Catch It All!: Generalizable Dynamic Catching with a Dexterous Hand

Author Names Omitted for Anonymous Review.

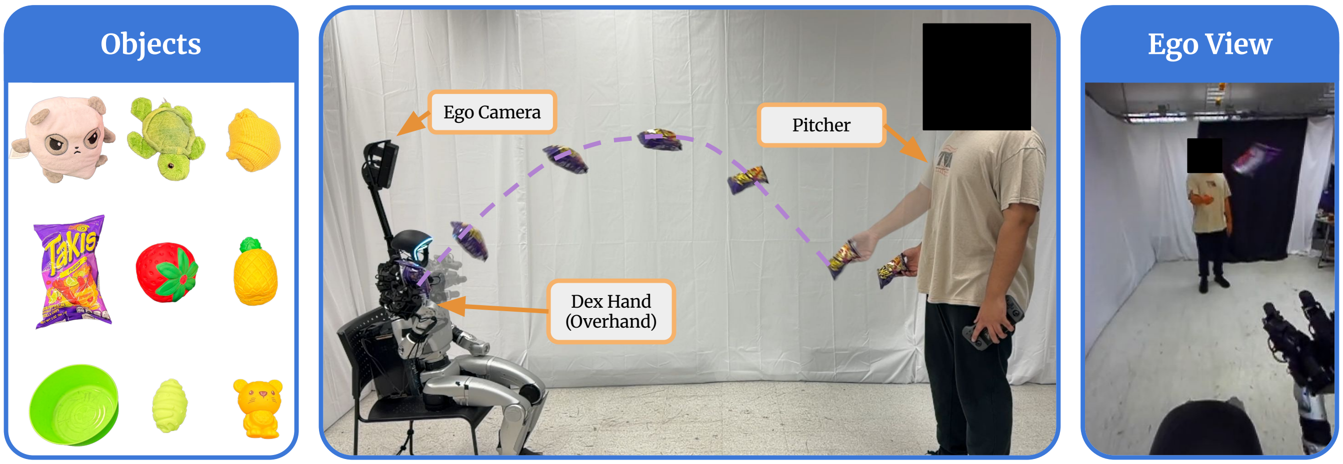

Fig. 1. **Overview.** Our method successfully catches diverse, unseen objects overhand and underhand using only an egocentric camera. Left: The real-world object test set includes 9 unseen objects with variable size, compliance, and shape. Middle: Catching setup with human tossing an object selected at random from the test set. Right: Egocentric view; this is the only camera in our setup.

***Abstract*—Catching arbitrary objects underhand and overhand requires robots to adapt in real-time to changing visual observations, unseen object properties, and diverse contact dynamics. This paper presents *Catch It All!*, a sim-to-real catching framework for humanoid robots that generalizes to novel objects with variable physical properties. Through careful reward and curriculum design, we learn robust grasps capable of effectively capturing objects both underhand and overhand. Meanwhile, our safety-aware curricula facilitate effective exploration near actuator limits to enable safe, high-frequency action execution. *Catch It All!* is deployed zero-shot on a Unitree G1 humanoid without any fine-tuning, successfully catching 9 real objects unseen during training with varying deformability, geometry, mass, volume, and restitution. We will release all real-world videos, code, and the webpage after anonymous review is complete.**

I. Introduction

Robotic catching is a dynamic interception problem that requires low-latency, temporally precise coordination between perception and actuation [4, 9]. Prior systems demonstrate high-speed interception on quadrupeds and industrial manipulators [3, 15, 4, 6], but remain limited in their ability to generalize to different objects or underhand/overhand tosses. We present ***Catch It All!***, a generalizable dexterous catching framework for underhand and overhand tosses that operates with egocentric visual sensing without motion capture or human demonstrations. Our contributions are: **(1)** generalizing to novel objects in both underhand and overhand tosses through grasp reward design and a toss curriculum, **(2)** safe action execution near hardware limits via torque clipping and reward penalty curricula, and **(3)** zero-shot sim-to-real transfer, capable of catching objects unseen in simulation in both underhand and overhand tosses.

II. Related Work

Prior approaches to dynamic catching often reduce task difficulty by simplifying physical interaction. Some works utilize simplified end effectors such as nets [3]. Huang et al. [6] tackle a multi-agent handover variant of dexterous catching, while Zhang et al. [19] utilize mobile manipulators to expand the workspace. However, existing dexterous catching work are limited in their ability to generalize to different object types due to simplified objects (e.g. sandbags, low rebound, simple shape) or limited dexterity and grasping ability. Generalizable catching remains an open problem, requiring robust grasps for objects with varying shapes and physical properties.

Moreover, extending these capabilities to humanoid platforms introduces additional challenges: compared to industrial manipulators, humanoid actuators typically exhibit lower torque limits and control bandwidth [7, 14]. Enabling dynamic dexterous manipulation on such platforms requires safe action execution at hardware limits.

III. Method

A. System Setup

Compared to industrial systems, humanoids often have lower actuator bandwidth and coarser control resolution. Thus, learning high-frequency, reactive control for arms while operating near hardware limits remains a roadblock for performing dynamic dexterous manipulation tasks such as catching with humanoids. So, we choose our hardware setup to tackle sim-to-real transfer for dexterous catching with a humanoid arm.

We deploy on a seated Unitree G1 with a mounted ZED2i for egocentric observations. The mounted 20-DoF House of

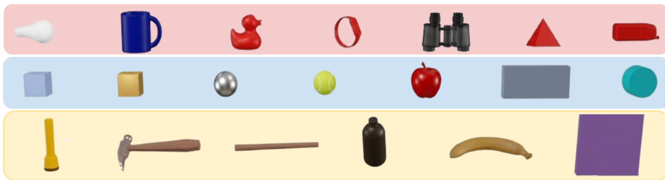

Fig. 2. **Simulation objects.** Top to bottom: irregular group, primitive objects, and long axis objects. Only primitive objects are used for training.

Dextra hand [2] is accessible due to repairability and low cost, while remaining smaller and lighter than other open source options like the LEAP hand [17]. The palm is printed with carbon fiber to remain safely within the G1's payload capacity.

B. *Policy Training*

We formulate our catching task as a Partially-Observable Markov Decision Process (POMDP) which can be described as tuple $\mathcal{M} = (\mathcal{S}, \mathcal{O}, \mathcal{A}, \mathcal{R}, \mathcal{T}, \rho, \gamma)$ with continuous state space $\mathcal{S}$ partially observed by observation space $\mathcal{O}$, action space $\mathcal{A}$, scalar reward function $\mathcal{R} : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$, environment dynamics function $\mathcal{T} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$, initial state distribution ρ, and discount factor $\gamma \in [0, 1]$. The policy receives proprioception, 3D object position extracted with SAM2 [12], and 3D bounding box dimensions for the object. The arm is controlled with full delta 6D pose, and the hand with joint control.

We train separate underhand and overhand policies. As seen in Sec. IV-A2, due to noisy stereo depth readings and dropped position prediction when objects leave the frame, exploration is challenging for online RL. So, we adopt a teacher-student framework: teacher policies are trained with PPO [16] with ground-truth object position observations, then distilled to student policies with noisy observations using DAgger [13].

To increase simulation throughput, we train on primitive shapes with simple meshes, depicted in Fig. 2. We randomize object friction, mass, scale, and restitution. Tosses are initialized at $0.5\,\mathrm{m}$ to $2.75\,\mathrm{m}$ forward, $\pm 1.0\,\mathrm{m}$ lateral, and $0.25\,\mathrm{m}$ to $2.0\,\mathrm{m}$ height, sampled at 40% short, 40% medium, and 20% long range. We also randomize observation delay, control delay, toss delay, initial 6D hand pose, gravity, PD gains, link masses and friction, and random force injections.

We use IsaacGym [11] for training and Mujoco [18] for sim-to-sim evaluations. Our codebase extends Humanoid-Verse [10]. Policies are trained and deployed on a single workstation with a Ryzen 5 5600X CPU and RTX 4090 GPU.

C. *Rewards*

In order to generalize well to diverse, unseen objects, we learn policies with robust grasps to successfully capture objects and prevent them from bouncing or slipping out of the hand. In Sec. IV-A4, we show a key reward term is our fingertip-object surface distance reward. Contact-based and stability rewards from prior work [19] struggle to generalize across our diverse evaluation objects, while our distance-based fingertip-surface reward is designed to provide denser signal as fingers approach the object, or when all fingers cannot touch the object simultaneously (e.g. small/irregular objects).

TABLE I
REWARD TERMS. RL TEACHER CATCHING POLICY.

Reward Term	**Equation**	**Scale**
	Task	
Approx Fingertip-Obj Surface Dist	$\exp(-10(\max_{\text{ftip}} \lVert p_* - p_{\text{obj}} \rVert - r_{\text{obj}}))$	4.0
Obj-Palm Dist	$\exp(-2\lVert p_{\text{obj}} - p_{\text{palm}} \rVert)$	4.0
Obj Height	$\text{clip}(p^z_{\text{obj}} - 0.2, 0.0, 1.3)/1.3$	1.0
	Posture / Contact	
DoF Default Pos	$\lVert q - q_{\text{default}} \rVert_1$	-1.0
Body Collision	$\lVert F_{\text{body}} \rVert_1$	-2.0
Arm Collision	$\lVert F_{\text{arm}} \rVert_1$	-1.0
	Joint Regularization	
DoF Track Error	$\lVert q - q_{\text{target}} \rVert^2$	-0.1
Action Rate	$\lVert a_t - a_{t-1} \rVert^2$	-0.01
DoF Smoothness	$\lVert a_t - 2a_{t-1} + a_{t-2} \rVert^2$	-0.05
DoF Power	$\lVert \lvert \dot{q} \rvert \odot \lvert \tau \rvert \rVert_1$	$-2e{-5}$
DoF Vel	$\lVert \dot{q} \rVert^2$	$-1e{-4}$
DoF Acc	$\lVert \ddot{q} \rVert^2$	$-2.5e{-7}$
Torques	$\lVert \tau \oslash K_p \rVert^2$	$-2.5e{-6}$
DoF Pos Lims	$\lVert (q_{\min} - q)_+ + (q - q_{\max})_+ \rVert_1$	-2.0
DoF Vel Lims	$\lVert [\lvert \dot{q} \rvert - 0.8\dot{q}_{\max}]_+ \rVert_1$	$-2e{-3}$
Torque Lims	$\lVert [\lvert \tau \rvert - 0.95\tau_{\max}]_+ \rVert_1$	-0.1
Action Vanish	$\lVert [a - a^{\max}]_+ + [a^{\min} - a]_+ \rVert_1$	-1.0
	Hand Regularization	
Hand Pos Rate	$\lVert p_t^{\text{hand}} - p_{t-1}^{\text{hand}} \rVert$	-0.05
Hand Smoothness	$\lVert p_t^{\text{hand}} - 2p_{t-1}^{\text{hand}} + p_{t-2}^{\text{hand}} \rVert$	-0.05
Hand Orientation	$\lVert g^{xy}_{\text{hand}} \rVert^2$	-0.025

We supplement our grasping reward with object-hand distance and object height task rewards to aid exploration. We additionally use common penalty rewards for humanoid tasks to regularize robot behavior (self-collisions; joint vel, acc, torque; joint pos, vel, and torque limits; default joint pos displacement; etc) [1, 5]. Equations for reward terms are available in Table I.

D. *Curricula*

Toss target curriculum. Learning overhand catching via random exploration is particularly difficult. To address this, we implement a toss target curriculum: initially, objects are thrown exactly $0.1\,\mathrm{m}$ above and $0.1\,\mathrm{m}$ behind the hand initial position, encouraging upright hand orientation when catching. As training progresses, the toss target range is expanded to fill the complete range of feasible overhand catches.

Torque clip curriculum. With the low torque limits of the Unitree G1 arm (as low as $5\,\mathrm{N{\cdot}m}$), naive torque clipping yields policies that output torques exceeding hardware limits. To address this, we propose a torque clipping curriculum: initially, torques are clipped at nominal torque limits to enable effective exploration. As training progresses, the torque clip limits increase to $2\times$ nominal torque limits, while torques beyond nominal limits are penalized in the reward, allowing the policy to explore effectively near actuator limits. Per Sec. IV-A2 and

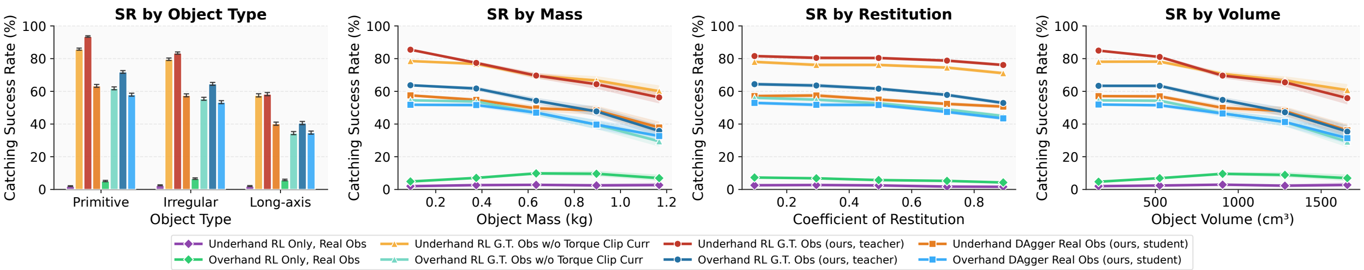

Fig. 3. **Robustness and generalization.** We evaluate how well our underhand and overhand policies generalize to new object geometries, as well as their robustness to mass, volume, and restitution randomizations. Primitive objects are seen during training, while irregular and long-axis objects are unseen during training. 95% CIs are shown via error bars and shaded regions. With 32,768 evaluation episodes, some CIs may be too small to see. Best viewed zoomed.

TABLE II
BASELINE SIM-TO-SIM SUCCESS RATE (%) WITH 95% CIS.

Method	Underhand	Overhand
RL Only, Real Obs	0.1 $\pm$ 0.4	0.9 $\pm$ 0.7
RL G.T. Obs w/o Torque Clip Curr	43.8 $\pm$ 2.7	8.8 $\pm$ 1.7
RL G.T. Obs (ours, teacher)	**81.6 $\pm$ 2.2**	22.1 $\pm$ 2.4
DAgger Real Obs (ours, student)	56.3 $\pm$ 2.7	**37.0 $\pm$ 2.7**

IV-A3, the torque clip curriculum improves performance while exhibiting lower joint mechanical power over time.

Penalty curriculum. Following [5, 10], we implement a penalty curriculum which scales joint and hand regularization penalties from 1$\times$ to 20$\times$. Combined with the torque clip curriculum, this yields progressively greater penalties for actions beyond nominal torque limits, ensuring safer behavior.

IV. EXPERIMENTS

A. Simulation and Sim-to-Sim Transfer Evaluations

1) Experimental Setup: We evaluate 32,768 episodes per policy in IsaacGym. Additionally, we report sim-to-sim transfer results in Mujoco with 1280 episodes per policy to test performance under different contact dynamics, friction, and collision response while providing larger-scale evaluations than is possible with real human-thrown tests.

For IsaacGym evaluations, we maintain all randomizations described in Sec. III-B. For sim-to-sim evaluations, we do not perform additional dynamics randomizations, however we do add observation noise and camera latency. Since the Unitree API does not limit torque output, torque clipping is disabled for all evaluations. See Fig. 2 for objects used in evaluation.

2) Performance and Generalization: **Does our method show strong generalization to unseen objects with variable physical properties?** To evaluate performance and generalization, we provide four baselines:

1) **RL Only, Real Obs**: RL policy trained directly on noisy/dropped object positions. As described in Sec. III-D, the torque clip and penalty reward curricula are enabled for underhand and overhand throws, and the toss target curriculum is enabled for overhand throws.
2) **RL G.T. Obs w/o Torque Clip Curr**: RL policy trained on ground-truth (G.T.) object positions, with the torque clip curriculum disabled. The remaining curricula are used as in **(1)**.
3) **RL G.T. Obs (ours, teacher)**: RL policy trained on ground-truth (G.T.) object positions. All curricula are enabled as in **(1)**. This policy is used as the teacher for the sim-to-real deployed student policy.
4) **DAgger Real Obs (ours, student)**: DAgger student policy distilled from teacher policy **(3)**. During student training, all curricula and torque clipping are disabled. This is the policy used for sim-to-real deployment.

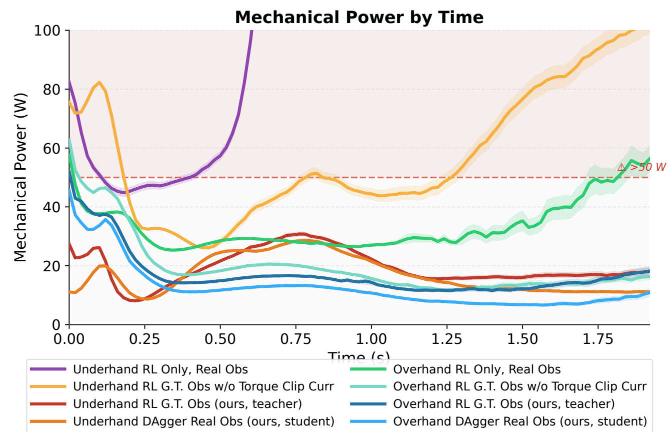

Fig. 4. **Joint mechanical power.** We plot mechanical power summed across joints, averaged over 32,768 episodes per policy. 95% CIs are shown via shaded regions (some may be too small to see). Best viewed zoomed.

Fig. 3 and Table II shows that baseline **(1)** (RL only, real obs) is ineffective, with near-zero success rate in sim and sim-to-sim evaluations. Baseline **(2)** shows that RL teacher policies trained on G.T. observations without the torque clip curriculum see degraded sim performance and very poor sim-to-sim transfer, indicating poor robustness to novel dynamics.

Meanwhile, our teacher policy **(3)** trained with the torque clip curriculum significantly outperforms baseline **(2)** without the torque clip curriculum in sim and sim-to-sim evaluations, highlighting the effectiveness of our proposed curriculum when operating at hardware limits. Our student policy **(4)** also greatly outperforms the RL-only baseline **(1)**.

Notably, both our teacher and student show strong robustness across restitution, slow decrease in performance as mass and volume increase, and good sim-to-sim transfer.

TABLE III

REWARD TERMS SIM AND SIM-TO-SIM SUCCESS RATE (%) WITH 95% CIS.

Reward	Sim		Sim-to-Sim	
	Underhand	Overhand	Underhand	Overhand
Object-Palm Distance Only	73.8 ± 0.5	58.5 ± 0.5	16.2 ± 2.1	8.2 ± 1.6
Object-Palm Stability [19]	17.1 ± 0.4	31.2 ± 0.5	18.0 ± 2.2	10.2 ± 1.8
Fingertip/Palm-Object Contact	9.0 ± 0.3	22.9 ± 0.5	5.8 ± 1.4	3.4 ± 1.1
Fingertip-Object Surface Distance (ours)	**79.5 ± 0.4**	**60.0 ± 0.5**	**81.6 ± 2.2**	**22.1 ± 2.4**
Fingertip Distance + Fingertip/Palm-Object Contact	33.1 ± 0.5	31.1 ± 0.5	15.3 ± 2.1	1.1 ± 0.7

TABLE IV

REAL-WORLD UNDERHAND AND OVERHAND CATCH SUCCESS RATE (%)

Object	Underhand SR (%) ↑	Overhand SR (%) ↑
Plush	50.0	33.3
Deformable	25.0	41.7
Rigid-Bouncy	41.7	33.3
All Objects	**38.9**	**36.1**

Although our policies are only trained on primitive objects, they generalize well to irregular object types. However, performance degrades for long-axis objects, which have high angular velocity not captured by our state parameterization. Overall, our method shows strong generalization to unseen irregular objects with variable physical properties.

3) Safe Action Execution via Torque Clip Curriculum: **Does our proposed torque clip curriculum yield safer action execution?** As shown in Sec. IV-A2, the torque clip curriculum results in improved sim and sim-to-sim performance. To investigate safe action execution, in Fig. 4 we plot summed joint mechanical power per timestep for baselines **(1)-(4)** from Sec. IV-A2. Our method with the torque clip curriculum shows low mechanical power for underhand and overhand tosses, whereas RL policies trained on G.T. observations without the torque clip curriculum exhibit much higher mechanical power. This demonstrates the effectiveness of our proposed curriculum in ensuring safe action execution.

4) Reward Ablation: **Does our proposed fingertip-object surface reward improve catching performance?** We compare five reward terms for learning effective capture during catching. Each policy utilizes all curricula as described in Sec. III-D. Object-palm distance, object height, and penalty terms described in Sec. III-C are kept the same. The default reward scale for the following rewards is 4.0:

1) **Object-Palm Distance Only**: Reward increases as distance between the object and palm decreases. Since this reward term is included by default with a reward scale of 4.0, we simply increase the reward scale to 8.0: $\exp(-2\|p_{\text{obj}} - p_{\text{goal}}\|)$
2) **Object-Palm Stability**: We incorporate the object stability reward from [19], which increases with the cumulative time the object is in the hand: $\Delta t/50.0$
3) **Fingertip/Palm-Object Contact**: First, reward increases for each finger in contact with the object up to 3 fingers. Second, if two fingers contact the object, a bonus is given for palm contact as well: $0.1 \cdot \mathbb{I}[n_{\text{finger}} \geq 1] + 0.2 \cdot \mathbb{I}[n_{\text{finger}} \geq 2] + 0.4 \cdot \mathbb{I}[n_{\text{finger}} \geq 3] + 0.3 \cdot \mathbb{I}[n_{\text{finger}} \geq 2] \cdot \mathbb{I}[\text{palm contact}]$
4) **Fingertip-Object Surface Distance (ours)**: Reward increases as the max distance from any finger to the object surface decreases: $\exp(-10(\max_{\text{ftip}} \|p_* - p_{\text{obj}}\| - r_{\text{obj}}))$
5) **Fingertip Distance + Fingertip/Palm-Object Contact**: Reward terms **(3)** and **(4)** are combined, each given an equal weight of 2.0.

Our fingertip-surface distance reward shows significant improvement in success rate for underhand and overhand in sim and sim-to-sim evaluations compared to other terms, including the object-palm stability term from prior work [19]. This demonstrates the effectiveness and necessity for this reward formulation for improving catching performance.

B. Real-World Experiments

We validate the generalization capability of our method through real-world underhand and overhand using objects unseen during training, pictured in Fig. 1. To ensure sufficient variety, we choose objects across three major categories:

1) **Plush (panda, turtle, sandbag)**: low rebound on contact, similar to sandbags used in prior work [6, 19].
2) **Deformable (Takis, strawberry, pineapple)**: deformables are unsupported by rigid-body simulators.
3) **Rigid-bouncy (green bowl, ellipsoid, bear)**: rebound off rigid surfaces, to test object capture ability.

For each real object in Fig. 1, we perform 1 long ($>1.8\,\text{m}$), 2 medium ($1.4\,\text{m}$ to $1.8\,\text{m}$), and 1 short ($1.0\,\text{m}$ to $1.4\,\text{m}$) toss. Shorter tosses require fast reaction time, while longer tosses require strong tracking. As seen in Table IV, we achieve a catching success rate of 38.9% for underhand and 36.1% across a diverse set of objects, demonstrating our method's strong sim-to-real transfer capability and generalizability. Per-object success rates are available in Appendix A.

Teleoperation baseline. We provide a teleoperated catching baseline using a bi-manual setup [8] based on Homie [1] which runs at 15 Hz. Even with an expert teleoperator, only 3/44 tosses were successful, a 6.8% success rate. The high latency and low control frequency of teleoperation make real data collection infeasible for such dynamic tasks, motivating our choice of online RL with high-throughput simulation.

V. CONCLUSION

We present *Catch It All!*, a framework for dynamic underhand and overhand catching with a dexterous hand. Through careful reward and curriculum design, we learn dexterous catching policies which generalize to unseen objects with diverse physical properties while operating near hardware limits. However, there remains room for improvement in the performance of our system, and we do not apply our framework to whole-body control. Therefore, in the future, we plan to extend our framework for dynamic dexterous manipulation near hardware limits to whole-body control, increasing the workspace and potentially improving system performance.

References

[1] Qingwei Ben, Feiyu Jia, Jia Zeng, Junting Dong, Dahua Lin, and Jiangmiao Pang. Homie: Humanoid loco-manipulation with isomorphic exoskeleton cockpit. *arXiv preprint arXiv:2502.13013*, 2025.

[2] Kehlani Fay, Darin Anthony Djapri, Anya Zorin, James Clinton, Ali El Lahib, Hao Su, Michael T. Tolley, Sha Yi, and Xiaolong Wang. Cross-embodied co-design for dexterous hands. arXiv preprint arXiv:2512.03743, December 2025. arXiv:2512.03743 [cs.RO].

[3] Benedek Forrai, Takahiro Miki, Daniel Gehrig, Marco Hutter, and Davide Scaramuzza. Event-based agile object catching with a quadrupedal robot. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 12177–12183. IEEE, 2023.

[4] Tobias Gold, Ralf Römer, Andreas Völz, and Knut Graichen. Catching objects with a robot arm using model predictive control. In *2022 American Control Conference (ACC)*, pages 1915–1920. IEEE, 2022.

[5] Tairan He, Zhengyi Luo, Xialin He, Wenli Xiao, Chong Zhang, Weinan Zhang, Kris M. Kitani, Changliu Liu, and Guanya Shi. Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard, editors, *Conference on Robot Learning, 6-9 November 2024, Munich, Germany*, volume 270 of *Proceedings of Machine Learning Research*, pages 1516–1540. PMLR, 2024. URL https://proceedings.mlr.press/v270/he25b.html.

[6] Binghao Huang, Yuanpei Chen, Tianyu Wang, Yuzhe Qin, Yaodong Yang, Nikolay Atanasov, and Xiaolong Wang. Dynamic handover: Throw and catch with bimanual hands. In Jie Tan, Marc Toussaint, and Kourosh Darvish, editors, *Conference on Robot Learning, CoRL 2023, 6-9 November 2023, Atlanta, GA, USA*, volume 229 of *Proceedings of Machine Learning Research*, pages 1887–1902. PMLR, 2023. URL https://proceedings.mlr.press/v229/huang23d.html.

[7] Jemin Hwangbo, Joonho Lee, Alexey Dosovitskiy, Dario Bellicoso, Vassilios Tsounis, Vladlen Koltun, and Marco Hutter. Learning agile and dynamic motor skills for legged robots. *Science Robotics*, 4(26):eaau5872, 2019.

[8] Guangqi Jiang, Yutong Liang, Jianglong Ye, Jia-Yang Huang, Changwei Jing, Rocky Duan, Pieter Abbeel, Xiaolong Wang, and Xueyan Zou. Cross-hand latent representation for vision-language-action models. *arXiv preprint*, 2026.

[9] Jens Kober, Matthew Glisson, and Michael Mistry. Playing catch and juggling with a humanoid robot. In *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, pages 875–881. IEEE, 2012.

[10] CMU LeCAR Lab. Humanoidverse: A multi-simulator framework for humanoid robot sim-to-real learning. https://github.com/LeCAR-Lab/HumanoidVerse, 2025.

[11] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, et al. Isaac Gym: High performance GPU-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*, 2021.

[12] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. URL https://arxiv.org/abs/2408.00714.

[13] Stephane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In Geoffrey Gordon, David Dunson, and Miroslav Dudík, editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 627–635, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR. URL https://proceedings.mlr.press/v15/ross11a.html.

[14] Nikita Rudin, David Hoeller, Philipp Reist, and Marco Hutter. Learning to walk in minutes using massively parallel deep reinforcement learning. In *Conference on Robot Learning (CoRL)*, pages 91–100. PMLR, 2022.

[15] André Schakkal, Guillaume Bellegarda, and Auke Ijspeert. Dynamic object catching with quadruped robot front legs. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024.

[16] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017. URL http://arxiv.org/abs/1707.06347.

[17] Kenneth Shaw, Ananye Agarwal, and Deepak Pathak. LEAP hand: Low-cost, efficient, and anthropomorphic hand for robot learning. In Kostas E. Bekris, Kris Hauser, Sylvia L. Herbert, and Jingjin Yu, editors, *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, 2023. doi: 10.15607/RSS.2023.XIX.089. URL https://doi.org/10.15607/RSS.2023.XIX.089.

[18] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2012, Vilamoura, Algarve, Portugal, October 7-12, 2012*, pages 5026–5033. IEEE, 2012. doi: 10.1109/IROS.2012.6386109. URL https://doi.org/10.1109/IROS.2012.6386109.

[19] Yuanhang Zhang, Tianhai Liang, Zhenyang Chen, Yanjie Ze, and Huazhe Xu. Catch it! learning to catch in flight with mobile dexterous hands. In *IEEE International Conference on Robotics and Automation, ICRA 2025, Atlanta, GA, USA, May 19-23, 2025*, pages 14385–14391. IEEE, 2025. doi: 10.1109/ICRA55743.2025.11127596. URL https://doi.org/10.1109/ICRA55743.2025.11127596.

APPENDIX

A. Real-World Results

Full per-object real-world success rates are provided below in Table V. For each real object in Fig. 1, we perform 1 long ($>1.8\,\mathrm{m}$), 2 medium ($1.4\,\mathrm{m}$ to $1.8\,\mathrm{m}$), and 1 short ($1.0\,\mathrm{m}$ to $1.4\,\mathrm{m}$) toss. Shorter tosses require fast reaction time, while longer tosses require strong tracking.

TABLE V
REAL-WORLD UNDERHAND AND OVERHAND CATCH SUCCESS RATE ($\%$)

Object	Underhand SR (%) ↑	Overhand SR (%) ↑
Plush		
Panda	75.0	25.0
Turtle	25.0	50.0
Sandbag	50.0	25.0
Deformable		
Takis	25.0	75.0
Strawberry	25.0	25.0
Pineapple	25.0	25.0
Rigid-Bouncy		
Green Bowl	50.0	25.0
Ellipsoid	25.0	50.0
Bear	50.0	25.0
All Objects	**38.9**	**36.1**