervised motion retargeting with safety guarantee, 2021. URL <https://arxiv.org/abs/2103.06447>.
 - [11] Waldez Gomes, Vishnu Radhakrishnan, Luigi Penco, Valerio Modugno, Jean-Baptiste Mouret, and Serena Ivaldi. Humanoid whole-body movement optimization from retargeted human motions. In *2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*, pages 178–185, 2019. doi: 10.1109/Humanoids43949.2019.9035070.
 - [12] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5142–5151, 2022. doi: 10.1109/CVPR52688.2022.00509.
 - [13] Félix G. Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. Robust motion in-betweening. *ACM Transactions on Graphics*, 39(4), August 2020. ISSN 1557-7368. doi: 10.1145/3386569.3392480. URL <http://dx.doi.org/10.1145/3386569.3392480>.
 - [14] Mohamed Hassan, Duygu Ceylan, Ruben Villegas, Jun Saito, Jimei Yang, Yi Zhou, and Michael Black. Stochastic scene-aware motion prediction, 2021. URL <https://arxiv.org/abs/2108.08284>.
 - [15] Inseo Jang, Soojin Choi, Seokhyeon Hong, Chaelin Kim, and Junyong Noh. Geometry-aware retargeting for two-skinned characters interaction. *ACM Trans. Graph.*, 43(6), November 2024. ISSN 0730-0301. doi: 10.1145/3687962. URL <https://doi.org/10.1145/3687962>.
 - [16] Michael Janner, Yilun Du, Joshua B. Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis, 2022. URL <https://arxiv.org/abs/2205.09991>.
 - [17] Korrawe Karunratanakul, Konpat Preechakul, Supasorn Suwajanakorn, and Siyu Tang. Guided motion diffusion for controllable human motion synthesis, 2023. URL <https://arxiv.org/abs/2305.12577>.
 - [18] Korrawe Karunratanakul, Konpat Preechakul, Emre Aksan, Thabo Beeler, Supasorn Suwajanakorn, and Siyu Tang. Optimizing diffusion noise can serve as universal motion priors, 2024. URL <https://arxiv.org/abs/2312.11994>.

- [19] Jiaman Li, Jiajun Wu, and C. Karen Liu. Object motion guided human motion synthesis, 2023. URL <https://arxiv.org/abs/2309.16237>.
- [20] Jing Lin, Ailing Zeng, Shunlin Lu, Yuanhao Cai, Ruimao Zhang, Haoqian Wang, and Lei Zhang. Motion-x: A large-scale 3d expressive whole-body human motion dataset, 2024. URL <https://arxiv.org/abs/2307.00818>.
- [21] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling, 2023. URL <https://arxiv.org/abs/2210.02747>.
- [22] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow, 2022. URL <https://arxiv.org/abs/2209.03003>.
- [23] Zhengyi Luo, Ye Yuan, Tingwu Wang, Chenran Li, Fernando Castañeda, Sirui Chen, Zi-Ang Cao, Jiefeng Li, David Minor, Qingwei Ben, Jinhyung Park, David Sami, Zi Wang, Xingye Da, Runyu Ding, Cyrus Hogg, Lina Song, Edy Lim, Eugene Jeong, Tairan He, Haoru Xue, Wenli Xiao, Simon Yuen, Jan Kautz, Yan Chang, Umar Iqbal, Linxi "Jim" Fan, and Yuke Zhu. Sonic: Supersizing motion tracking for natural humanoid whole-body control, 2026. URL <https://arxiv.org/abs/2511.07820>.
- [24] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. Amass: Archive of motion capture as surface shapes, 2019. URL <https://arxiv.org/abs/1904.03278>.
- [25] Jiageng Mao, Siheng Zhao, Siqi Song, Tianheng Shi, Junjie Ye, Mingtong Zhang, Haoran Geng, Jitendra Malik, Vitor Guizilini, and Yue Wang. Learning from massive human videos for universal humanoid pose control, 2024. URL <https://arxiv.org/abs/2412.14172>.
- [26] Marko Mihajlovic, Siwei Zhang, Gen Li, Kaifeng Zhao, Lea Müller, and Siyu Tang. Volumetricmpl: A neural volumetric body model for efficient interactions, contacts, and collisions, 2025. URL <https://arxiv.org/abs/2506.23236>.
- [27] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [28] Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. Deepmimic: example-guided deep reinforcement learning of physics-based character skills. *ACM Transactions on Graphics*, 37(4):1–14, 2018. ISSN 1557-7368. doi: 10.1145/3197517.3201311. URL <http://dx.doi.org/10.1145/3197517.3201311>.
- [29] Xue Bin Peng, Angjoo Kanazawa, Jitendra Malik, Pieter Abbeel, and Sergey Levine. Sfv: Reinforcement learning of physical skills from videos, 2018. URL <https://arxiv.org/abs/1810.03599>.
- [30] Xue Bin Peng, Ze Ma, Pieter Abbeel, Sergey Levine, and Angjoo Kanazawa. Amp: adversarial motion priors for stylized physics-based character control. *ACM Transactions on Graphics*, 40(4):1–20, 2021. ISSN 1557-7368. doi: 10.1145/3450626.3459670. URL <http://dx.doi.org/10.1145/3450626.3459670>.
- [31] Xue Bin Peng, Yunrong Guo, Lina Halper, Sergey Levine, and Sanja Fidler. Ase: large-scale reusable adversarial skill embeddings for physically simulated characters. *ACM Transactions on Graphics*, 41(4):1–17, 2022. ISSN 1557-7368. doi: 10.1145/3528223.3530110. URL <http://dx.doi.org/10.1145/3528223.3530110>.
- [32] Davis Rempe, Mathis Petrovich, Ye Yuan, Haotian Zhang, Xue Bin Peng, Yifeng Jiang, Tingwu Wang, Umar Iqbal, David Minor, Michael de Ruyter, Jiefeng Li, Chen Tessler, Edy Lim, Eugene Jeong, Sam Wu, Ehsan Hassani, Michael Huang, Jin-Bey Yu, Chaeyeon Chung, Lina Song, Olivier Dionne, Jan Kautz, Simon Yuen, and Sanja Fidler. Kimodo: Scaling controllable human motion generation, 2026. URL <https://arxiv.org/abs/2603.15546>.
- [33] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL <https://arxiv.org/abs/1707.06347>.
- [34] Agon Serifi, Ruben Grandia, Espen Knoop, Markus Gross, and Moritz Bächer. Robot motion diffusion model: Motion generation for robotic characters. In *SIGGRAPH Asia 2024 Conference Papers*, SA '24, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400711312. doi: 10.1145/3680528.3687626. URL <https://doi.org/10.1145/3680528.3687626>.

- [35] Yonatan Shafir, Guy Tevet, Roy Kapon, and Amit H. Bermano. Human motion diffusion as a generative prior, 2023. URL <https://arxiv.org/abs/2303.01418>.
- [36] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. URL <https://arxiv.org/abs/2010.02502>.
- [37] Chen Tessler, Yifeng Jiang, Erwin Coumans, Zhengyi Luo, Gal Chechik, and Xue Bin Peng. Maskedmanipulator: Versatile whole-body manipulation, 2025. URL <https://arxiv.org/abs/2505.19086>.
- [38] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H. Bermano. Human motion diffusion model, 2022. URL <https://arxiv.org/abs/2209.14916>.
- [39] Guy Tevet, Sigal Raab, Setareh Cohan, Daniele Reda, Zhengyi Luo, Xue Bin Peng, Amit H. Bermano, and Michiel van de Panne. Cload: Closing the loop between simulation and diffusion for multi-task character control, 2024. URL <https://arxiv.org/abs/2410.03441>.
- [40] Huayi Wang, Wentao Zhang, Runyi Yu, Tao Huang, Junli Ren, Feiyu Jia, Zirui Wang, Xiaojie Niu, Xiao Chen, Jiahe Chen, Qifeng Chen, Jingbo Wang, and Jiangmiao Pang. Physshi: Towards a real-world generalizable and natural humanoid-scene interaction system, 2025. URL <https://arxiv.org/abs/2510.11072>.
- [41] Yinhuai Wang, Jing Lin, Ailing Zeng, Zhengyi Luo, Jian Zhang, and Lei Zhang. Physshoi: Physics-based imitation of dynamic human-object interaction, 2023. URL <https://arxiv.org/abs/2312.04393>.
- [42] Yinhuai Wang, Qihan Zhao, Runyi Yu, Hok Wai Tsui, Ailing Zeng, Jing Lin, Zhengyi Luo, Jiwen Yu, Xiu Li, Qifeng Chen, Jian Zhang, Lei Zhang, and Ping Tan. Skillmimic: Learning basketball interaction skills from demonstrations, 2025. URL <https://arxiv.org/abs/2408.15270>.
- [43] Yinhuai Wang, Qihan Zhao, Yuen Fui Lau, Runyi Yu, Hok Wai Tsui, Qifeng Chen, Jingbo Wang, Jiangmiao Pang, and Ping Tan. Humanx: Toward agile and generalizable humanoid interaction skills from human videos, 2026. URL <https://arxiv.org/abs/2602.02473>.
- [44] Zhendong Wang, Jonathan J Hunt, and Mingyuan Zhou. Diffusion policies as an expressive policy class for offline reinforcement learning, 2023. URL <https://arxiv.org/abs/2208.06193>.
- [45] Haoyang Weng, Yitang Li, Nikhil Sobanbabu, Zihan Wang, Zhengyi Luo, Tairan He, Deva Ramanan, and Guanya Shi. Hdmi: Learning interactive humanoid whole-body control from human videos, 2025. URL <https://arxiv.org/abs/2509.16757>.
- [46] Yiming Xie, Varun Jampani, Lei Zhong, Deqing Sun, and Huaizu Jiang. Omnicontrol: Control any joint at any time for human motion generation, 2024. URL <https://arxiv.org/abs/2310.08580>.
- [47] Liang Xu, Xintao Lv, Yichao Yan, Xin Jin, Shuwen Wu, Congsheng Xu, Yifan Liu, Yizhou Zhou, Fengyun Rao, Xingdong Sheng, Yunhui Liu, Wenjun Zeng, and Xiaokang Yang. Inter-x: Towards versatile human-human interaction analysis, 2023. URL <https://arxiv.org/abs/2312.16051>.
- [48] Michael Xu, Yi Shi, KangKang Yin, and Xue Bin Peng. Parc: Physics-based augmentation with reinforcement learning for character controllers. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, SIGGRAPH Conference Papers '25, page 1–11. ACM, 2025. doi: 10.1145/3721238.3730616. URL <http://dx.doi.org/10.1145/3721238.3730616>.
- [49] Sirui Xu, Hung Yu Ling, Yu-Xiong Wang, and Liang-Yan Gui. Intermimic: Towards universal whole-body control for physics-based human-object interactions, 2026. URL <https://arxiv.org/abs/2502.20390>.
- [50] Lujie Yang, Xiaoyu Huang, Zhen Wu, Angjoo Kanazawa, Pieter Abbeel, Carmelo Sferrazza, C. Karen Liu, Rocky Duan, and Guanya Shi. Omniretarget: Interaction-preserving data generation for humanoid whole-body loco-manipulation and scene interaction, 2026. URL <https://arxiv.org/abs/2509.26633>.
- [51] Shaofeng Yin, Yanjie Ze, Hong-Xing Yu, C. Karen Liu, and Jiajun Wu. Visualmimic: Visual humanoid loco-manipulation via motion tracking and generation, 2025. URL <https://arxiv.org/abs/2509.20322>.

- [52] Xiaohan Zhang, Bharat Lal Bhatnagar, Vladimir Guzov, Sebastian Starke, and Gerard Pons-Moll. Couch: Towards controllable human-chair interactions, 2022. URL <https://arxiv.org/abs/2205.00541>.
- [53] Kaifeng Zhao, Gen Li, and Siyu Tang. Dartcontrol: A diffusion-based autoregressive motion model for real-time text-driven motion control, 2025. URL <https://arxiv.org/abs/2410.05260>.
- [54] Yanan Zheng, Ruiming Liang, Kexin Zheng, Jinliang Zheng, Liyuan Mao, Jianxiong Li, Weihao Gu, Rui Ai, Shengbo Eben Li, Xianyuan Zhan, and Jingjing Liu. Diffusion-based planning for autonomous driving with flexible guidance, 2025. URL <https://arxiv.org/abs/2501.15564>.

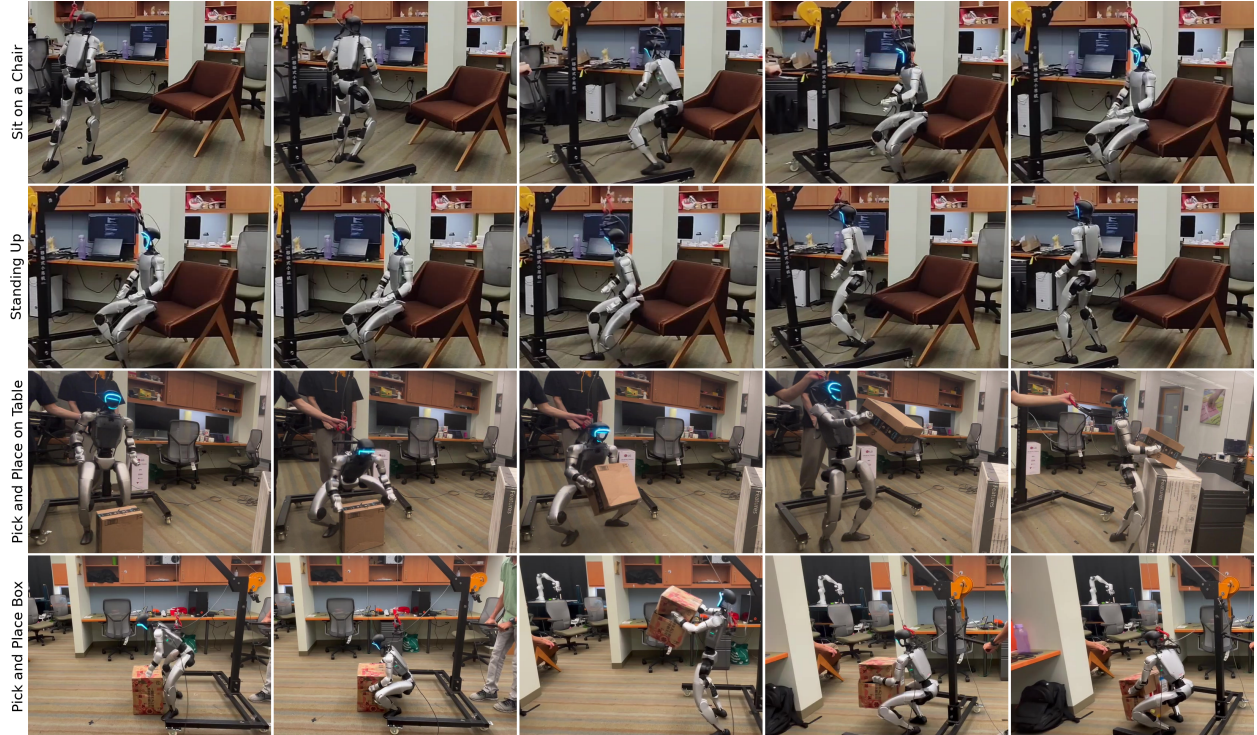


Figure 6. Real-world deployment of the distilled students, using only onboard sensor observations. The robot approaches a chair and sits down (Sit on a Chair); stands back up from the seated state (Standing Up); lifts a box from the floor onto a table (Pick and Place on Table); and picks up and carries a box (Pick and Place Box).

A Real-world deployment

Figure 6 shows the distilled student running on the physical robot for four tasks, from onboard sensing alone.

B Student evaluation in simulation

Figure 7 shows the qualitative results for the trained students in simulation on all eight tasks, with depth and proprioception the policy’s only inputs.

C Motion-generation results

C.1 Per-task results

Comparison setup. Table 1 reports the motion-generation results per task. All four methods are tested on an identical scene and constraint set. The baselines are sampled with 50 DDIM denoising steps, while HIGenNTO uses 10 steps to limit the memory and compute required for backpropagation through the denoising process; Kimodo best-of-10 draws ten samples per motion sequence from the motion prior and keeps the one scoring lowest under the weighted objective of Appendix G.2.

Runtime and compute. One noise-optimization iteration of HIGenNTO backpropagates through a 10-step DDIM process (Alg. 1). On a single RTX A6000 it takes 9.7–26 s at batch size 20 per iteration. The motions are optimized for $K=50$ –100 iterations.

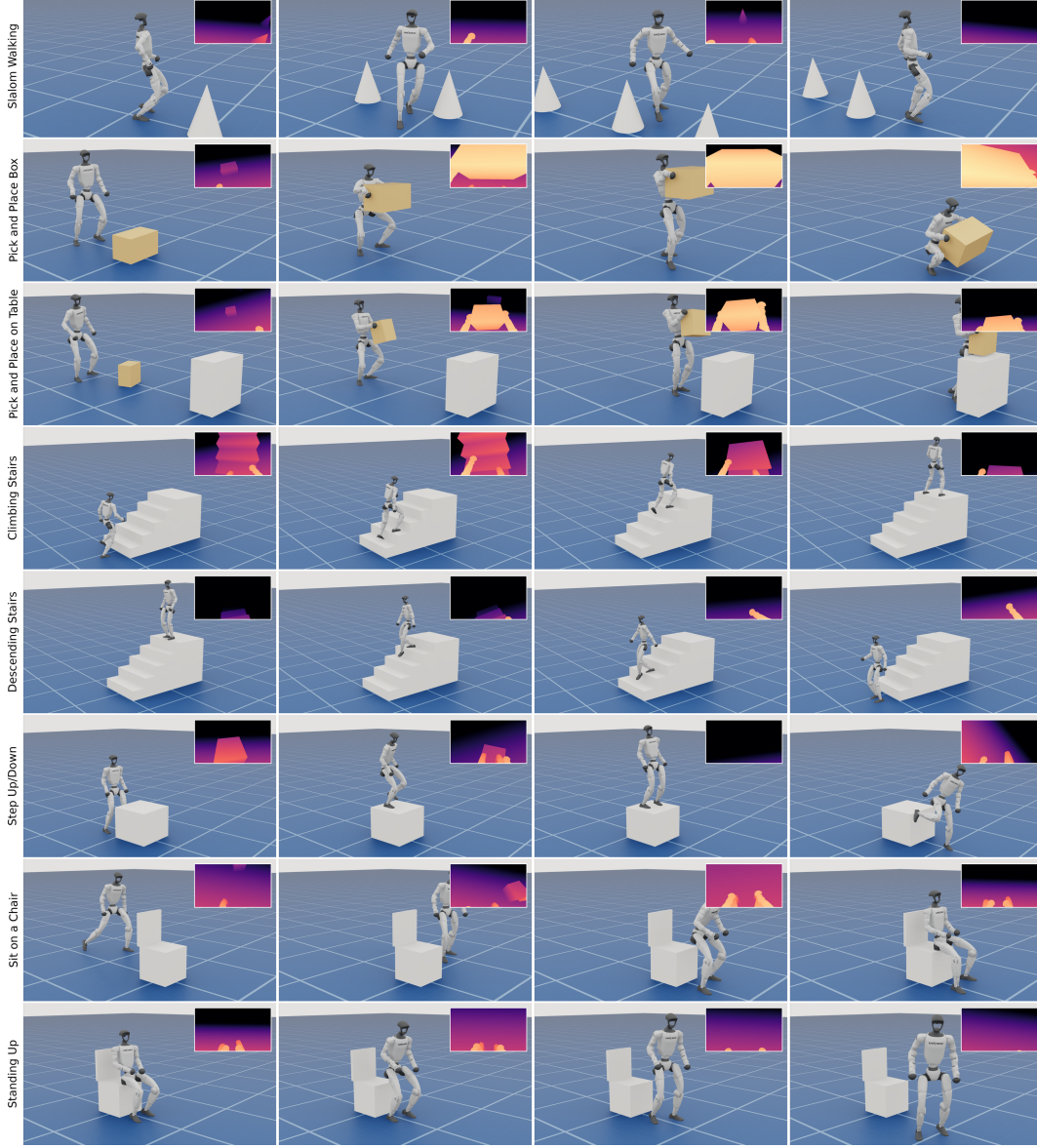


Figure 7. Evaluation of the trained student policies in simulation. The inset in each frame is the depth image the policy receives at that instant.

C.2 Evaluation metrics

This section defines the four metrics of Table 1. All are reported in centimeters and lower is better. $\mathbf{x}_j^t \in \mathbb{R}^3$ is the world position of joint j at frame t of a generated motion sequence, over T frames and J skeleton joints. The contact threshold τ is the clearance within which a query point counts as touching a surface. The skin distance δ_j is the clearance each joint is expected to keep clear of static geometry. Against a moving obstacle the corresponding clearance is δ_j^{mov} .

Root path error. The mean Euclidean distance, in the world ground plane, between the generated smoothed root position $\mathbf{r}^t$ and its target $\hat{\mathbf{r}}^t$, evaluated only at the constrained frames $\mathcal{T}_r$:

$$E_{\text{root}} = \frac{1}{|\mathcal{T}_r|} \sum_{t \in \mathcal{T}_r} \|\Pi_{xy} \mathbf{r}^t - \hat{\mathbf{r}}^t\|_2, \quad (2)$$

Method	Free	Object		Terrain				
	Slalom Walking	Pick and Place Box	Pick and Place on Table	Climbing Stairs	Descending Stairs	Step Up/Down	Sit on a Chair	Standing Up
<i>Root path error (cm) ↓</i>								
Ours	4.53 ± 0.82	2.65 ± 0.26	2.73 ± 0.53	2.78 ± 1.63	6.70 ± 3.40	3.89 ± 0.61	3.86 ± 0.74	2.87 ± 0.29
Kimodo	7.23 ± 3.20	2.93 ± 0.65	2.29 ± 0.51	34.00 ± 27.82	16.41 ± 7.50	2.86 ± 0.33	4.61 ± 1.15	2.96 ± 0.23
Kimodo-CG	7.23 ± 2.84	2.58 ± 0.32	2.10 ± 0.35	10.29 ± 8.85	12.81 ± 5.57	2.90 ± 0.66	4.49 ± 0.99	2.95 ± 0.26
Kimodo best-of-10	7.22 ± 3.43	2.48 ± 0.48	2.30 ± 0.47	28.25 ± 19.41	17.06 ± 4.38	2.76 ± 0.20	4.46 ± 1.35	3.14 ± 0.19
<i>Hand target error (cm) ↓</i>								
Ours	–	6.36 ± 2.26	6.61 ± 2.67	–	–	–	–	–
Kimodo	–	18.93 ± 2.18	21.25 ± 1.78	–	–	–	–	–
Kimodo-CG	–	19.18 ± 2.23	20.60 ± 1.48	–	–	–	–	–
Kimodo best-of-10	–	18.29 ± 2.37	19.97 ± 1.59	–	–	–	–	–
<i>Scene penetration (cm) ↓</i>								
Ours	0.01 ± 0.02	0.58 ± 0.46	0.51 ± 0.47	1.10 ± 0.49	1.94 ± 0.62	0.08 ± 0.08	1.25 ± 0.57	1.38 ± 1.63
Kimodo	33.96 ± 4.11	7.07 ± 6.74	10.35 ± 7.51	0.06 ± 0.08	232.79 ± 164.20	8.45 ± 3.39	9.42 ± 10.17	115.73 ± 118.03
Kimodo-CG	2.99 ± 0.95	7.95 ± 6.50	10.62 ± 7.09	0.01 ± 0.02	1.63 ± 0.88	1.97 ± 2.32	6.04 ± 7.22	107.59 ± 111.51
Kimodo best-of-10	27.37 ± 2.84	4.99 ± 6.08	6.68 ± 5.70	0.04 ± 0.05	149.14 ± 103.53	4.63 ± 2.32	5.39 ± 6.26	86.75 ± 97.79
<i>Foot–support gap (cm) ↓</i>								
Ours	0.01 ± 0.04	2.60 ± 0.22	2.82 ± 0.22	2.31 ± 0.30	3.17 ± 1.21	0.46 ± 0.59	0.21 ± 0.14	0.19 ± 0.10
Kimodo	0.00 ± 0.01	3.77 ± 0.80	3.24 ± 1.51	51.22 ± 14.42	0.42 ± 0.27	2.69 ± 2.06	0.75 ± 1.18	1.62 ± 2.93
Kimodo-CG	0.01 ± 0.06	3.17 ± 0.73	2.96 ± 0.56	24.57 ± 4.92	18.78 ± 5.65	3.48 ± 1.90	0.47 ± 0.24	1.66 ± 2.91
Kimodo best-of-10	0.00 ± 0.00	5.25 ± 1.02	4.94 ± 1.27	39.46 ± 12.92	0.59 ± 0.30	2.99 ± 1.77	1.12 ± 1.21	1.92 ± 2.51

Table 1. Motion-generation ablation, per task, on our dataset. All four methods are provided identical scene and constraints for each sequence. **Bold** = best mean per metric. Metric definitions in Appendix C.2.

with Π_{xy} the projection onto the ground plane. Every task carries a root path constraint, so this is defined throughout; because that path is handed to all four methods, the metric reports how well each preserves a constraint it was given.

Hand target error. The mean Euclidean distance between a constrained hand joint j_e and its keyframe target $\hat{\mathbf{x}}_{j_e}^t$ over the specified keyframes $\mathcal{T}_e$:

$$E_{\text{hand}} = \frac{1}{|\mathcal{T}_e|} \sum_{t \in \mathcal{T}_e} \|\mathbf{x}_{j_e}^t - \hat{\mathbf{x}}_{j_e}^t\|_2. \quad (3)$$

It is defined only for tasks whose constraint set specifies hand end-effector targets, and reported as “–” elsewhere.

Scene penetration. The per-frame penetration depth into scene geometry, summed over body query points and averaged over the sequence. Writing $s(\cdot)$ for the signed distance to the static scene and $s_o(\cdot)$ for the signed distance to moving obstacle o ,

$$E_{\text{pen}} = \frac{1}{T} \sum_{t=1}^T \left[\sum_j \max(0, \delta_j - s(\mathbf{x}_j^t)) + \sum_{p \in \mathcal{P}} \max(0, \tau - s(\mathbf{x}_p^t)) + \sum_o \sum_j \max(0, \delta_j^{\text{mov}} - s_o(\mathbf{x}_j^t)) \right], \quad (4)$$

where $\mathcal{P}$ is the set of sole points of both feet; querying the sole rather than the ankle joint captures the actual surface of the feet. Joints gripping a moving obstacle are excluded while the object is held, so an intended grasp is not penalized, but other joints are penalized for penetrating the obstacle.

Foot–support gap. The mean height of the lower foot’s nearest sole point above the support surface, beyond a contact threshold:

$$E_{\text{gap}} = \frac{1}{T} \sum_{t=1}^T \max\left(0, \min_{p \in \mathcal{P}} s(\mathbf{x}_p^t) - \tau\right). \quad (5)$$

The minimum runs over the same $\mathcal{P}$, so it selects the lowest sole point of whichever foot is lower.

Why the last two are reported together. Scene penetration and foot–support gap are the two one-sided halves of a single signed clearance: a motion can drive penetration to zero by hovering above the scene entirely, and can close the support gap by penetrating through it. Both failure modes occur among the baselines (Table 1), which is why we report the pair.

Method	Free	Object		Terrain				
	Slalom Walking	Pick and Place Box	Pick and Place on Table	Climbing Stairs	Descending Stairs	Step Up/Down	Sit on a Chair	Standing Up
<i>Joint error (rad) ↓</i>								
Teacher (ours)	0.141 $\pm$ 0.014	0.182 $\pm$ 0.022	0.296 $\pm$ 0.065	0.087 $\pm$ 0.008	0.107 $\pm$ 0.011	0.099 $\pm$ 0.010	0.087 $\pm$ 0.008	0.068 $\pm$ 0.012
SONIC	0.231 $\pm$ 0.059	0.114 $\pm$ 0.009	0.100 $\pm$ 0.025	0.326 $\pm$ 0.128	0.220 $\pm$ 0.104	0.124 $\pm$ 0.050	0.115 $\pm$ 0.025	0.388 $\pm$ 0.138
<i>Root position error (m) ↓</i>								
Teacher (ours)	0.131 $\pm$ 0.047	0.076 $\pm$ 0.010	0.128 $\pm$ 0.069	0.056 $\pm$ 0.008	0.060 $\pm$ 0.009	0.059 $\pm$ 0.007	0.058 $\pm$ 0.008	0.062 $\pm$ 0.016
SONIC	1.153 $\pm$ 0.387	0.462 $\pm$ 0.080	0.526 $\pm$ 0.151	1.399 $\pm$ 0.297	0.763 $\pm$ 0.278	1.538 $\pm$ 0.670	0.341 $\pm$ 0.124	1.006 $\pm$ 0.174
<i>Root orientation error (rad) ↓</i>								
Teacher (ours)	0.221 $\pm$ 0.047	0.193 $\pm$ 0.033	0.301 $\pm$ 0.119	0.144 $\pm$ 0.021	0.168 $\pm$ 0.031	0.127 $\pm$ 0.022	0.145 $\pm$ 0.020	0.140 $\pm$ 0.034
SONIC	0.453 $\pm$ 0.167	0.153 $\pm$ 0.009	0.154 $\pm$ 0.060	0.747 $\pm$ 0.324	0.485 $\pm$ 0.269	0.224 $\pm$ 0.107	0.198 $\pm$ 0.041	0.864 $\pm$ 0.368
<i>Task success (%) ↑</i>								
Teacher (ours)	97.7 $\pm$ 2.1	99.5 $\pm$ 0.8	71.1 $\pm$ 34.8	99.7 $\pm$ 0.2	99.5 $\pm$ 0.5	100.0 $\pm$ 0.0	99.9 $\pm$ 0.1	97.6 $\pm$ 2.2
Student (ours)	65.9 $\pm$ 25.4	80.9 $\pm$ 18.0	57.4 $\pm$ 36.7	96.5 $\pm$ 5.2	95.2 $\pm$ 4.9	98.3 $\pm$ 1.6	88.5 $\pm$ 20.1	96.3 $\pm$ 4.2
SONIC	3.1 $\pm$ 4.9	8.0 $\pm$ 8.3	0.0 $\pm$ 0.0	0.2 $\pm$ 0.5	28.8 $\pm$ 23.0	0.2 $\pm$ 0.4	19.5 $\pm$ 22.6	3.9 $\pm$ 7.0

Table 2. Motion tracking, per task. **Bold** = best mean per metric. Setup and rubrics in Appendix E.2.

D Repairing retargeted interaction motions

The same optimization framework can repair an existing interaction trajectory by initializing from its inverted latent noise rather than from random noise. We demonstrate this on two Inter-X sequences [47], Hug and HighFive, retargeted with OmniRetarget [50]: in the resulting references the hands fail to meet during the high-five, while the hug shows body interpenetration without the intended surface contact (Figure 8, first row).

To construct the repair constraints we threshold inter-body distances in the original human-human sequence for candidate contact locations and times, then use a vision-language model to retain the interaction-relevant ones as $\mathcal{C}$. The human is represented by the SMPL-X [27] body SDF from VolumetricSMPL [26], and the latent is initialized by DDIM inversion [36] of the retargeted trajectory.

HIGenNTO then restores the intended contacts in both interactions while preserving the structure of the original motion (Figure 8, last row). Random initialization suffices for the simpler high-five but fails on the more tightly constrained hug (second row), indicating that inversion provides an important initialization when repairing interactions with dense contact requirements.

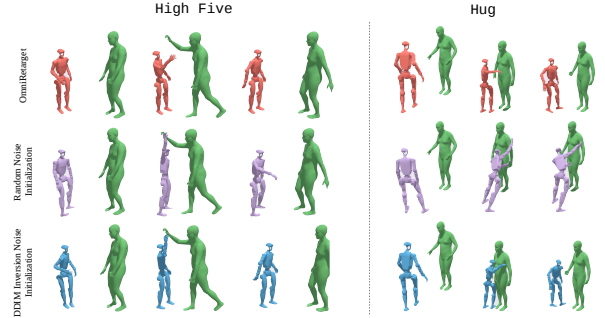


Figure 8. Motion reference repair. HIGenNTO initialized with DDIM-inverted noise.

E Motion-tracking results

E.1 Per-task results

Table 2 reports the motion-tracking results per task, aggregated by interaction type (free-space, object, terrain) in the main text. The two trackers are scored on every metric and the student, which receives no motion reference, is scored on task success alone.

E.2 Evaluation metrics and setup

Metrics. The three error metrics are accumulated once per control step and averaged over the attempt. *Joint error* is the mean absolute difference between reference and measured joint angles over the 29 actuated degrees of freedom, in radians. *Root position error* is the Euclidean distance between reference and measured root positions in the world frame, in meters, and *root orientation error* the geodesic angle between the two root quaternions, in radians.

Uncensored attempts. Every attempt plays its reference from the first frame to the last: terminations are recorded, but the environment keeps running rather than resetting. This avoids the bias of censored measurements, under which a policy that terminates early is scored almost entirely on the easy opening of every sequence.

Task success. *Task success* measures how well the expected behavior is executed. It combines a small set of criteria selected per task, derived from the reference motions: terminal root height and uprightness for stair climbing; the same plus terminal root location for stair descent and standing up; terminal root location and height for sitting; terminal root height, uprightness, and peak height for stepping onto a box; terminal root location and uprightness for slalom walking; and terminal object position for the box-manipulation tasks.

Evaluation setup. Each policy is rolled with 4096 environments for the teacher and SONIC and 384 for the students, which are limited by the memory requirements of the simulated depth camera. Students are evaluated on clean rendered depth with no augmentation.

F Task specifications

F.1 Per-task specifications

Every task is specified through the three components of Sec. 3.1: a prompt y for the frozen Kimodo prior [32], sparse spatiotemporal constraints $\mathcal{C}$ on a few keypoints (typically root/pelvis and hands, plus a heading direction), and a scene $\mathcal{S}$ of SDFs and terrain heightfields. It needs no additional data collection. The feet are never given targets: foot placement comes entirely from the contact and edge losses against the scene. The chair, object, and slalom tasks sample an initial root pose and an obstacle-avoiding path, which is their main source of behavioral diversity; the stair and step tasks prescribe the root trajectory from the geometry instead, varying the terrain and the sampled noise. Per-task loss weights are in Appendix G.2.

- **Climbing Stairs:** prompt “A person climbs up stairs.” The root follows a stepped path up the staircase while the foot-contact and edge losses place the feet on the treads. *Scene:* a closed-form staircase SDF over a ground plane, with a heightfield obtained from the same geometry, which is used to compute the edges for $\mathcal{L}_{\text{edge}}$. *Randomization:* tread (step depth) $\in \{0.3, 0.4\}$ m $\times$ rise (step height) $\in \{0.1, 0.2\}$ m over five steps.
- **Descending Stairs:** prompt “A person climbs down stairs.” *Scene* and *randomization* as for Climbing Stairs.
- **Step Up/Down:** prompts “A person climbs up a box.” then “A person climbs down a box.”, optimized as two overlapping windows. The root follows a prescribed trajectory: a walk in, a rise onto the box, a pause on top, and a drop. *Scene:* a closed-form box SDF over a ground plane. *Randomization:* box heights $\{0.3, 0.4, 0.5\}$ m against depths $\{0.3, 0.6\}$ m.
- **Sit on a Chair:** prompt “A person walks for sometime and sits down on a chair.” The root follows an approach path and is then held at the seat, with the robot turned to face away from the backrest, as it must be to sit. *Scene:* a parametric chair SDF consisting of 2 cuboids (seat and backrest) on a ground plane. *Randomization:* seat height $\in \{0.2, 0.3, 0.4, 0.5\}$ m, with start positions drawn within a 2.0 m radius.
- **Standing Up:** prompt “A person sitting on a chair stands up and walks forward.”, which starts the motion seated without any explicit initialization. The root is held at the seat, with its heading held in the same seated orientation as Sit on a Chair; it is then constrained along a prescribed path 0.5 m out from the chair’s front edge. There is no explicit stand-up constraint, the rise results from the prompt. *Scene:* the same chair SDF as Sit on a Chair. *Randomization:* seat height $\in \{0.2, 0.3, 0.4, 0.5\}$ m.
- **Slalom Walking:** prompt “A person walks forward, turning left and right to avoid obstacles.” A path weaving between the pillars is prescribed and paced at 1.0 m/s; the root tracks it and the pelvis height is held at its nominal standing value. *Scene:* a union of SDFs of pillars which act as obstacles over a ground plane. *Randomization:* pillar shape $\in \{\text{box}, \text{cone}\} \times \text{count} \in \{3, 4, 5\} \times \text{spacing} \in \{0.9, 1.2, 1.5\}$ m.
- **Pick and Place Box:** prompts “A person walks.”, “A person lifts a crate with both hands.”, “A person carrying a crate with both hands walks forward.”, and “A person places a crate on the floor with both hands.”, optimized as four overlapping windows. Both hand end effectors are constrained to the box’s side faces and the pelvis height profile is prescribed for the lift and the set-down. *Scene:* a ground plane and a moving box SDF driven by an object

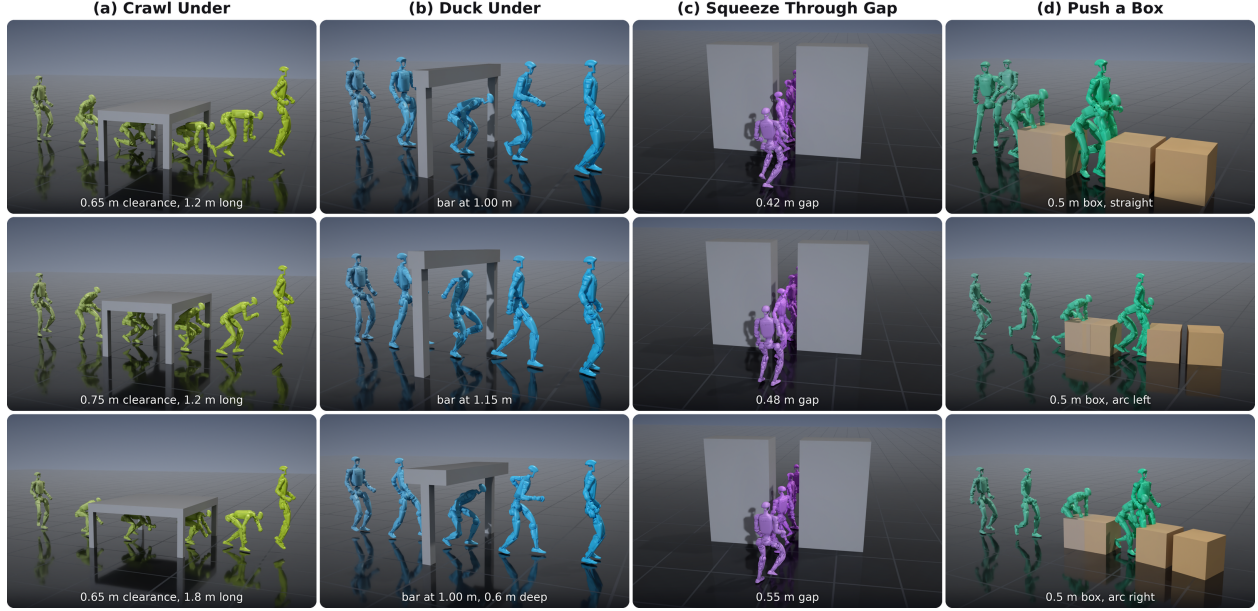


Figure 9. Agent-authored behaviors across their randomization settings. Each panel overlays frames from a single generated motion. The three rows are different scene configurations, and the columns are (a) Crawl Under, varying slab clearance and length; (b) Duck Under, varying slab clearance and depth; (c) Squeeze Through Gap, varying gap width; and (d) Push a Box, varying box path. The prompt and constraints behind every panel were written by the coding agent, with no task-specific demonstrations and no hand-authored constraints.

trajectory. *Randomization*: boxes with side lengths in $[0.15, 0.50]$ m; the start pose is drawn from a wedge-shaped region 0.55–1.0 m behind the box.

- **Pick and Place on Table**: the same four windows as **Pick and Place Box**, with the fourth prompt “A person puts a crate down with both hands.” Only the destination for the constraints differs: the hands settle at the table’s height and the pelvis height is constrained to remain near standing height rather than dropping into a squat. *Scene*: adds a static table cuboid. *Randomization*: boxes with side lengths in $[0.15, 0.50]$ m, with table height $\in \{0.2, 0.4, 0.6\}$ m and table width $\in \{0.3, 0.5\}$ m at a fixed 0.6 m length.

F.2 Agent-authored task specification

This section details the procedure of Sec. 4.3, by which a frontier coding agent (Claude Code running Claude Fable 5) authors the task programs consumed by HIGenNTO.

Task interface. The agent writes a single function per task. This function takes a few task-specific parameters as input and returns the prompt y , the constraints $\mathcal{C}$, and the scene $\mathcal{S}$ of Sec. 3.1, together with the object trajectory and the frames in which contact is required. Varying those parameters is what yields the diverse scene configurations and initial poses for each task. The agent is instructed not to modify the optimizer or the loss terms.

Procedure. Given a short natural-language brief and a pointer to the text annotations of the corpus the prior was trained on, the agent: (i) mines the annotations for the group describing the requested behavior and adopts the corpus’s own phrasing as the prompt y ; (ii) probes the frozen prior by running the candidate prompts with no scene and no constraints, then measures the resulting motion for an initial estimate of plausible scene sizes and constraints; (iii) drafts the first iteration of the function that builds the scene and constraints for the task; (iv) generates a batch of motions ($2 \text{ settings} \times 5 \text{ motions} \times 100 \text{ optimization steps}$), reports per-term losses, renders the motions as videos to check that they follow the intended behavior and look natural, and iterates if needed. Steps (ii) and (iv) close the loop: the agent verifies that a task works before involving a human.

Agent-authored behaviors. An agent can author tasks in two ways. In the *user-specified* mode the human names the behavior (Pick and Place on Table, Slalom Walking, Step Up/Down). In the *discovery* mode (Crawl Under,

Duck Under, Squeeze Through Gap, Push a Box) the human names no task at all: given only the corpus annotations, the agent surveys the prior’s vocabulary and returns grounded task proposals, each citing the annotation group that makes it in-distribution, sketching an implementation, and proposing axes along which to randomize the motions. We use the annotations from the BONES-SEED dataset [5], used in Kimodo [32].

Figure 9 shows generated motions for each of these discovered tasks, with three scene configurations per task. All four use the same interface as the tasks of Appendix F.1, and their optimizer settings and loss weights are listed alongside them in Table 3.

- **Crawl Under:** prompts “A person walks forward and crouches down.”, “A person crawls forward on hands and knees.”, and “A person stands up and walks forward.”, generated over three windows. The root follows a prescribed path whose height descends from standing to a hands-and-knees pelvis height (this height is found by probing the prior as described in step (ii)) before the slab’s near face, is held through the slab span, and rises only once the far face is cleared. Each window boundary therefore falls where the body is momentarily at rest. *Scene:* a horizontal slab raised on four corner posts over a ground plane (5 cuboid SDFs), where the *clearance* is the free gap between the ground and the slab’s underside. *Randomization:* clearance $\in \{0.65, 0.75\}$ m and slab length $\in \{1.2, 1.8\}$ m, with start positions randomized up to 0.25 m back along the approach path.
- **Duck Under:** prompt “A person walks forward and bends down to avoid an obstacle.” The root follows a straight path at a constant speed, with a pelvis-height constraint that dips across the slab span and returns to standing once past it; the SDF-based loss against the slab forces the head to clear it. *Scene:* an overhead slab spanning the path on two side posts over a ground plane (3 cuboid SDFs), where the *clearance* is the gap between the ground and the slab’s underside. *Randomization:* clearance $\in \{1.00, 1.15\}$ m and slab depth $\in \{0.20, 0.60\}$ m, with start positions randomized up to 0.30 m back along the approach path.
- **Squeeze Through Gap:** prompt “A person turns sideways and walks through a narrow gap.” The root path runs straight through the gap center, while a $\pm\pi/2$ heading constraint on the frames at the narrowest point turns the body sideways for the squeeze. *Scene:* two facing wall blocks (2 cuboid SDFs, 2.0 m tall) on a ground plane, leaving a corridor of *gap width* across the path. *Randomization:* gap width $\in \{0.42, 0.48, 0.55\}$ m, the leading shoulder sampled left or right per motion sequence, and start positions randomized up to 0.25 m back along the approach path.
- **Push a Box:** prompts “A person walks.”, “A person strains while slowly moving forward, pushing a large box.”, and “A person stands still.”, generated over three windows. The path of the box is prescribed first as a constant-curvature arc, and both hand targets are derived from it as rigid offsets on the box’s rear face. The root is constrained to trail the rear face at a fixed distance, with its heading held to the path tangent and its pelvis lowered through the pushing phase; it steps back once the box is released. *Scene:* a ground plane and a rigid box represented as a moving box SDF following the prescribed object trajectory. *Randomization:* box edge $\in \{0.5, 0.6\}$ m $\times$ path $\in \{\text{straight, left arc, right arc}\}$ with curvature $\pm 0.15 \text{ m}^{-1}$, pushed 1.75–2.0 m, with start positions drawn from a wedge-shaped region 0.9–1.8 m behind the box.

G Constraint-guided motion optimization

G.1 Optimization algorithm

G.2 Loss terms

We detail the loss terms summed in line 5 of Alg. 1. Let $\mathbf{x}_j^t \in \mathbb{R}^3$ denote the world-frame position of joint j at frame t for a single sample. The scene SDF $s(\cdot)$ (negative inside geometry), the contact threshold τ , and the skin distances δ_j and δ_j^{mov} are as defined in Appendix C.2, as is the set of sole points $\mathcal{P}$; $\mathcal{F} = \{\text{left, right}\}$ indexes the two feet.

Goal ($\mathcal{L}_{\text{goal}}$). A mask-normalized Huber loss between the denoised joints and the targets $\hat{\mathbf{x}}_j^t$ obtained from $\mathcal{C}$, gated by a per-frame mask $m_j^t \in \{0, 1\}$ that selects only the constrained (t, j) entries:

$$\mathcal{L}_{\text{goal}} = \frac{\sum_{t,j} m_j^t H_1(\mathbf{x}_j^t, \hat{\mathbf{x}}_j^t)}{\sum_{t,j} m_j^t}, \quad (6)$$

Algorithm 1 Constraint-guided motion optimization.

Require: prompt y , constraints $\mathcal{C}$, scene $\mathcal{S}$, outer steps K , DDIM steps N , batch size B

- 1: Initialize $\mathbf{z} \sim \mathcal{N}(0, \sigma_{\text{init}}^2 \mathbf{I})$ for B samples
 - 2: **for** $k = 1, \dots, K$ **do**
 - 3: $\mathbf{x} \leftarrow M_\theta(\mathbf{z} \mid y, \mathcal{C})$ $\triangleright$ full N -step DDIM denoise with gradients enabled
 - 4: Align $\mathbf{x}$ to the global coordinate frame at the desired frame-0 root position and orientation
 - 5: $\mathcal{L} \leftarrow w_g \mathcal{L}_{\text{goal}}(\mathbf{x}, \mathcal{C}) + w_c \mathcal{L}_{\text{coll}}(\mathbf{x}, \mathcal{S}) + w_f \mathcal{L}_{\text{foot}}(\mathbf{x}, \mathcal{S}) + w_h \mathcal{L}_{\text{hand}}(\mathbf{x}, \mathcal{S}) + w_e \mathcal{L}_{\text{edge}}(\mathbf{x}, \mathcal{S})$
 - 6: $\mathbf{g} \leftarrow \nabla_{\mathbf{z}} \mathcal{L}$
 - 7: $\mathbf{z} \leftarrow \text{ADAM}(\mathbf{z}, \mathbf{g})$; record per-sample best $\mathbf{x}_b^*$ at the lowest $\mathcal{L}_b$ seen so far
 - 8: **end for**
 - 9: **return** $\{\mathbf{x}_b^*\}_{b=1}^B$
-

where H_1 is the Huber loss ($\delta_H = 1$) summed over the three coordinates. This single term handles the constraints on the root, hand, and full-body; the mask is the only thing that changes across tasks.

Foot contact ($\mathcal{L}_{\text{foot}}$). Penalizes the foot nearest to the ground from floating, encouraging at least one foot to stay in contact at all times. It is queried at points on the sole of the feet $\mathcal{P}$.

$$\mathcal{L}_{\text{foot}} = \frac{1}{T} \sum_{t=1}^T \max\left(0, \min_{p \in \mathcal{P}} s(\mathbf{x}_p^t) - \tau\right). \quad (7)$$

Hand contact ($\mathcal{L}_{\text{hand}}$). Holds the hands at a fixed palm clearance d_{palm} off the object’s faces, measured against the object’s moving SDF over the grip window $\mathcal{G}$ and the hand keypoints $\mathcal{H}$. Unlike the terms above it is two-sided (a hand drifting off the object is penalized as much as one sinking into it) and tolerates a deviation of up to τ from the target clearance:

$$\mathcal{L}_{\text{hand}} = \frac{1}{|\mathcal{G}| |\mathcal{H}|} \sum_{t \in \mathcal{G}} \sum_{j \in \mathcal{H}} \max\left(0, |s_o(\mathbf{x}_j^t) - d_{\text{palm}}| - \tau\right). \quad (8)$$

It is active on the tasks involving object manipulation.

Collision ($\mathcal{L}_{\text{coll}}$). Penalizes penetration of any joint into scene geometry up to its skin distance:

$$\mathcal{L}_{\text{coll}} = \frac{1}{T} \sum_{t=1}^T \sum_{j=1}^J \max\left(0, \delta_j - s(\mathbf{x}_j^t)\right). \quad (9)$$

Each moving obstacle contributes an additional term of the same form, queried against its per-frame (moving) SDF.

Edge contact ($\mathcal{L}_{\text{edge}}$). This term keeps the planted foot away from sharp terrain edges. Each foot f carries a query set $\mathcal{Q}_f$ of its ankle and toe points, and we define a per-point risk as a linear ramp in the distance $d(q)$ to the nearest edge,

$$\ell(q) = 1 - \text{clamp}(d(q)/\rho, 0, 1) \in [0, 1], \quad (10)$$

where ρ is a safety radius and $d(q)$ is the distance from an edge precomputed from the terrain heightfield. The planted foot f^* is the one whose lowest query point sits closest to the terrain,

$$f^* = \arg \min_{f \in \mathcal{F}} \min_{q \in \mathcal{Q}_f} (z(q^t) - h(q^t)), \quad (11)$$

with $z(\cdot)$ the height of a query point and $h(\cdot)$ the terrain height beneath it. The term is the planted foot’s mean point risk, averaged over the sequence,

$$\mathcal{L}_{\text{edge}} = \frac{1}{T} \sum_{t=1}^T \frac{1}{|\mathcal{Q}_{f^*}|} \sum_{q \in \mathcal{Q}_{f^*}} \ell(q^t). \quad (12)$$

The choice of f^* is treated as a constant in the backward pass, so gradients flow through the selected foot’s risk but not through which foot is selected.

Task	K	η	w_c	w_f	w_e	w_h
Chair (sit, stand)	50	0.01	0.1	0.1	0	0
Stairs (up, down)	50	0.05	1.0	1.5	1.0	0
Step up/down	50	0.05	2.0	1.5	1.0	0
Slalom	100	0.05	2.0	1.5	0	0
Pick and place	100	0.01	0.5	0.1	0	1.0
<i>Agent-authored (Appendix F.2)</i>						
Crawl under	100	0.05	2.0	1.5	0	0
Duck under	100	0.05	2.0	1.5	0	0
Squeeze through gap	100	0.05	2.0	1.5	0	0
Push a box	100	0.05	2.0	1.5	0	1.0

Table 3. Optimizer settings and loss weights per task family, for the eight evaluated tasks and the four agent-authored behaviors. K is the number of outer Adam iterations of Alg. 1 and η its learning rate; $w_g=1.0$ throughout. The two pick tasks share one recipe, as do the two chair tasks and the two stair tasks. The agent-authored tasks were all generated with a single recipe, differing only in whether the task carries a grip.

Loss weights. The goal weight is $w_g=1.0$ throughout; the remaining weights and the optimizer settings are per task, and are listed in Table 3. The shared constants are the contact threshold $\tau=0.01$ m; the skin distance $\delta_j=0.05$ m (0.01 m on the ankles and toes) and $\delta_j^{\text{mov}}=0.05$ m against moving obstacles (0.03 m on the legs and toes); the safety radius $\rho=0.10$ m; and Huber $\delta_H=1$. The latent is initialized at $\sigma_{\text{init}}=0.1$ (Alg. 1), which we observe to produce more natural-looking results when using noise optimization; the baselines use $\sigma=1$. The edge term is enabled only where the scene carries a heightfield, and the hand-contact term only where the task carries a grip.

H Policy learning

H.1 Simulation setup and dynamics randomization

Simulator and robot. Teachers and students are trained in Isaac Sim through the HoLosoma training stack [2]. Physics is stepped at 200 Hz with a control decimation of four, giving the 50 Hz control rate. The robot is a Unitree G1 with 29 actuated degrees of freedom over 32 bodies, driven by joint-space PD control with per-joint gains and the manufacturer’s torque limits; the policy output is a residual on a fixed default pose.

Episodes and termination. Teachers are trained with 4096 parallel environments and students with 384. Episodes last 10–15 s, drawing a reference motion sequence uniformly from the task’s batch at each reset. An episode terminates when the root drifts more than 0.5 m from the reference, when the projected gravity of the root differs from the reference by more than 0.8, when any of four bodies (the two ankles and the two wrists) exceeds 0.25 m of tracking error, or, on object tasks, when the object’s pose relative to the robot deviates by more than 0.25 m or 0.8 rad.

Dynamics randomization. The following are resampled per environment: robot friction (0.3–1.6 static, 0.3–1.2 dynamic) and restitution (0.0–0.5); object friction (0.1–0.6), restitution (0.0–1.0), mass (1–4 $\times$) and principal inertia (0.5–1.5 $\times$); base center-of-mass offset (± 0.025 m longitudinally, ± 0.05 m laterally and vertically); and a ± 0.01 rad joint-position bias. Random pushes are applied every 1–3 s at up to 0.5 m/s linear and 0.78 rad/s angular velocity. Observations carry Gaussian noise on base angular velocity (0.2), joint position (0.01), joint velocity (0.5), and reference orientation (0.05). The teacher and student stages randomize identically; the student adds the depth randomization (Appendix H.4).

H.2 Motion-tracking teacher

A single teacher (Sec. 3.3) is trained per task with PPO [33] on the full batch of motions generated for that task: at each episode reset, a reference trajectory is sampled uniformly from the batch and the policy is required to track it. The actor consumes the robot’s proprioception, the kinematic reference over a short horizon of future steps, and a privileged terrain height scan; on object tasks it additionally receives the object’s pose, velocity, and sampled surface points in the robot frame. It outputs a 29-D joint-angle residual over the default pose, applied through PD control. The critic is further privileged with reference-body and tracked-body poses and the base linear velocity. Rewards combine standard motion-tracking terms (global and per-body pose, position, and velocity tracking, plus object-pose tracking on

object tasks) with regularizers on action rate, DoF limits, and undesired contacts. The actor-critic architecture and PPO hyperparameters are the default values from [2].

H.3 Depth-conditioned flow student

Objective. Along the straight-line path $\mathbf{a}_t = (1 - t) \mathbf{a}^{\text{tea}} + t \epsilon$ that connects the teacher action $\mathbf{a}^{\text{tea}}$ at $t=0$ to Gaussian noise $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ at $t=1$, the student (Sec. 3.3) regresses a conditional velocity field v_θ onto the velocity of that path, given the onboard sensor context $\mathbf{c}$:

$$\mathcal{L}_{\text{student}} = \mathbb{E}_{t, \epsilon} \| v_\theta(\mathbf{a}_t, t, \mathbf{c}) - (\epsilon - \mathbf{a}^{\text{tea}}) \|^2. \quad (13)$$

At test time an action is obtained by integrating v_θ from $t=1$ to $t=0$.

Observations. The student (Sec. 3.3) observes only quantities the robot carries: a short history of head-mounted depth frames, each encoded by a shared depth CNN into a compact embedding, and a proprioception stream (base angular velocity, projected gravity, joint-angle deviations from the default pose, joint velocities, end-effector positions in the base frame, and the previous action) kept as both a short recent window and a longer buffer compressed by a temporal 1-D CNN. Shorter-horizon tasks use a smaller memory. It outputs the same 29-D joint-angle residual as the teacher.

Architecture. The velocity field v_θ of Eq. 13 is a small transformer that attends over tokens for the proprioception history, the depth frames, the current action $\mathbf{a}_t$, and the flow time t (alongside learnable register tokens), and projects the action token’s output back to the 29-D velocity.

Training. Distillation is on-policy behavior cloning: the student rolls out in simulation and the frozen teacher relabels every visited state with its action, the regression target of Eq. 13; the teacher itself never acts in the environment ($\beta=0$ throughout). At evaluation and deployment the field is integrated with 5 fixed Euler steps. A value critic is trained alongside by standard value regression [33] but contributes no gradient to the student. Training uses AdamW (learning rate 5×10^{-4}) for 25k iterations across 384 parallel environments; depth randomization is applied throughout.

H.4 Depth randomization for sim-to-real

Rendered depth is clean, whereas the on-board camera returns holes, invalid regions, self-occlusion from the robot’s body, and slight mount misalignment. To close this gap we add depth randomization during distillation, following [51]. It acts on the preprocessed frame stack of Appendix H.3 (normalized so that +0.5 is far or missing and −0.5 near) per minibatch, in two stages.

Camera-rotation jitter. Each sample draws one in-plane rotation $\alpha \sim \mathcal{U}(-5^\circ, 5^\circ)$, applied to every frame of that sample’s stack, modeling a fixed but uncertain mount orientation rather than per-frame shake.

Random masking. Two families of occlusion are added. Fixed blocks anchored to the bottom corners of the frame model self-occlusion by the robot’s own body; they are drawn per sample and held fixed across the stack, since the body stays in view over the whole window. Random rectangles model transient holes and dropouts; these are drawn independently per frame. Randomization is applied only during training and not during deployment.