Real-to-Sim-to-Real at the Last Millimeter

Tianchong Jiang[1,*] **Ruize Xu**[2,*] **Yicheng Li**[2] **Xiangshan Tan**[1] **Matthew Walter**[1]
[1]Toyota Technological Institute at Chicago, Chicago, IL, USA
[2]University of Chicago, Chicago, IL, USA
{tianchongj,vincenttann,mwalter}@ttic.edu, {richard1xur,yicheng1}@uchicago.edu

Abstract: We study drawer and charger insertion with 0.7–2 mm clearances on a roughly $1,000 arm through real-to-sim-to-real learning. Coding agents write scripted teachers that read the simulator state, and image-based student policies imitate them. Although our simulator cannot match the arm to within the clearance, the students succeed in 12/20 drawer and 18/20 charger trials on the real arm. We report three lessons. First, scripted teachers overfit the simulator less than RL teachers, which learn behaviors that work only in simulation. Second, without human guidance, coding agents can edit scripted teachers to be much easier for students to learn. Third, without force sensors, the student senses contact from commanded and measured joint positions, so it need not line the part up perfectly before contact. Video: https://ripl.github.io/last-millimeter.

Keywords: Real-to-sim-to-real, Precise insertion, Teacher-student distillation

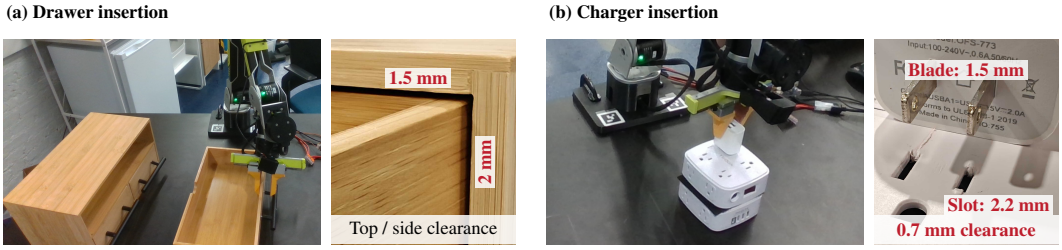


Figure 1: Real drawer (a) and charger (b) insertion on a roughly $1,000 arm. Close-ups on the right show the measured clearances.

Introduction. At millimeter clearances, slight misalignment can jam an insertion. Prior work has shown such insertions on a Franka FR3 [1] and on roughly $5,000 ALOHA 2 arms [2]. Despite substantial system identification, our simulator cannot match our roughly $1,000 arm to within the clearance. The arm usually repeats a motion within 1 mm, but friction stops its joints up to 7° short of their targets. Our best rigid model still misplaces the wrist camera by a median 10 mm, suggesting the arm is not rigid (Fig. 2b). Even within simulation, replaying the same action sequence on GPU instead of CPU, or with a smaller timestep, shifts the final drawer position by 1–2 cm (Fig. 2a). Policies must therefore tolerate these errors. We report three lessons from training such policies with scripted teachers in simulation.

Tasks and pipeline. A 6-DoF Seeed reBot B601-DM picks up a drawer and inserts it into a cabinet, or transfers a phone charger between adjacent outlets (Fig. 1). The Isaac Sim drawer model is fitted to ten teleoperated demonstrations. The charger model is built from one, with geometry measured in its images.

In simulation, a teacher sees the simulator state and labels actions for a student. Prior work usually trains the teacher with reinforcement learning (RL) [1, 3, 4, 5, 6]. Our teachers are instead Python

[*]Equal contribution.

Submitted to the Late-Breaking Results track of the 10th Conference on Robot Learning (CoRL 2026).

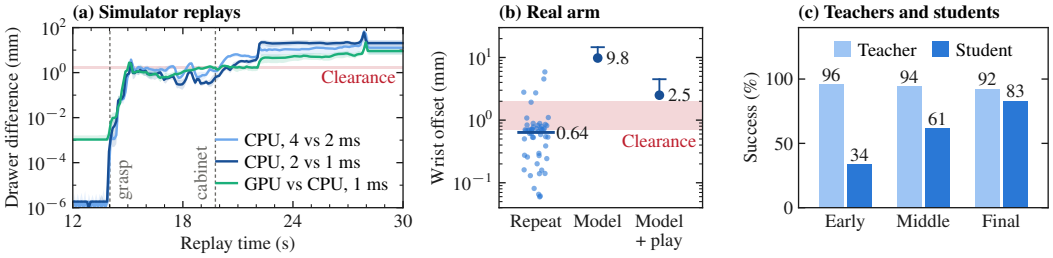


Figure 2: (a) Drawer-position differences between replays of the same action sequence at different timesteps or on GPU versus CPU (median and interquartile range, 64 paired starts). (b) Real wrist offsets across repeats, and model error without and with joint play (median, maximum). (c) Early, middle and final drawer teachers and their students. Panels (a) and (c) show simulation results.

programs written and revised by coding agents [7]. The student is a fine-tuned DINOv2 [8] with an MLP head. It sees scene and wrist images, plus commanded and measured joint positions, and outputs action chunks [9]. We train it with online DAgger [10], in which the teacher labels the states the student visits. Training randomizes masses, frictions, grip force, lighting, colors, backgrounds and camera poses.

1. Scripted teachers overfit the simulator less than RL teachers. The real demonstrations used to fit a simulator do not reveal subtle contact details [11]. Even under randomized physics, RL teachers can exploit the simulator's errors in these details and learn brittle contact strategies that work only in simulation. For example, one RL teacher grasped the drawer with only the fingertips of the gripper. In simulation this grasp is stable, but on the real arm it is not, making insertion nearly impossible. A scripted teacher does only what its code states, so it has less room to exploit these errors. Correcting this grasp through RL required task-specific reward shaping and retraining. A coding agent fixes a script in minutes. We spent roughly two weeks developing RL teachers and two days developing scripted teachers, with coding agents assisting both. On the charger, the RL student succeeded as often as the scripted one in simulation (85.1% vs. 83.9%) but never on the real arm (Table 1).

Teacher	Drawer	Charger
Scripted	12/20	18/20
RL	6/20	0/20

Table 1: Real-world success of students distilled from scripted and RL teachers.

2. Successful scripts must also be learnable. Image-based students often find planners and scripts hard to imitate [12, 13, 14], but we find a coding agent can make scripts much easier to learn without human guidance. Our teachers read only the objects' true poses and, for the drawer, finger contact, all of which the student can estimate. Even so, our early drawer teacher succeeded over 90% in simulation, but its student only 34% (Fig. 2c). The teacher's gripper command flipped at a sharp pose threshold, so nearly identical views got opposite commands. The agent proposed and implemented smoothing such rules, raising student success to 61% while the teacher stayed above 90%. A single-action student also stalls where the teacher pauses, as the unchanged state keeps producing the same command [15]. One such student succeeded 90% in simulation but stalled on the real arm, so we adopted action chunking.

3. Use contact to resolve alignment uncertainty. Our arm has no joint torque, wrist force/torque, or tactile sensors. We therefore use the difference between commanded joint positions q_{des} and measured joint positions q under joint proportional–derivative (PD) control as an indirect cue to contact resistance [16]. We provide both signals to the student alongside images so it can use contact to guide insertion. Like a human, both students press the part against the mating surface and slide it until it aligns with the opening and enters. The part therefore need not be aligned to within the insertion clearance before contact. In a stuck real insertion, the drawer student stops pushing when the gap between commanded and measured positions is large, and pushes again when we shrink only this gap in its input, showing that it uses the gap to sense when the drawer is blocked.

References

[1] T. Sun, B. Han, P. Yin, R. Xu, H. He, A. Gupta, S. Rusinkiewicz, and Y. Shao. Visual sim-to-real learning for robotic insertion under geometric variations: Application to rebar installation. *arXiv preprint arXiv:2609.20477*, 2026.

[2] T. Z. Zhao, J. Tompson, D. Driess, P. Florence, S. K. S. Ghasemipour, C. Finn, and A. Wahid. ALOHA Unleashed: A simple recipe for robot dexterity. In *Conference on Robot Learning (CoRL)*, 2024.

[3] P. Yin, T. Westenbroek, Z. Zhang, J. Tran, I. Dagnino, E. Shilamkar, N. Mbiziwo-Tiapo, S. Bagaria, X. Liu, G. Mullins, A. Kolobov, and A. Gupta. Emergent dexterity via diverse resets and large-scale reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2026.

[4] T. He, Z. Wang, H. Xue, Q. Ben, Z. Luo, W. Xiao, Y. Yuan, X. Da, F. Castañeda, S. Sastry, C. Liu, G. Shi, L. Fan, and Y. Zhu. VIRAL: Visual sim-to-real at scale for humanoid loco-manipulation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

[5] X. Cheng, K. Shi, A. Agarwal, and D. Pathak. Extreme parkour with legged robots. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[6] Z. Zhuang, Z. Fu, J. Wang, C. Atkeson, S. Schwertfeger, C. Finn, and H. Zhao. Robot parkour learning. In *Conference on Robot Learning (CoRL)*, 2023.

[7] J. Liang, W. Huang, F. Xia, P. Xu, K. Hausman, B. Ichter, P. Florence, and A. Zeng. Code as Policies: Language model programs for embodied control. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[8] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jégou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

[9] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems (RSS)*, 2023.

[10] S. Ross, G. J. Gordon, and J. A. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011.

[11] N. Fazeli, R. Tedrake, and A. Rodriguez. Identifiability analysis of planar rigid-body frictional contact. In *International Symposium on Robotics Research (ISRR)*, 2017.

[12] M. J. McDonald and D. Hadfield-Menell. Guided imitation of task and motion planning. In *Conference on Robot Learning (CoRL)*, 2021.

[13] M. Dalal, A. Mandlekar, C. Garrett, A. Handa, R. Salakhutdinov, and D. Fox. Imitating task and motion planning with visuomotor transformers. In *Conference on Robot Learning (CoRL)*, 2023.

[14] Z. Xu, R. Gong, M. V. Minniti, K. Sivakumar, A. S. Gundogdu, E. Rosen, R. Yan, T. Kusnur, Z. Wang, D. Deng, P. Stone, X. Zhang, and K. Schmeckpeper. ExpertGen: Scalable sim-to-real expert policy learning from imperfect behavior priors. *arXiv preprint arXiv:2603.15956*, 2026.

[15] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song. Diffusion Policy: Visuomotor policy learning via action diffusion. In *Robotics: Science and Systems (RSS)*, 2023.

[16] A. Bronars, Y. Park, K. Kedia, and P. Agrawal. Blind precise assembly: Proprioceptive sim-to-real peg-in-hole for diverse parts. *arXiv preprint arXiv:2610.23849*, 2026.