

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

Arth Shukla^{*,1}, Kehlani Fay^{*,1}, Huaze Liu², Darin Djapri¹, Alexiy Buynitsky¹, Nicklas Hansen¹

¹UC San Diego ²Harvey Mudd College ^{*}Equal contribution

<https://arth-shukla.github.io/catch-it-all>

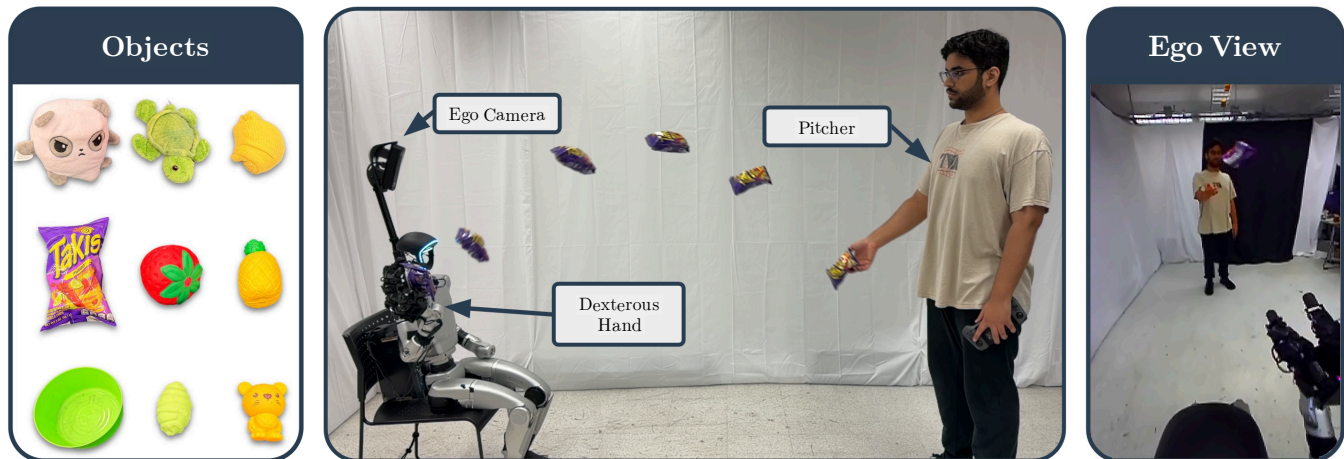


Fig. 1. **Overview.** Our method successfully catches diverse, unseen objects overhand and underhand using only an egocentric camera. Left: The real-world object test set includes 9 unseen objects with variable size, elasticity, deformability, and shape. Middle: Catching setup with the robot positioned overhand, and the 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 requires generalization to novel projectiles and real-time adaptation to unpredictable contact dynamics under noisy visual observations. This paper presents *Catch It All!*, a sim-to-real catching framework for humanoid robots that generalizes and adapts to unseen objects with variable physical properties for underhand and overhand catches, using only one egocentric camera. We learn policies which generalize to unseen objects with varying geometry, deformability, elasticity, and physical properties. Effective grasps which adapt real-time to object poses and shapes are learned through careful grasp reward design. We provide safety-aware curricula which enable safe, high-frequency action execution near hardware limits, including wrist torque limits as low as 5 N·m. We validate our system by deploying zero-shot on a Unitree G1. With only one egocentric camera, *Catch It All!* successfully performs underhand and overhand catches on 9 unseen objects not seen during training, including low-rebound plushes, semi-deformable toys and chip bags, and rigid-elastic objects with high rebound on contact.

I. INTRODUCTION

Dexterous catching is a fundamental dynamic manipulation problem, requiring precise synergy between vision and control under tight timing constraints [1], [2]. For a successful catch, robots must achieve low-latency, adaptable, and generalizable visuomotor reflexes, as thrown objects vary widely in their contact and grasp dynamics. On humanoid platforms, this problem is compounded. Unlike industrial manipulators, humanoid arms typically operate under significantly lower torque limits and control bandwidth, making it easy to exceed hardware limits and damage a robot. Humanoids rely on lighter actuators to ensure safety and work

within mobile power constraints. Catching requires learning high-frequency, reactive behaviors without exceeding these hardware limits. Prior systems demonstrate high-speed interception on quadrupeds and industrial manipulators [1], [3], [4], [5]; unlike these platforms, with wrist torque limits as low as 5 N·m, humanoids are more susceptible to hardware limits during high-frequency action execution.

Generalizable catching across diverse objects with a wide range of physical properties is a challenge. Prior sim-to-real works rely on simplified projectiles with low elasticity such as sandbags [5], [6] which are more forgiving to imprecise timing or weak grasps. In simulation, prior work has achieved catching diverse projectiles [7]; however, zero-shot transfer to general unseen objects in the real world remains an open problem. To overcome these limitations, our framework leverages careful reward and curriculum design, achieving zero-shot real world transfer with generalization to unseen objects while executing safely within hardware limits.

We present *Catch It All!*, a generalizable dexterous catching framework for underhand and overhand catching that operates on a single egocentric camera without motion capture or human demonstrations. Our contributions are: (1) generalizing to novel objects in both underhand and overhand catches through grasp reward design and a toss curriculum, (2) safe action execution near hardware limits via curricula on torque clipping and reward penalties, and (3) zero-shot sim-to-real transfer, capable of catching objects unseen in simulation both underhand and overhand.

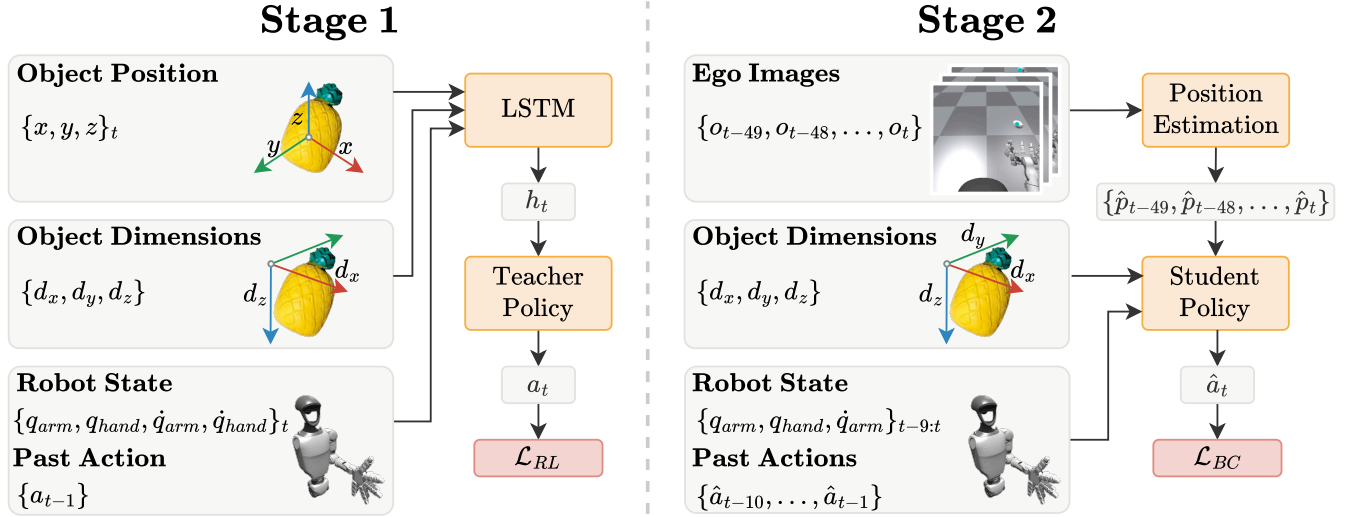


Fig. 2. **Teacher-student training.** *Stage 1:* An RL teacher policy is trained on ground-truth projectile position. *Stage 2:* A student policy imitates the frozen teacher via DAgger using noisy camera observations and a 50-step (1 second) history.

II. RELATED WORK

A. Challenges in Robotic Catching

Catching requires a robot to rapidly adapt control outputs to noisy, evolving visual estimates of an object’s state. The robot must estimate the object’s state, predict its trajectory, and plan and execute body control within milliseconds [8], [9], [10]. With an egocentric camera, the object may also leave the field of view during flight, making continuous state estimation more difficult. These estimates cannot be treated as fixed because new, noisy observations continually update and affect both the predicted trajectory and the required control correction. For catching, the controller must adapt and correct its actions rapidly online.

This challenge is compounded by the fact that the object’s motion depends on dynamics that are often unknown in advance. Properties such as spin [11] can substantially affect the predicted trajectory, so a model tuned to one object offers little guarantee of generalizing to another. The same uncertainty extends to selecting a stable catch configuration, which depends on the object’s mass, center of mass, and inertia. These properties must be estimated or learned online, under uncertainty, prior to contact [12]. At impact, sensorimotor imprecision may cause the fingers to close too abruptly, letting the object escape the hand before the grasp is complete [13]. The robot’s grasp must also quickly adapt to the object’s geometry and compliance to remain stable. Due to these complexities, catching remains an open problem.

B. Generalization Limitations

Prior work has made catching tractable by reducing task difficulty at the cost of task generalization. This has included simplifying catching by using a predefined set of known objects with dynamics fitted individually per object [8], [13]. However, this limits generalizability to unseen objects under varying contact dynamics and physical properties. Similarly, prior work has leveraged precise external motion capture systems rather than egocentric sensing [8], [13],

[14]. Egocentric vision must handle high noise characteristics of a camera mounted on a shifting robot, reducing state estimation accuracy of the object.

Other approaches simplify the object itself for simpler contact dynamics. For instance, restricting catching to sandbags, which exhibit low rebound and are more forgiving of tracking error [6], or to standard balls with no variable shape [14], [15], [16]. In the context of multi-agent catching, [5] reduces the physical toss range to a short handover between two close robot arms trained in a multi-agent setting. While this requires learning coordination between arms, it also limits the diversity of tosses the catching arm is capable of capturing. Others leverage simpler manipulators to reduce the search space and degrees of freedom, such as nets mounted on quadrupeds [3] or a standing quadruped’s front legs [4].

Moreover, extending these capabilities to humanoid platforms introduces additional challenges: compared to industrial manipulators, humanoid actuators typically exhibit lower torque limits and control bandwidth [17], [18]. Enabling dynamic dexterous manipulation on such platforms requires safe action execution at hardware limits. These simplifications in object diversity, vision, toss range, and manipulator dexterity collectively limit prior systems to narrow, controlled catching scenarios, motivating the need for an approach that catches diverse, previously unseen objects using only onboard, egocentric perception.

III. METHOD

A. System Setup

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

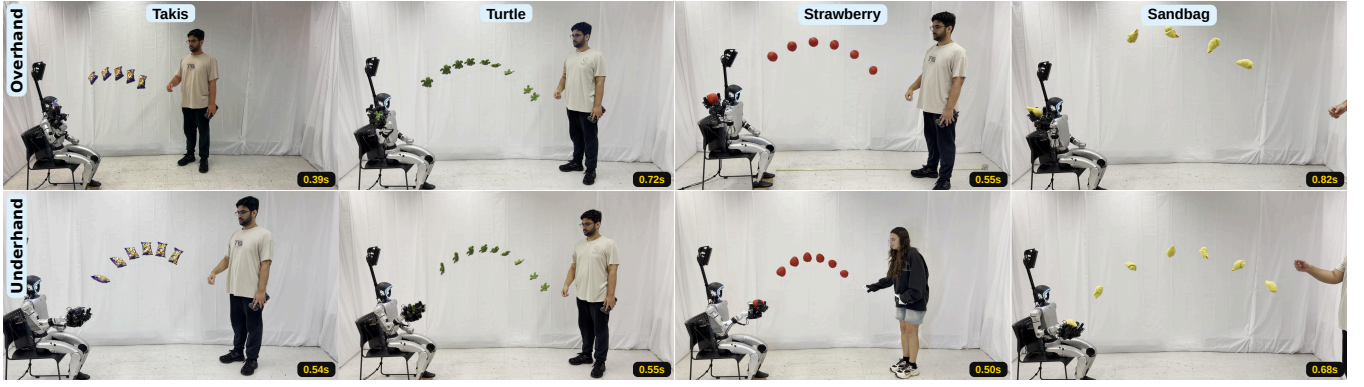


Fig. 3. **Qualitative Results.** Examples of successful catches across 4 of the 9 real-world objects: Takis, turtle, strawberry, and sandbag. Top, overhand: policy catches short tosses by moving the hand forward, while it catches long tosses by moving the hand backwards mid-capture. Bottom, underhand: the hand tracks the object during the toss to intercept and grasp it.

For the robot catching setup, we deploy on a seated Unitree G1 EDU with a ZED2i mounted overhead for full view of egocentric observations. We use the 20-DoF House of Dextra hand [19]. Compared to the 16-DoF LEAP Hand (595 g) [20] and 6-DoF Inspire RH56BFX (540 g), the House of Dextra hand provides 4 additional DoF through an extra finger, with comparable mass (600 g), and a smaller profile than LEAP. The high DoF allows for rapid adaptation with diverse grasps across objects with varying size and shape (see Fig. 9 for examples of grasps from real-world tosses). We additionally print a hollow, carbon-fiber palm to ensure the hand remains within the G1 arm’s nominal 3 kg payload.

Moreover, while we train our policies in a rigid-body simulator, our real-world experiments also include semi-deformable objects. The deformability of real objects under 1 pound of force is pictured in Fig. 6.

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-Tiny [21], and 3D bounding box dimensions for the object. The arm is controlled with full delta 6D pose via Inverse Kinematics, and the hand with absolute joint control.

We train separate underhand and overhand policies. As seen in Sec. IV-A.2, due to noisy stereo depth readings and dropped position prediction when objects leave the frame, exploration is challenging for purely online RL. So, as depicted in Fig. 2, we adopt a teacher-student framework: teacher policies are trained with PPO [22] with ground-truth object position observations, then distilled to student policies with noisy observations using DAGger [23].

To increase simulation throughput, we train on primitive shapes with simple meshes. We additionally evaluate on irregular objects and long-axis objects, depicted in Fig. 4. We

randomize object friction, mass, scale, and restitution. Tosses are initialized at 0.5 m to 2.75 m forward, ± 1.0 m lateral, and 0.25 m to 2.0 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 [24] for policy training and Mujoco [25] for sim-to-sim evaluations. Our training codebase extends HumanoidVerse [26]. Policies are trained, evaluated, 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-A.4, we show a key reward term is our fingertip-object surface distance reward, which we design to provide denser signal as fingers approach the object, or when all fingers cannot touch the object simultaneously (e.g. small/irregular objects). First, we approximate finger-surface distance with an estimated object radius r_{obj} . Importantly, this increases simulation throughput compared to true mesh-link distance computation, and is a reasonable approximation given the simpler geometries of our training set of primitive objects. Then, reward increases as the maximum distance from any finger to the object surface decreases: $\exp(-10(\max_{\text{ftip}} \|p_* - p_{\text{obj}}\| - r_{\text{obj}}))$. In Sec. IV-A.4, we also ablate against stability rewards from prior work [6], simpler distance-based rewards, and contact-based rewards. However, these have lower performance across our diverse evaluation objects, pictured in Fig. 4.

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 velocity, acceleration, torque; joint position, velocity, and torque limits; default joint position displacement; etc) [27], [28]. Equations for reward terms are available in Appendix V-A.

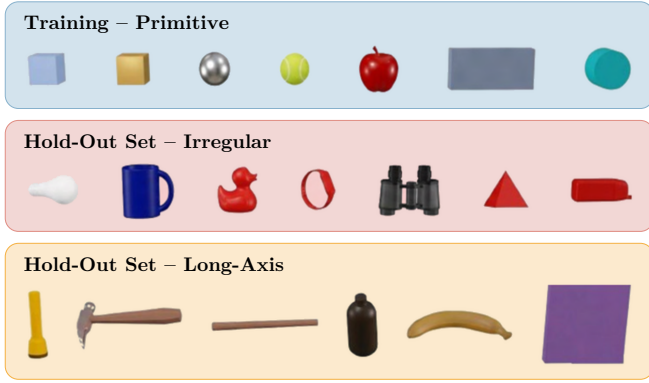


Fig. 4. **Simulation objects.** Top: training uses only primitive objects with simple meshes to increase throughput. Middle: hold-out set of irregular objects with novel geometries, unseen during training. Bottom: hold-out set of long-axis objects with high angular velocity, unseen during training.

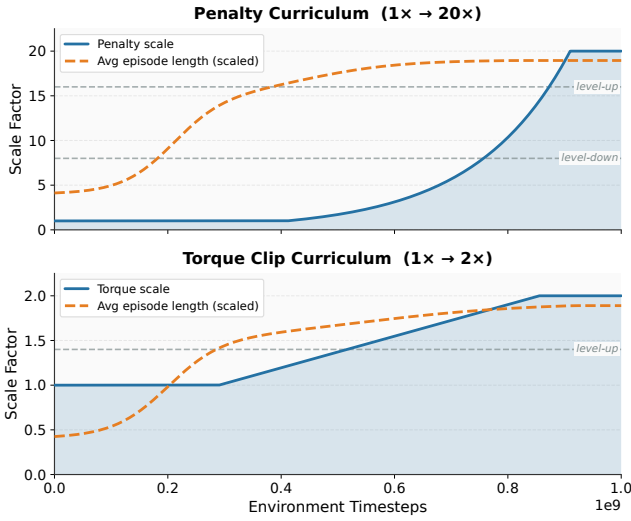


Fig. 5. **Curricula in RL training.** Penalty and reward curricula progression during RL training.

D. Curricula

In order to improve safety and performance, we introduce three curricula used during RL policy training: (1) toss target curriculum (overhand only), (2) torque clip curriculum (underhand and overhand), and (3) penalty curriculum (underhand and overhand). The progression of the torque clip and penalty curricula during training is depicted in Fig. 5.

Toss target curriculum. Learning overhand catching via online exploration is particularly difficult. To address this, we implement a toss target curriculum for overhand catches: initially, objects are thrown exactly 0.1 m above and 0.1 m behind the hand initial position, encouraging upright hand orientation when catching. As training progresses, the toss target range is iteratively 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 N·m), naive torque clipping

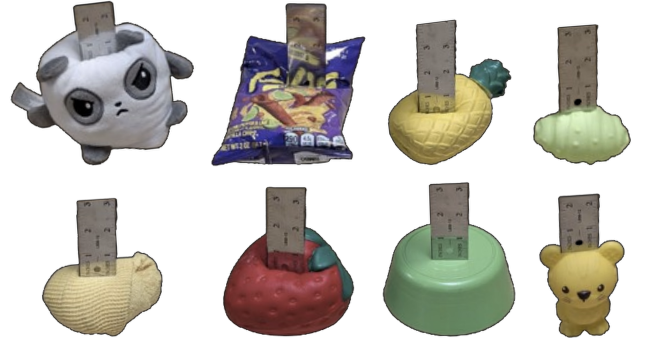


Fig. 6. **Real-world tossing objects.** Real-world projectiles unseen in training; we exert 1 pound of force to show variations in deformability, geometry, and physical properties.

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× nominal torque limits. Torques beyond nominal limits remain penalized in the reward, allowing the policy to explore effectively near actuator limits. Per Sec. IV-A.2 and IV-A.3, the torque clip curriculum improves performance while exhibiting lower joint mechanical power over time.

Penalty curriculum. Following [26], [28], we implement a penalty curriculum which scales arm and hand-related regularization penalties from 1× to 20×. Combined with the torque clip curriculum, this yields progressively greater penalties for actions beyond nominal torque limits, ensuring safer behavior.

IV. EXPERIMENTS

We evaluate our method in simulation (IsaacGym), sim-to-sim transfer (Mujoco), and real-world deployment. First, we demonstrate the generalization of our underhand and overhand policies across 20 objects pictured in Fig. 4, with variable geometry and randomized mass, elasticity, and volume. Second, we show our policies are robust to different contact dynamics, friction, and collision response via sim-to-sim in Mujoco, which enables larger-scale evaluations than is possible with real human-thrown tests. Third, we demonstrate that the torque clip curriculum yields policies which execute safer actions during catching rollouts. Fourth, we ablate the grasp reward term, comparing our per-finger distance-based grasp reward to other distance and contact-based alternatives in sim and sim-to-sim evaluations. Finally, we validate our setup by transferring zero-shot to the real world, testing underhand and overhand catching using novel objects with varying shape, deformability, and elasticity.

A. Simulation and Sim-to-Sim Transfer Evaluations

As discussed in Sec III-B, policies are trained in IsaacGym on a limited set of 7 primitive objects. Then, policies are evaluated in sim (IsaacGym) and sim-to-sim (Mujoco) on a

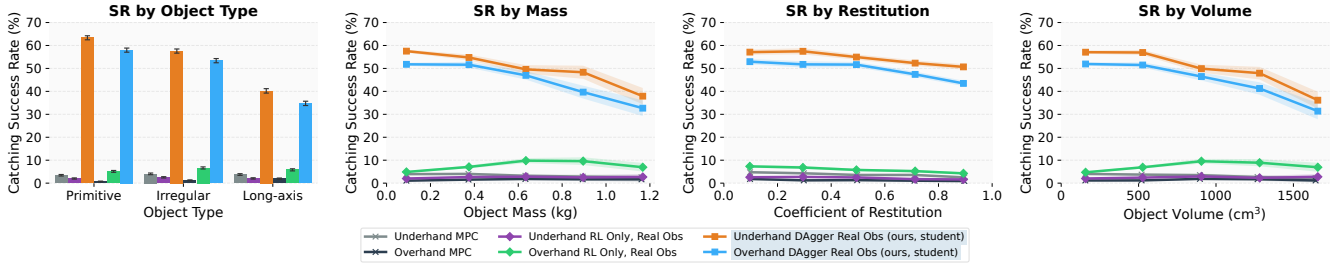


Fig. 7. **Robustness and generalization.** We test generalization to new object geometries and robustness to mass, volume, and restitution randomization. We evaluate underhand and overhand variants of the MPC, RL-only, and teacher-student distillation methods. In RL, primitive objects are seen during training, while irregular and long-axis objects are novel and unseen during training. 95% CIs are shown via error bars and shaded regions; some CIs may be too small to see. Our method is highlighted blue in the legend. Best viewed zoomed.

TABLE I. **Sim and sim-to-sim evaluations.** Success rate (%) of all policy types, 95% CIs. Highest success rate per column is bolded; second-highest is underlined; our method is highlighted blue.

Method	Sim SR (%) $\uparrow$		Sim-to-Sim SR (%) $\uparrow$	
	Underhand	Overhand	Underhand	Overhand
<i>Ground Truth Object Position Obs</i>				
MPC	3.7 $\pm$ 0.2	1.3 $\pm$ 0.1	0.1 $\pm$ 0.4	0.7 $\pm$ 0.6
RL G.T. Obs w/o Torque Clip Curriculum	<u>75.2 $\pm$ 0.5</u>	<u>51.4 $\pm$ 0.5</u>	43.8 $\pm$ 2.7	8.8 $\pm$ 1.7
RL G.T. Obs (ours, teacher)	79.5 $\pm$ 0.4	60.0 $\pm$ 0.5	81.6 $\pm$ 2.2	22.1 $\pm$ 2.4
<i>Realistic Object Position Obs (Noisy, Dropped Outside Cam Frame)</i>				
RL Only, Real Obs	2.2 $\pm$ 0.2	5.8 $\pm$ 0.3	0.1 $\pm$ 0.4	0.9 $\pm$ 0.7
DAgger Real Obs (ours, student)	54.4 $\pm$ 0.5	49.4 $\pm$ 0.5	<u>56.3 $\pm$ 2.7</u>	37.0 $\pm$ 2.7

total of 20 objects, pictured in Fig. 4: the training set of 7 primitive objects, a held-out set of 7 irregular objects with unseen geometries, and a held-out set of 6 long-axis objects with high angular velocity.

1) *Experimental Setup:* We evaluate 32,768 episodes per policy in IsaacGym, and 1280 episodes per policy in Mujoco. 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. 4 for objects used in simulation evaluation.

2) *Performance and Generalization:* To evaluate performance and generalization, we evaluate five policy types:

- 1) **MPC, G.T. Obs:** MPC [29] baselines for underhand and overhand catching.
- 2) **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 catching, and the toss target curriculum is enabled for overhand catching.
- 3) **RL, G.T. Obs w/o Torque Clip Curriculum:** RL policy trained on ground-truth (G.T.) object positions, with the torque clip curriculum disabled. The remaining curricula are used as in (2).

- 4) **RL, G.T. Obs (ours, teacher):** RL policy trained on ground-truth (G.T.) object positions. All curricula are enabled as in (2). This policy is used as the teacher for the sim-to-real deployed student policy.
- 5) **DAgger, Real Obs (ours, student):** DAgger student policy distilled from teacher policy (4). During student training, all curricula and torque clipping are disabled. This is the policy used for sim-to-real deployment.

Does our method show strong generalization to unseen objects with variable physical properties? In Fig. 7, we evaluate the distilled student policy trained under our method on objects of variable geometry, mass, restitution, and volume. Here, the G.T. teacher policy is trained with our grasp reward, and with all curricula enabled. The torque-clip curriculum and grasp reward formulation are ablated further in Sec. IV-A.3 and Sec. IV-A.4 respectively.

Baselines (1) (MPC) and (2) (RL only, real obs) are ineffective, with near-zero success rates, significantly lower than our final distilled policy. Moreover, while our policies are only trained on primitive objects, they generalize well to irregular object types with a performance degradation of approximately 5%. However, performance degrades by approximately 20% on long-axis objects, which have high angular velocity not captured by our state parameterization.

Additionally, our method exhibits minimal degradation across restitution coefficients, indicating the learned grasps and movements are robust to highly elastic objects. Moreover, while performance degrades by approximately 20% for very large mass and volume, performance only degrades by approximately 5% for masses up to 0.6 kg and volume up to 1000 cm³. Overall, our method shows strong generalization to unseen irregular objects with variable physical properties.

Is our method robust to novel robot and contact dynamics? In Table I, we evaluate all training methods in sim (IsaacGym) and sim-to-sim (Mujoco) to evaluate robustness to novel contact dynamics, friction, and collision response at a scale larger than is possible with real human-thrown tests. Here, baselines (1) and (2) again perform poorly in both sim and sim-to-sim transfer.

While baseline (3) (RL G.T. obs, w/o torque clip curriculum) achieves similar performance to (4) (our method’s RL

Mechanical Power by Time ($\downarrow$ lower is better)

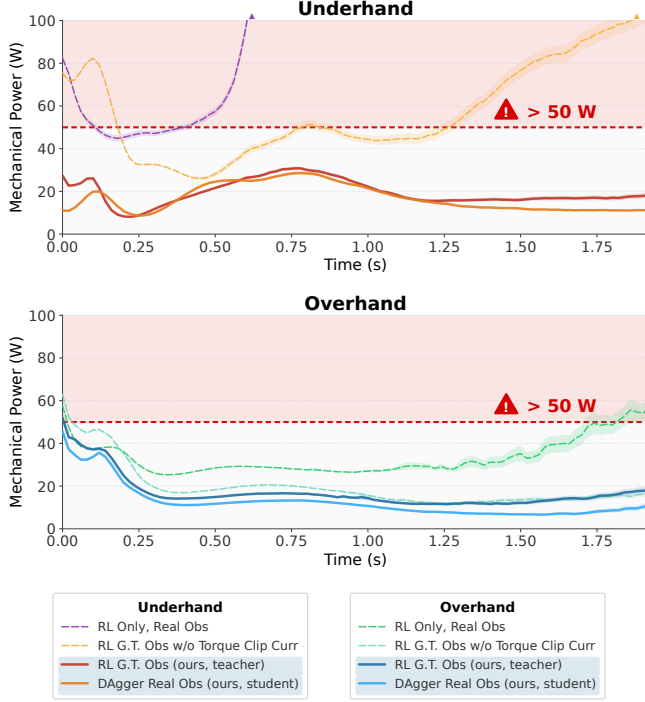


Fig. 8. **Joint mechanical power.** We plot mechanical power summed across joints, averaged over 32,768 episodes per policy. 95% CIs are shown via shaded regions; some CIs may be too small to see. Our method is highlighted blue in the legend. Best viewed zoomed.

teacher policy) in sim evaluations, its performance drops significantly in sim-to-sim evaluations. This indicates the torque clip curriculum is important for learning actions robust to novel dynamics.

Meanwhile, both our teacher policy (4) and our distilled student policy (5) outperform all other baselines in sim-to-sim transfer, demonstrating our method’s robustness to novel contact and robot dynamics.

3) *Safe Action Execution via Torque Clip Curriculum:* **Does our proposed torque clip curriculum yield safer action execution?** As discussed in Sec. IV-A.2, the torque clip curriculum is important for robustness to novel dynamics in sim-to-sim transfer. Here, we additionally investigate the torque clip curriculum’s impact on safe action execution. In Fig. 8 we plot joint mechanical power summed across joints per timestep for baselines (2)-(5) in both underhand and overhand catching.

Although baseline (2) is trained with the torque clip curriculum, it exhibits the highest joint mechanical power, likely due to its poor overall performance as discussed in Sec. IV-A.2. Meanwhile, although baseline (3) achieves similar success rate as our teacher policy (4) in sim evaluation, it exhibits significantly higher joint mechanical power in underhand catching, and slightly higher in overhand catching. Finally, our teacher policy (4) trained with the torque clip curriculum exhibits lower joint mechanical power than the

TABLE II. **Grasp reward ablation.** Sim and sim-to-sim success rate (%), 95% CIs. Highest success rate per column is bolded; second-highest is underlined; our method is highlighted blue.

Reward	Sim SR (%) $\uparrow$		Sim-to-Sim SR (%) $\uparrow$	
	Underhand	Overhand	Underhand	Overhand
Object-Palm Distance Only	73.8 $\pm$ 0.5	58.5 $\pm$ 0.5	16.2 $\pm$ 2.1	8.2 $\pm$ 1.6
Object-Palm Stability [6]	17.1 $\pm$ 0.4	31.2 $\pm$ 0.5	18.0 $\pm$ 2.2	10.2 $\pm$ 1.8
Fingertip/Palm-Object Contact	9.0 $\pm$ 0.3	22.9 $\pm$ 0.5	5.8 $\pm$ 1.4	3.4 $\pm$ 1.1
Fingertip-Object Surface Distance (ours)	79.5 $\pm$ 0.4	60.0 $\pm$ 0.5	81.6 $\pm$ 2.2	22.1 $\pm$ 2.4
Fingertip Distance + Fingertip/Palm-Object Contact	33.1 $\pm$ 0.5	31.1 $\pm$ 0.5	15.3 $\pm$ 2.1	1.1 $\pm$ 0.7

baselines, as does the distilled student policy (5).

Policies trained with the torque clip curriculum (or distilled thereof) demonstrate robustness to novel dynamics and exhibit lower joint mechanical power, demonstrating that our proposed curriculum enables safe, high-frequency action execution during catching.

4) *Reward Ablation:* **Does our proposed fingertip-object surface reward improve catching performance?** We compare five reward terms for learning effective capture during catching, including one proposed by [6]. 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 terms 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 (see Appendix V-A for details on default rewards): $\exp(-2\|p_{\text{obj}} - p_{\text{palm}}\|)$
- 2) **Object-Palm Stability:** We incorporate the object stability reward from [6], which increases with the cumulative time the object is in the hand: $\Delta t/50.0$.
- 3) **Fingertip/Palm-Object Contact:** As discussed in Sec. III-B, our distance-based reward is designed to support grasps where all fingers may not fully reach the target object (e.g. irregular geometry, varying size, etc). To ensure a fair comparison, we design our contact-based rewards similarly: first, reward increases for each finger in contact with the object up to 3 fingers. Second, if at least 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):** As described in Sec. III-C, this reward increases as the maximum 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.

Per Table II, naive object-palm distance demonstrates strong performance in sim, but poor transfer sim-to-sim. Contact-based rewards demonstrate poor performance in sim

TABLE III. **Real-world evaluations.** Underhand and overhand catch success rate (%).

Object	Underhand SR (%) $\uparrow$	Overhand SR (%) $\uparrow$
Plush	50.0	33.3
Deformable	25.0	41.7
Rigid-Elastic	41.7	33.3
All Objects	38.9	36.1



Fig. 9. **Qualitative Grasp Examples.** Examples of successful underhand and overhand grasps across different objects from real-world catching trials. Grasps adjust to object geometry and pose in-hand, yielding secure grasps.

and sim-to-sim evaluation, even combined with our proposed fingertip-object surface distance reward. The object-palm stability term from [6] demonstrates the second-strongest sim-to-sim transfer. However, our fingertip-object surface distance reward outperforms all baselines in both sim and sim-to-sim evaluations. This demonstrates the effectiveness and necessity for our proposed grasp reward for improving catching performance.

B. Real-World Experiments

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

- 1) **Plush (panda, turtle, sandbag):** low rebound on contact, like sandbags used in prior work [5], [6].
- 2) **Deformable (Takis, strawberry, pineapple):** deformables are unsupported by rigid-body simulators, and are thus novel objects for our policies.
- 3) **Rigid-elastic (green bowl, ellipsoid, bear):** these objects often rebound off rigid surfaces in unpredictable ways; we include these to test grasp quality and object capture ability.

For each real object in Fig. 1, we perform 1 long (>1.8 m), 2 medium (1.4 m to 1.8 m), and 1 short (1.0 m to 1.4 m) toss. Shorter tosses require fast reaction time, while longer tosses require effective tracking. As seen in Table III, we achieve a catching success rate of 38.9% underhand and 36.1% overhand across a diverse set of objects, demonstrating our method’s strong sim-to-real transfer capability and

generalizability. See Fig. 9 for qualitative examples of real-world underhand and overhand post-catch grasps in the real world across objects.

Infeasibility of teleoperated data collection in real world. We provide a teleoperated catching baseline using a bi-manual setup [30] based on Homie [27] which runs at 15 Hz. Even with an expert teleoperator, only 3/44 catches 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, using one egocentric camera. Through careful reward and curriculum design, we learn dexterous catching policies which generalize to unseen objects with diverse physical properties while operating safely near hardware limits. However, we do not apply our framework to whole-body control. We plan to extend our framework for dynamic dexterous manipulation near hardware limits to whole-body humanoid control, increasing the workspace and potentially improving system performance as a result.

APPENDIX

TABLE IV. **Reward equations.** RL teacher catching policy.

Reward Term	Equation	Scale
<i>Task</i>		
Approx		
Fingertip-Obj	$\exp(-10(\max_{\text{ftip}} \ p_* - p_{\text{obj}}\ - r_{\text{obj}}))$	4.0
Surface Dist		
Obj-Palm Dist	$\exp(-2\ p_{\text{obj}} - p_{\text{palm}}\)$	4.0
Obj Height	$\text{clip}(p_{\text{obj}}^z - 0.2, 0.0, 1.3)/1.3$	1.0
<i>Posture / Contact</i>		
DoF Default Pos	$\ q - q_{\text{default}}\ _1$	-1.0
Body Collision	$\ F_{\text{body}}\ _1$	-2.0
Arm Collision	$\ F_{\text{arm}}\ _1$	-1.0
<i>Joint Regularization</i>		
DoF Track Error	$\ q - q_{\text{target}}\ ^2$	-0.1
Action Rate	$\ a_t - a_{t-1}\ ^2$	-0.01
DoF Smoothness	$\ a_t - 2a_{t-1} + a_{t-2}\ ^2$	-0.05
DoF Power	$\ \dot{q}\ \odot \ \tau\ _1$	$-2e-5$
DoF Vel	$\ \dot{q}\ ^2$	$-1e-4$
DoF Acc	$\ \ddot{q}\ ^2$	$-2.5e-7$
Torques	$\ \tau \oslash K_p\ ^2$	$-2.5e-6$
DoF Pos Lims	$\ (q_{\min} - q)_+ + (q - q_{\max})_+\ _1$	-2.0
DoF Vel Lims	$\ [\dot{q}] - 0.8\dot{q}_{\max}\ _+ \ _1$	$-2e-3$
Torque Lims	$\ [\tau] - 0.95\tau_{\max}\ _+ \ _1$	-0.1
Action Vanish	$\ [a - a^{\max}]_+ + [a^{\min} - a]_+\ _1$	-1.0
<i>Hand Regularization</i>		
Hand Pos Rate	$\ p_t^{\text{hand}} - p_{t-1}^{\text{hand}}\ $	-0.05
Hand Smoothness	$\ p_t^{\text{hand}} - 2p_{t-1}^{\text{hand}} + p_{t-2}^{\text{hand}}\ $	-0.05
Hand Orientation	$\ g_{\text{hand}}^{xy}\ ^2$	-0.025

A. Reward Equations

Table IV lists all rewards, using the following variables:

p_{item}	Cartesian position of item (object, link, etc.)
r_{item}	(Approximate) radius of item
F_{item}	Force applied to item
$q, \dot{q}, \ddot{q}$	Joint position, velocity, acceleration
q_{target}	Controller joint target
q_{default}	Default joint positions
a_t	Action at timestep t
τ	Torques
$x_{\min}, x_{\max}$	Min/max values for quantity x
g_{item}	Projected gravity vector of item
K_p, K_d	PD Gains

REFERENCES

- [1] T. Gold, R. Römer, A. Völz, and K. Graichen, "Catching objects with a robot arm using model predictive control," in *2022 American Control Conference (ACC)*, 2022, pp. 1915–1920.
- [2] J. Kober, M. Glisson, and M. Mistry, "Playing catch and juggling with a humanoid robot," in *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, 2012, pp. 875–881.
- [3] B. Forrai, T. Miki, D. Gehrig, M. Hutter, and D. Scaramuzza, "Event-based agile object catching with a quadrupedal robot," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 12 177–12 183.
- [4] A. Schakkal, G. Bellegarda, and A. Ijspeert, "Dynamic object catching with quadruped robot front legs," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024, pp. 6848–6855.
- [5] B. Huang, Y. Chen, T. Wang, Y. Qin, Y. Yang, N. Atanasov, and X. Wang, "Dynamic handover: Throw and catch with bimanual hands," in *Proceedings of The 7th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, J. Tan, M. Toussaint, and K. Darvish, Eds., vol. 229. PMLR, 06–09 Nov 2023, pp. 1887–1902. [Online]. Available: <https://proceedings.mlr.press/v229/huang23d.html>
- [6] Y. Zhang, T. Liang, Z. Chen, Y. Ze, and H. Xu, "Catch it! learning to catch in flight with mobile dexterous hands," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 2025, pp. 14 385–14 391.
- [7] F. Lan, S. Wang, Y. Zhang, H. Xu, O. O. Oseni, Z. Zhang, Y. Gao, and T. Zhang, "Dexcatch: Learning to catch arbitrary objects with dexterous hands," in *Proceedings of The 8th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, vol. 270. PMLR, 2025, pp. 2965–2981. [Online]. Available: <https://proceedings.mlr.press/v270/lan25a.html>
- [8] S. Kim, A. Shukla, and A. Billard, "Catching objects in flight," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1049–1065, 2014.
- [9] B. Bäuml, T. Wimböck, and G. Hirzinger, "Kinematically optimal catching a flying ball with a hand-arm-system," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2010, pp. 2592–2599.
- [10] R. Lampariello, D. Nguyen-Tuong, C. Castellini, G. Hirzinger, and J. Peters, "Trajectory planning for optimal robot catching in real-time," in *2011 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2011, pp. 3719–3726.
- [11] Y. Zhang, Y. Zhao, R. Xiong, Y. Wang, J. Wang, and J. Chu, "Spin observation and trajectory prediction of a ping-pong ball," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 4108–4114.
- [12] N. Mavrakis and R. Stolkin, "Estimation and exploitation of objects' inertial parameters in robotic grasping and manipulation: A survey," *Robotics and Autonomous Systems*, vol. 124, p. 103374, 2020.
- [13] S. S. M. Salehian, M. Khoramshahi, and A. Billard, "A dynamical system approach for softly catching a flying object: Theory and experiment," *IEEE Transactions on Robotics*, vol. 32, no. 2, pp. 462–471, 2016.
- [14] K. Dong, K. Pereida, F. Shkurti, and A. P. Schoellig, "Catch the ball: Accurate high-speed motions for mobile manipulators via inverse dynamics learning," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 6718–6725.
- [15] B. Bäuml, O. Birbach, T. Wimböck, U. Frese, A. Dietrich, and G. Hirzinger, "Catching flying balls with a mobile humanoid: System overview and design considerations," in *2011 11th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2011, pp. 513–520.
- [16] U. Frese, B. Bäuml, S. Haidacher, G. Schreiber, I. Schäfer, M. Hähnel, and G. Hirzinger, "Off-the-shelf vision for a robotic ball catcher," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2001, pp. 1623–1629.
- [17] J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter, "Learning agile and dynamic motor skills for legged robots," *Science Robotics*, vol. 4, no. 26, p. eaau5872, 2019. [Online]. Available: <https://www.science.org/doi/abs/10.1126/scirobotics.aau5872>
- [18] N. Rudin, D. Hoeller, P. Reist, and M. Hutter, "Learning to walk in minutes using massively parallel deep reinforcement learning," in *Conference on Robot Learning (CoRL)*. PMLR, 2022, pp. 91–100.
- [19] K. Fay, D. Djapri, A. Zorin, J. Clinton, A. El Lahib, H. Su, M. Tolley, S. Yi, and X. Wang, "House of dextra : Cross-embodied co-design for dexterous hands," in *International Conference on Learning Representations*, C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust, Eds., vol. 2026, 2026, pp. 14 889–14 912. [Online]. Available: https://proceedings.iclr.cc/paper_files/paper/2026/file/18c54ed6e0cc390d750f64927dbc4e93-Paper-Conference.pdf
- [20] K. Shaw, A. Agarwal, and D. Pathak, "Leap hand: Low-cost, efficient, and anthropomorphic hand for robot learning," *Robotics: Science and Systems (RSS)*, 2023.
- [21] N. R. et al., "SAM 2: Segment anything in images and videos," in *The Thirteenth International Conference on Learning Representations*, 2025. [Online]. Available: <https://openreview.net/forum?id=Ha6RTeWMd0>
- [22] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," vol. abs/1707.06347, 2017. [Online]. Available: <http://arxiv.org/abs/1707.06347>
- [23] S. Ross, G. Gordon, and D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, G. Gordon, D. Dunson, and M. Dudík, Eds., vol. 15. Fort Lauderdale, FL, USA: PMLR, 11–13 Apr 2011, pp. 627–635. [Online]. Available: <https://proceedings.mlr.press/v15/ross11a.html>
- [24] V. Makovychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, and G. State, "Isaac Gym: High Performance GPU-Based Physics Simulation for Robot Learning," in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 28 402–28 414.
- [25] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 5026–5033.
- [26] C. L. Lab, "Humanoidverse: A multi-simulator framework for humanoid robot sim-to-real learning," <https://github.com/LeCAR-Lab/HumanoidVerse>, 2025.
- [27] Q. Ben, F. Jia, J. Zeng, J. Dong, D. Lin, and J. Pang, "HOMIE: Humanoid Loco-Manipulation with Isomorphic Exoskeleton Cockpit," in *Proceedings of Robotics: Science and Systems*, Los Angeles, CA, USA, June 2025.
- [28] T. He, Z. Luo, X. He, W. Xiao, C. Zhang, W. Zhang, K. M. Kitani, C. Liu, and G. Shi, "Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning," in *Proceedings of The 8th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, P. Agrawal, O. Kroemer, and W. Burgard, Eds., vol. 270. PMLR, 06–09 Nov 2025, pp. 1516–1540. [Online]. Available: <https://proceedings.mlr.press/v270/he25b.html>
- [29] B. Amos, I. Jimenez, J. Sacks, B. Boots, and J. Z. Kolter, "Differentiable mpc for end-to-end planning and control," in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31. Curran Associates, Inc., 2018. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2018/file/ba6d843eb4251a4526ce65d1807a9309-Paper.pdf
- [30] G. Jiang, Y. Liang, J. Ye, J.-Y. Huang, C. Jing, R. Duan, P. Abbeel, X. Wang, and X. Zou, "Cross-hand latent representation for vision-language-action models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2026, pp. 13 496–13 507.