DexH2R: Task-oriented Dexterous Manipulation from Human to Robots

Shuqi Zhao*, Xinghao Zhu*, Yuxin Chen, Chenran Li, Xiang Zhang, Mingyu Ding†, Masayoshi Tomizuka

***Abstract*— Dexterous manipulation is a critical aspect of human capability, enabling interaction with a wide variety of objects. Recent advancements in learning from human demonstrations and teleoperation have enabled progress for robots in such ability. However, these approaches either require complex data collection such as costly human effort for eye-robot contact, or suffer from poor generalization when faced with novel scenarios. To solve both challenges, we propose a framework, DexH2R, that combines human hand motion retargeting with a task-oriented residual action policy, improving task performance by bridging the embodiment gap between human and robotic dexterous hands. Specifically, DexH2R learns the residual policy directly from retargeted primitive actions and task-oriented rewards, eliminating the need for labor-intensive teleoperation systems. Moreover, we incorporate test-time guidance for novel scenarios by taking in desired trajectories of human hands and objects, allowing the dexterous hand to acquire new skills with high generalizability. Extensive experiments in both simulation and real-world environments demonstrate the effectiveness of our work, outperforming prior state-of-the-arts by 40% across various settings.**

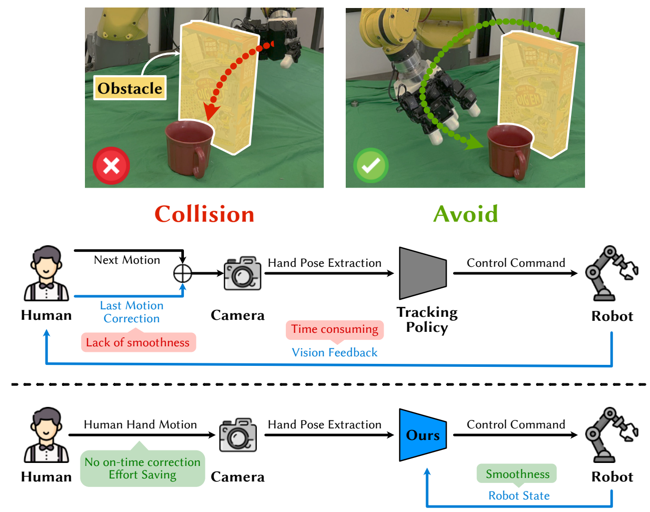

Fig. 1: When provided with the same object trajectories and initial states, DexH2R effectively navigates around obstacles by incorporating human hand trajectories, allowing it to avoid collisions (top right). In contrast, the current grasping policy offers a single solution (in red) that ignores environmental factors, often resulting in collisions (top left). Unlike traditional teleoperation systems (middle), DexH2R operates without real-time human intervention (bottom), significantly reducing human effort, ensuring smooth operation, and eliminating the need for expensive systems.

I. Introduction

Humans exhibit an extraordinary ability to manipulate a wide range of objects with their hands, both in everyday activities and industrial processes. In contrast, enabling robots to handle such diverse objects with the same level of reliability remains a significant challenge. Recent advances in reinforcement learning (RL) have led to notable progress in robotic dexterous manipulation, enabling robots to perform tasks such as grasping [1], [2], in-hand object rotation [3], and tool usage [4], [5].

However, training RL policies for dexterous manipulation remains challenging due to the large action space. For instance, attaching a Leap hand [6] to a 6-degree-of-freedom (DoF) manipulator creates a combined 22 DoF system. To solve this problem, approaches such as [7], [1], [8] incorporate pre-computed grasps into the manipulation policies as additional constraints. Methods like [9], [10], [3] take a step further by involving human demonstrations, allowing the policy to better imitate how humans interact with objects. While these human-defined grasps or trajectories [11] can improve learning efficiency, they often face generalization issues. The trained policies rely heavily on human priors, making it difficult to generalize to new scenarios, e.g., constraints or objects that require novel grasp poses.

Previous attempts to address the above problem have relied on either simulations [2] to create diverse training scenarios, or teleoperation systems [12] to collect high-quality data. While these approaches have attracted increasing attention in robotic manipulation and whole-body control, they typically assume a relatively small embodiment gap from the demonstration to the robot, as in the case of dexterous hands like the Shadow hand [13], which closely resemble the human hand and allow for efficient retargeting to initialize the policy [14], [9], [7]. However, such assumptions hinder generalization, as many cost-effective dexterous hands, such as the Allegro [15] and Leap hands [6], exhibit significant embodiment gaps, making it difficult to achieve robust retargeting. Furthermore, most teleoperation works focus on retargeting human motions to robots [16], [17], [18]. These approaches often assume that humans can adjust trajectories during execution but place less emphasis on task completion during policy training, requiring additional human attention and effort for eye-robot contact.

In this work, we address the challenges on both sides by learning generalizable and guideable robotic dexterous manipulation from human hand motions. Compared to teleoperation, in addition to using a retargeting module to provide primitive actions, we learn a residual action policy to bridge the gap between human hands and robotic dexterous hands to improve task completion. Specifically, by taking

*Equal Contribution. †Corresponding author. All authors are with the Department of Mechanical Engineering, the University of California, Berkeley.

in desired trajectories of human hands and objects, the policy is able to output a residual action to compensate for primitive actions from retargeting, thus achieving as close tracking of hand movements as possible while completing tasks. Furthermore, we enable test-time guidance for novel scenarios by incorporating human hand trajectories into the state representation. This allows the dexterous hand to learn new skills, such as adapting from training in open spaces to grasping in narrow spaces during inference.

The proposed framework offers mainly two benefits. 1) Besides primitive actions acquired through retargeting, we learn a residual action policy that bridges the gap between human and robotic dexterous hands, allowing the model to adapt to a wide range of tasks and environments. 2) We incorporate human hand trajectories into the state representation, enabling real-time guidance and skill adaptation to new and unforeseen scenarios. Together, they provide a comprehensive solution that enables a single dexterous policy to both follow human hand movements while completing tasks and generalize to novel environments using human motion cues at inference time.

With extensive experiments, DexH2R achieves success rates of 70.9% in grasping objects and 52.7% in completing the whole trajectories without dropping objects, outperforming retargeting methods commonly used in teleoperation by approximately 40%. Additionally, we demonstrate DexH2R's superiority in generalizing to novel graspings in real-world experiments.

II. Related Work

Recently, dexterous manipulation remains a big challenge due to its high-dimensional action space [19] and various, also complicated mechanical structures among different types of dexterous hands [6], [13], [15], [20] which enables dexterous hands to successfully perform different tasks such as pick-and-place, in-hand rotation, and bi-manual manipulation [21], [22], [23], [24], [23], [25]. As a result, directly applying a specially designed controller or calculating dexterous actions optimization can be limited and high-cost, thus leads to the exploration of following methods.

A. Tele-operation in Dexterous Manipulation

Due to the similarity between structures of human hands and dexterous hands, *Tele-operation* plays an important role in this area [26], [27], [28], [29], [30], [31], [32], [33]. Tele-operation normally refers to an on-time system that can take in current human motion through sensors and enable robots to perform similar actions [34], [35], [36], [37], [38], [39]. Different from teleoperation system for gripper, teleoperation system for dexterous hand requires various control for each finger and thus requires different embodiment transfer approaches. Among them, retargeting is considered a fundamental principle and has been widely applied in most systems to mapping finger motions between two embodiments [40], [18], [41], [42], [43]. However, accuracy of retargeting is limited because of multiple reasons thus needs human correction (looking at robot and adjusting its movements) to finish certain tasks. Our method, compared to teleoperation, takes task property into consideration, thus requires no human correction when transferring human motion to robot motion with finishing the tasks.

B. Reinforcement Learning in Dexterous Manipulation

Furthermore, *Reinforcement Learning* is widely introduced to dexterous manipulation to provide a task-oriented approach [44], [7], [21], [45], [46], [47], [48] for it requires no on-time correction. Focusing on grasping objects and following targeted object trajectories [49], [50], [51], PGDM [7] trains individual policy for each trajectory and emphasizes pre-grasp as a crucial feature in the object grasping phase. By leveraging real robot data, GraspGF learns a score-based grasping primitive action and trains a residual policy through RL as a task-oriented finetune [44]. Compared to their methods, ours leverages human demonstration datasets to guide dexterous hand behavior and obtains robot-object interaction information through simulation, requiring no real robot data involved.

C. Learning from Human Demonstration in Dexterous Manipulation

Due to the structure similarity between human and dexterous hand, learning from Human Demonstration[52], [53] has been widely applied and further explored in dexterous hand, and often is combined with Imitation Learning (IL) [54], [2]. By leveraging human video[10], [55], [56] or extracted human hand trajectories[57], [58], [59], [60], IL shows relatively convincing performances with less real robot data. Furthermore, [9] and [61] provide a reinforcement learning policy that enables objects to follow targeted trajectories after retrieving primitive actions through human demonstration. However, the aforementioned approach may lead to collisions when facing complicated environments such as narrow spaces. Our method, by taking in human hand trajectories, can approach objects through certain directions, thus increasing its adjustability among different situations.

III. Methods

As shown in Figure 2, our method for transferring human hand actions to robot hand actions consists of two key stages: acquiring primitive actions through retargeting and obtaining residual actions via reinforcement learning. In the first stage, we enhance the diversity of human demonstrations by applying data augmentation. Then, we derive primitive actions through retargeting optimization based on human hand motion, which provides a dexterous grasping solution at the kinematic level. Following this, we employ reinforcement learning to develop a residual policy that fine-tunes these primitive actions, producing the final actions. This approach enables the robot to successfully manipulate objects along specified trajectories. By integrating human hand demonstrations, object trajectories, and the robot's state, our method effectively transfers human grasping actions to robot hand movements during inference, ensuring successful task execution. Overall, our method can output a series of task-accomplished actions by taking in demonstration trajectories

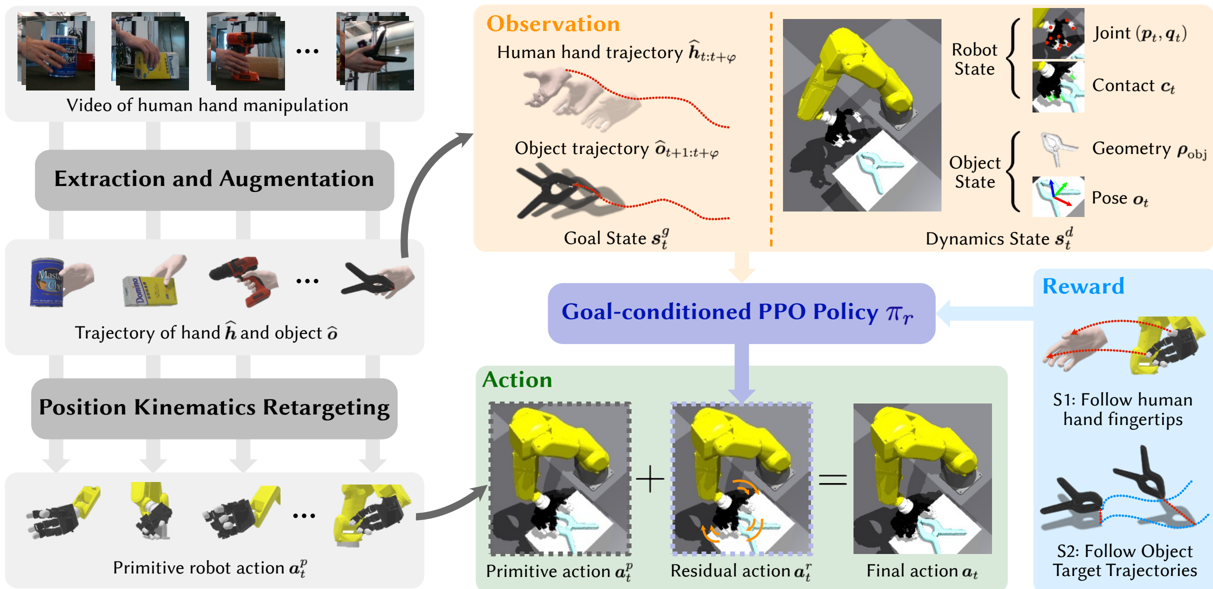

Fig. 2: **Framework overview of DexH2R.** After extraction and performing demonstration data augmentation, we obtain a large amount of trajectories distributed over all workspace, and perform position kinematics retargeting to acquire primitive actions $\boldsymbol{a}_t^p$. Then, By taking in both human hand and object trajectories, we learn a residual action $\boldsymbol{a}_t^r$ to finetune our method with task completion information, and combine two actions together as the final actions.

from both human hands and objects as well as the current robot state and object state.

A. Human Demonstration Modification

To extract the poses of human hands and objects from images of human hand grasping objects, we first identify the hand poses and keypoints using the MANO model [62]. To train generalizable policies from a limited set of human hand demonstrations, we augment both the hand and object poses within the existing dataset, as proposed in [58]. Specifically, let $\boldsymbol{\tau}^i = \{(\boldsymbol{h}_1^i, \boldsymbol{o}_1^i), (\boldsymbol{h}_2^i, \boldsymbol{o}_2^i), \ldots, (\boldsymbol{h}_N^i, \boldsymbol{o}_N^i)\}$ represent a trajectory composed of human hand poses $\boldsymbol{h}_t^i$ and object poses $\boldsymbol{o}_t^i$. The augmented trajectory $\boldsymbol{\tau}^j$ is defined as follows:

$$\boldsymbol{h}_t^j = \boldsymbol{\mathcal{T}}_i^j \boldsymbol{h}_t^i, \;\; \boldsymbol{o}_t^j = \boldsymbol{\mathcal{T}}_i^j \boldsymbol{o}_t^i, \tag{1}$$

$$\boldsymbol{\tau}^j = \left\{ \left(\boldsymbol{h}_1^j, \boldsymbol{o}_1^j\right), \ldots, \left(\boldsymbol{h}_t^j, \boldsymbol{o}_t^j\right), \ldots, \left(\boldsymbol{h}_N^j, \boldsymbol{o}_N^j\right) \right\}, \tag{2}$$

where $\boldsymbol{\mathcal{T}}_i^j$ is the transformation matrix. This transformation ensures that the relative movement between the human hand and the object remains consistent and physically plausible, meaning the direction of gravity is preserved. We perform translations of the trajectory in 3D space and rotate it along the gravity direction to maintain these properties. By applying such transformations, we generate additional feasible demonstration trajectories with varied object poses, without requiring more human data collection. This approach significantly enhances the diversity of the demonstration dataset in Cartesian space, which is crucial for effective policy training.

B. Primitive Actions: Kinematic Retargeting

To facilitate policy training and minimize redundant exploration, we initialize the policy using retargeting solutions. The retargeting process maps human hand motions to the robot's hand joints based on human demonstrations. However, due to differences in size and shape between the human and robot hands, direct keypoint mapping [18], [6] may result in infeasible robotic motions. To address this, we apply a scaling factor α to the human manipulation trajectory to approximate the size of the robot hand. Subsequently, we solve a non-linear optimization problem to minimize the topological discrepancies between human and robot hand keypoints. The optimization problem is formulated as follows:

$$\min_{\boldsymbol{q}_t} \sum_{k=0}^{N} \left\| \alpha \left(\boldsymbol{h}_t^k - \boldsymbol{o}_t \right) - \left(f_k(\boldsymbol{q}_t) - \boldsymbol{o}_t \right) \right\|^2, \tag{3}$$

$$\text{s.t. } \|\boldsymbol{q}_t - \boldsymbol{q}_{t-1}\| \le \boldsymbol{d}. \tag{4}$$

Here, $\boldsymbol{q}_t$ represents the robot's joint position at time step t, and $\boldsymbol{h}_t^k$ and $\boldsymbol{o}_t$ are the positions of the k-th human hand keypoint and the object, respectively. The function $f_k(\boldsymbol{q}_t)$ computes the position of the k-th robot hand keypoint through forward kinematics. The parameter N denotes the number of keypoints, and $\boldsymbol{d}$ is the threshold for joint position differences between consecutive frames. The parameters N, $\boldsymbol{d}$, and α can be adjusted to accommodate different robot embodiments and task-specific requirements without altering the optimization framework. Additionally, the scaling factor α reflects the fact that objects manipulated by dexterous robotic hands are generally larger than those typically handled by human hands, enabling better alignment between human demonstrations and robot actions. After solving the optimization, we define primitive action as $\boldsymbol{a}_t^p = \boldsymbol{q}_t - \boldsymbol{q}_{t-1}$ to represent the delta joint angles, which are used along with the residual actions.

C. Residual Actions: Goal-conditioned Reinforcement Learning

While the primitive action $\boldsymbol{a}_t^p$ provides a solution for manipulation, its performance can be highly unstable during real-world deployment. The retargeting process focuses solely on mapping human hand movements to the robot hand, without considering the specific task requirements. This results in an open-loop system, lacking human correction as seen in previous works [16], [17], [18], where kinematic retargeting errors propagate along the trajectory. We observed that the performance of retargeting is significantly affected by keypoint detection inaccuracies and control errors during deployment, often leading to task failure.

To address this issue, we introduce a residual policy trained through reinforcement learning. This residual policy refines the final actions by evaluating whether the objects follow the desired trajectories. By incorporating task-specific feedback, the policy compensates for errors in both keypoint detection and control, leading to improved stability and success rates during manipulation.

State. The state includes two parts and is defined as:

$$\boldsymbol{s}_t \triangleq (\boldsymbol{s}_t^d, \boldsymbol{s}_t^g), \tag{5}$$

$$\boldsymbol{s}_t^d \triangleq (\boldsymbol{\rho}_{obj}, \boldsymbol{o}_t, \boldsymbol{q}_t,\ \boldsymbol{p}_t - \boldsymbol{o}_t,\ \boldsymbol{c}_t), \tag{6}$$

$$\boldsymbol{s}_t^g \triangleq (\widehat{\boldsymbol{h}}_{t:t+\varphi}, \widehat{\boldsymbol{o}}_{t+1:t+\varphi} - \boldsymbol{o}_t), \tag{7}$$

where $\boldsymbol{s}_t^d$ stands for the dynamic state, which is decided only by the environment and goal state. $\boldsymbol{q}_t$ represents current joint positions, $\boldsymbol{p}_t - \boldsymbol{o}_t$ represents the relative positions between current robot joint positions and object positions, and the binary valuable $\boldsymbol{c}_t$ represents whether each robot finger has contact with objects. In the goal state, $\boldsymbol{\rho}_{obj}$ denotes the object information, which includes object shape and object scale. $\hat{\boldsymbol{h}}_{t:t+\varphi}$ and $\hat{\boldsymbol{o}}_{t:t+\varphi}$ represents the poses of human hand and the object from frame t (or $t{+}1$) to $t{+}\varphi$ in the demonstration.

Reward. Our reward design is divided into two stages to guide the manipulation process: (1) Following the human demonstration when the robot hand is away from the object, and (2) tracking the object trajectory when the robot hand is close to the object. The reward function is defined as follows:

$$r_t = \begin{cases} r_t^{\text{hand}} & \text{if } t > t_0 \\ r_t^{\text{obj}} & \text{if } t \le t_0 \end{cases} \tag{8}$$

When the robot hand is far from the object, the reward r_t^{hand} incentivizes the robot hand to move closer to the object. It's important to highlight that if the robot relies solely on fingertip positions from the retargeting process, the solution may not be unique. This is where residual actions become crucial, refining the retargeting optimization to produce smooth and consistent trajectories. Once the robot hand is close to the object, the reward r_t^{obj} ensures that the object follows the targeted trajectory, which is the primary objective of the manipulation task. The transition between these two stages occurs at a pre-defined switching time t_0, set as 15 steps before the object is lifted according to the human demonstration. This staged reward system ensures a smooth and accurate manipulation process, from approaching the object to completing the task. Details are shown as follow:

$$r_t^{\text{hand}} = \beta_{\text{hand}} * \exp\left(-\gamma_{\text{hand}} \textstyle\sum_{m=0}^{4} ||\widehat{\boldsymbol{h}}_t^m - \boldsymbol{p}_t^m||^2\right), \tag{9}$$

$$\begin{aligned} r_t^{\text{obj}} = \beta_{\text{obj}}^{\text{close}} * \exp\left(-\gamma_{\text{obj}}^{\text{close}} \textstyle\sum_{m=0}^{4} ||\boldsymbol{p}_t^m - \boldsymbol{o}_t||^2\right) \\ + \beta_{\text{obj}}^{\text{follow}} * \exp\left(-\gamma_{\text{obj}}^{\text{follow}} ||\widehat{\boldsymbol{o}}_t - \boldsymbol{o}_t||^2\right). \end{aligned} \tag{10}$$

Here, m represents the index for each fingertip, while both β and γ are hyperparameters.

IV. Experiments

In this section, we aim to address the following key questions: Are primitive actions sufficient for successful manipulation? Is the residual policy necessary for performance improvement? What components are critical to the success of our policy? Can our policy bridge the sim-to-real gap and be successfully deployed on a real-world robot?

A. Experimental Setup

Demonstration Dataset. For our human demonstration dataset, we utilize DexYCB [63], which focuses on grasping tasks. DexYCB contains 20 objects in total, with approximately 25 grasping trajectories for each object using various grasp poses. In our study, we select 4 objects as testing objects while using the remaining objects for training. Due to the inherent differences between human hands and the hardware of dexterous robot hands, we filter the dataset based on two key criteria: 1) The trajectory must be achievable using only rigid body contact, without relying on deformable materials to provide additional friction or forces from multiple directions (e.g., tasks like lifting an upside-down bowl). 2) The object should be stable when placed, without the risk of falling or rolling away during manipulation. We further expand the dataset using the approach detailed in Section III-A, which allows us to generate demonstrations that cover the entire workspace. Additionally, we interpolate the robot joint trajectories to match its limited velocity constraints for robots.

Baselines. We compare our method against three widely-used retargeting techniques that aim to transfer human hand motions to robot motions, which are commonly employed in teleoperation systems: 1) Position Retargeting [18]: This method minimizes the distance between the keypoints of the human hand and the robot hand. It is the same approach used to generate the primitive actions in our method. 2) Vector Retargeting [18]: This technique aligns the vector from the wrist to the fingertips between the human and robot embodiments. It also calculates the robot arm joint positions using IK based on the relative poses between the human wrist and the robot's initial frame. 3) Dexpilot Retargeting [40]: Similar to vector retargeting, this method aligns both the vector from the wrist to the fingertips and the vector between fingertips for each embodiment. The arm joint positions are also computed using IK.

Evaluation Metrics. We use four key metrics to quantitatively evaluate the performance of our method and the baseline methods: success grasp rate SR_{Grasp}, success follow rate

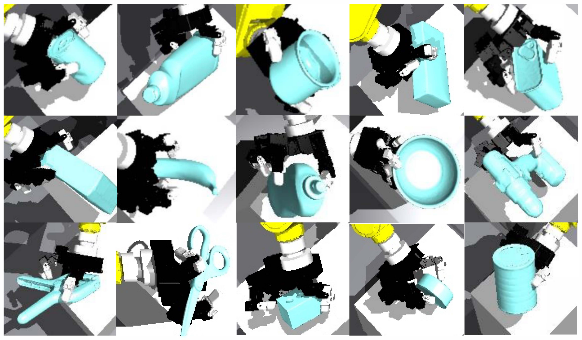

Fig. 3: Simulation Result. DexH2R can generalize to both seen and unseen objects even for those with long-tailed and rare shapes.

$\text{SR}_{\text{Follow}}$, position error E_p (m) and rotation error E_r (radian). SR_{Grasp} represents that a successful grasp and lift happens during the whole trajectory. $\text{SR}_{\text{Follow}}$ represents a successful trajectory, when objects have been constantly lifted without falling off from the robot hand during the whole trajectory. Two success rates are counted among the whole test set. E_p and E_r are the average distances and angles between the current trajectory and the targeted trajectory after the objects are supposed to be grasped. Those two parameters are only counted in successful trajectories.

B. Simulation Experiments

Implementation details. We use Isaac Gym [64] for simulation. The robot's control frequency is set to 10 Hz, and we utilize a PD controller for low-level position control. We employ PPO [65] to train the residual policy. For encoding features, all inputs are encoded with MLP except for the object shape, which is presented with point cloud and encoded using PointNeXt [66].

Results. As shown in Figure 4, when coming across unseen obstacles, DexH2R can reach objects by following given human hand trajectories, while method without hand input collides and fails to finish tasks. This strongly demonstrates that our method can be effectively adapted to different environments through human guidance. The quantitative results and visualizations from our simulation experiments are presented in Table I and Figure 3. From these results, it is evident that without real-time human correction, all the baseline retargeting methods fail to achieve optimal performance. In contrast, our method (DexH2R) significantly outperforms the baselines in both success rates and tracking errors. This demonstrates the necessity of fine-tuning the human-to-robot policy by considering the desired object poses and human-object interactions. Additionally, it is important to note the reduced gap between the grasp success rate and the follow success rate in DexH2R. This indicates that our method is more effective at preventing objects from falling once they are lifted, highlighting the robustness of the grasp and the improved coordination of the robot's fingers during object movement and rotation. Moreover, DexH2R achieves convincing results for both seen and unseen objects, demonstrating its strong generalizability across different environments

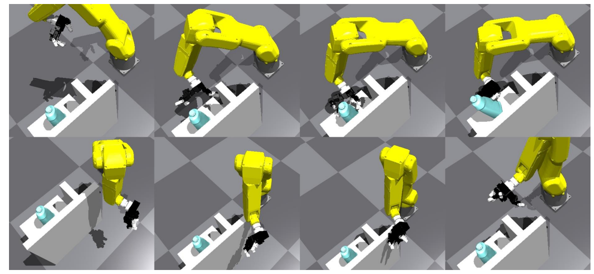

Fig. 4: Collision avoidance in novel scenarios. When facing new obstacles or scenes, DexH2R (top) can effectively avoid collision while other methods (bottom) can not.

TABLE I: Quantitative results for seen objects, unseen trajectories and unseen objects with our method.

Test set	Method	$\text{SR}_{\text{Grasp}} \uparrow$	$\text{SR}_{\text{Follow}} \uparrow$	$E_p \downarrow$	$E_r \downarrow$
Seen objects	Position[18]	32.0%	13.7%	0.083	0.364
	Vector[18]	32.7%	9.1%	0.108	0.649
	Dexpilot[40]	32.8%	10.7%	0.111	0.523
	Ours	**69.8%**	**52.1%**	**0.048**	**0.452**
Unseen objects	Position[18]	29.0%	14.3%	0.091	0.389
	Vector[18]	31.8%	10.2%	0.103	0.533
	Dexpilot[40]	32.2%	11.0%	0.105	0.545
	Ours	**70.9%**	**52.7%**	**0.055**	**0.476**

and tasks. This ability to generalize further confirms the effectiveness of our approach in transferring human hand actions to robot hands for complex manipulation tasks.

C. Ablation Studies and analysis

As demonstrated in Table II, we investigate the impact of different features by systematically removing them from our model. Across all experiments, performance declined when specific features were omitted, but it is clear that the hand-reward guidance is the most critical component. Without the guidance provided by human hand rewards, the exploration of the residual policy becomes almost unmanageable due to the high-dimensional action space, which aligns with the challenges described in [7]. This highlights the necessity of hand guidance in enabling effective policy learning. Additionally, data augmentation plays a significant role in improving final performance. This is expected, as data augmentation allows the dataset to cover a broader range of the workspace, providing the policy with more diverse examples during training. When primitive actions are removed, although the performance does not decline drastically, the convergence time for the policy significantly increases. This underscores the benefit of primitive actions in speeding up the learning process, as they provide a good initialization for the policy. Both hand guidance with a long horizon and contact information in observations contribute positively to the final results. Long-horizon hand guidance allows the policy to predict future states more effectively, resulting in smoother actions, similar to motion prediction control. Meanwhile, contact information helps the policy understand the interaction between the hand and the object, providing a direct representation of grasping dynamics that other features may not fully capture. We also conducted

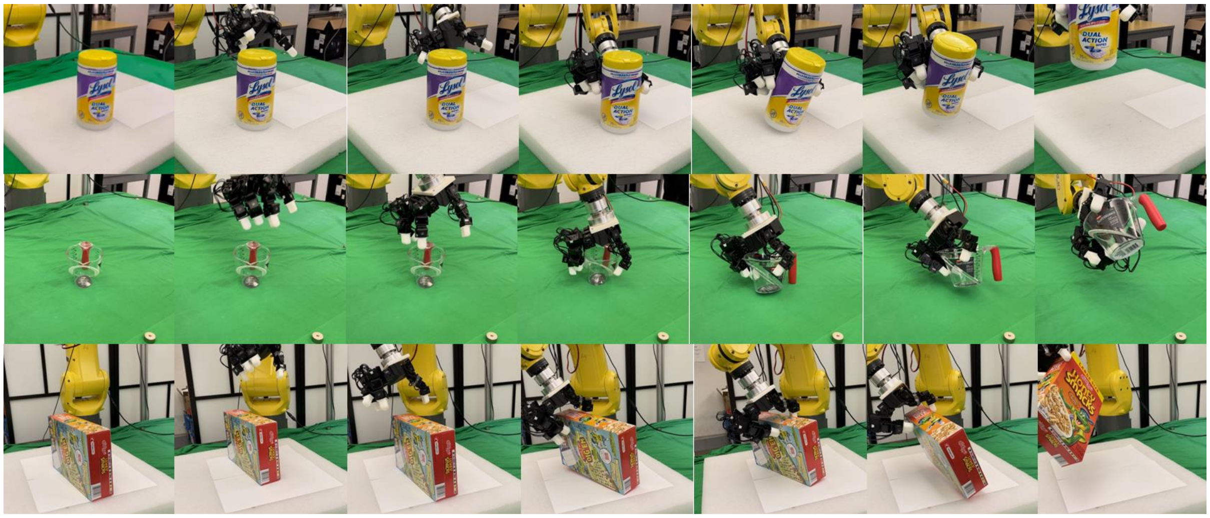

Fig. 5: Real-world experiments. DexH2R can generalize to various objects with each row as a manipulation action sequence.

TABLE II: Ablation study of our method on unseen objects.

Method	SR_{Grasp} ↑	SR_{Follow} ↑	E_p ↓	E_r ↓
Ours	**70.9%**	**52.7%**	**0.055**	**0.476**
w/o data aug	54.7%	34.1%	0.079	0.557
w/o horizon hand	65.8%	43.6%	0.061	0.525
w/o contacts	67.6%	47.5%	0.056	0.511
w/o prim actions	66.3%	47.4%	0.055	0.553
w/ sparse reward	17.4%	2.7%	0.094	0.487

TABLE III: Performance of our method on different objects.

Objects		SR_{Grasp} ↑	SR_{Follow} ↑	E_p ↓	E_r ↓
Seen objects	master chef can	84.7%	63.3%	0.035	0.259
	bleach cleanser	87.5%	75.9%	0.031	0.301
	scissors	31.2%	29.6%	0.066	0.598
	power drill	65.8%	34.8%	0.05	0.408
	Overall	**69.8%**	**52.1%**	**0.048**	**0.452**
Unseen objects	potted meat can	71.9%	57.3%	0.045	0.496
	mug	62.0%	47.0%	0.062	0.489
	Overall	**70.9%**	**52.7%**	**0.055**	**0.476**

experiments to assess how object properties affect performance. As demonstrated in Table III, the dexterous hand achieved higher success rates with objects that have regular shapes and suitable sizes. These objects present more feasible grasping solutions compared to irregularly shaped or overly large or small objects. This finding highlights the importance of object properties in determining the effectiveness of manipulation tasks.

D. Real-world Experiments

Our robot system consists of a FANUC LR Mate robot with a Leap Hand[6]. We use an Intel RealSense D415 to capture the scene. To validate our policy, we evaluated it on four different objects, including two seen and two unseen from the YCB dataset [67]. We scaled up objects to match the dimensions used in human demonstrations. We conducted 10 experiments for each object with open-loop execution, recording two key metrics: the grasp success rate and the follow success rate. These metrics help assess the system's ability to not only grasp objects successfully but also maintain control during object manipulation.

As shown in Table IV, our real-world experiments demonstrate that the proposed method successfully overcomes the sim-to-real gap. All four objects achieved relatively high performance in terms of grasp success. However, there was a noticeable decline in the success rate for maintaining a grasp without object slippage or falling during manipulation. During the experiments, we observed that when objects require rotation, the real-world dexterous hand struggles with finger coordination due to variations in object mass, friction, and contact modes. For objects with regular shapes, the hand pose remains fairly consistent throughout the manipulation process, which simplifies control. However, for objects with irregular shapes, adapting finger modalities to adjust for object rotation is crucial, as the mass distribution can shift from frame to frame. This indicates that dynamic adjustments are vital for handling complex objects, where finger coordination must account for these variations to ensure stability during manipulation.

TABLE IV: Success rates of real-world experiments on both seen and unseen objects.

Objects		SR_{Grasp} ↑	SR_{Follow} ↑
Seen objects	sugar box	10/10	9/10
	tomato soup can	9/10	7/10
Unseen objects	mug	8/10	5/10
	mustard bottle	8/10	4/10

V. Conclusions and Future Work

In this work, we introduce DexH2R, which can transfer human hand motion to robot dexterous hand motion while finishing the tasks. By taking in trajectories of both human hands and objects, DexH2R requires no on-time human correction to finish certain tasks and ensures generalization capability among complicated environments such as narrow space. With generalizability among different embodiments, DexH2R has large potential to be a data collection method, which would lower the demand for hardware in this area. As a result, the future work may lie in generalizing DexH2R into different tasks such as in-hand manipulation.

References

[1] Y. Xu, W. Wan, J. Zhang, H. Liu, Z. Shan, H. Shen, R. Wang, H. Geng, Y. Weng, J. Chen, *et al.*, "Unidexgrasp: Universal robotic dexterous grasping via learning diverse proposal generation and goal-conditioned policy," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4737–4746.

[2] W. Wan, H. Geng, Y. Liu, Z. Shan, Y. Yang, L. Yi, and H. Wang, "Unidexgrasp++: Improving dexterous grasping policy learning via geometry-aware curriculum and iterative generalist-specialist learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 3891–3902.

[3] X. Zhu, J. Ke, Z. Xu, Z. Sun, B. Bai, J. Lv, Q. Liu, Y. Zeng, Q. Ye, C. Lu, *et al.*, "Diff-lfd: Contact-aware model-based learning from visual demonstration for robotic manipulation via differentiable physics-based simulation and rendering," in *Conference on Robot Learning*. PMLR, 2023, pp. 499–512.

[4] T. Lin, Y. Zhang, Q. Li, H. Qi, B. Yi, S. Levine, and J. Malik, "Learning visuotactile skills with two multifingered hands," *arXiv:2404.16823*, 2024.

[5] T. Lin, Z.-H. Yin, H. Qi, P. Abbeel, and J. Malik, "Twisting lids off with two hands," *arXiv:2403.02338*, 2024.

[6] K. Shaw, A. Agarwal, and D. Pathak, "Leap hand: Low-cost, efficient, and anthropomorphic hand for robot learning," *arXiv preprint arXiv:2309.06440*, 2023.

[7] S. Dasari, A. Gupta, and V. Kumar, "Learning dexterous manipulation from exemplar object trajectories and pre-grasps," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 3889–3896.

[8] T. Wu, Y. Gan, M. Wu, J. Cheng, Y. Yang, Y. Zhu, and H. Dong, "Unidexfpm: Universal dexterous functional pre-grasp manipulation via diffusion policy," *arXiv preprint arXiv:2403.12421*, 2024.

[9] Z. Chen, S. Chen, C. Schmid, and I. Laptev, "Vividex: Learning vision-based dexterous manipulation from human videos," *arXiv preprint arXiv:2404.15709*, 2024.

[10] Y. Qin, Y.-H. Wu, S. Liu, H. Jiang, R. Yang, Y. Fu, and X. Wang, "Dexmv: Imitation learning for dexterous manipulation from human videos," in *European Conference on Computer Vision*. Springer, 2022, pp. 570–587.

[11] Y. Fan, X. Zhu, and M. Tomizuka, "Optimization model for planning precision grasps with multi-fingered hands," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 1548–1554.

[12] T. He, Z. Luo, X. He, W. Xiao, C. Zhang, W. Zhang, K. Kitani, C. Liu, and G. Shi, "Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning," *arXiv preprint arXiv:2406.08858*, 2024.

[13] 'shadow hand'. [Online]. Available: https://www.wonikrobotics.com/research-robot-hand

[14] Y. Qin, Y.-H. Wu, S. Liu, H. Jiang, R. Yang, Y. Fu, and X. Wang, "Dexmv: Imitation learning for dexterous manipulation from human videos," *arXiv preprint arXiv:2403.04436*.

[15] 'allegro hand'. [Online]. Available: https://ninjatek.com/shop/edge/

[16] T. He, Z. Luo, W. Xiao, C. Zhang, K. Kitani, C. Liu, and G. Shi, "Learning human-to-humanoid real-time whole-body teleoperation," *arXiv preprint arXiv:2403.04436*, 2024.

[17] X. Cheng, J. Li, S. Yang, G. Yang, and X. Wang, "Open-television: teleoperation with immersive active visual feedback," *arXiv preprint arXiv:2407.01512*, 2024.

[18] Y. Qin, W. Yang, B. Huang, K. Van Wyk, H. Su, X. Wang, Y.-W. Chao, and D. Fox, "Anyteleop: A general vision-based dexterous robot arm-hand teleoperation system," *arXiv preprint arXiv:2307.04577*, 2023.

[19] H. Mania, A. Guy, and B. Recht, "Simple random search provides a competitive approach to reinforcement learning," *arXiv preprint arXiv:1803.07055*, 2018.

[20] Z. Si, K. Zhang, O. Kroemer, and F. Z. Temel, "Deltahands: A synergistic dexterous hand framework based on delta robots," *IEEE Robotics and Automation Letters*, 2024.

[21] A. Rajeswaran, V. Kumar, A. Gupta, G. Vezzani, J. Schulman, E. Todorov, and S. Levine, "Learning complex dexterous manipulation with deep reinforcement learning and demonstrations," *arXiv preprint arXiv:1709.10087*, 2017.

[22] T. Chen, J. Xu, and P. Agrawal, "A system for general in-hand object re-orientation," in *Conference on Robot Learning*. PMLR, 2022, pp. 297–307.

[23] W. Huang, I. Mordatch, P. Abbeel, and D. Pathak, "Generalization in dexterous manipulation via geometry-aware multi-task learning," *arXiv preprint arXiv:2111.03062*, 2021.

[24] A. Gupta, J. Yu, T. Z. Zhao, V. Kumar, A. Rovinsky, K. Xu, T. Devlin, and S. Levine, "Reset-free reinforcement learning via multi-task learning: Learning dexterous manipulation behaviors without human intervention," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 6664–6671.

[25] Y. Chen, T. Wu, S. Wang, X. Feng, J. Jiang, Z. Lu, S. McAleer, H. Dong, S.-C. Zhu, and Y. Yang, "Towards human-level bimanual dexterous manipulation with reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 5150–5163, 2022.

[26] Q. Gao, Z. Ju, Y. Chen, Q. Wang, and C. Chi, "An efficient rgb-d hand gesture detection framework for dexterous robot hand-arm teleoperation system," *IEEE Transactions on Human-Machine Systems*, vol. 53, no. 1, pp. 13–23, 2022.

[27] Y. Qin, H. Su, and X. Wang, "From one hand to multiple hands: Imitation learning for dexterous manipulation from single-camera teleoperation," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 873–10 881, 2022.

[28] Z. Si, K. L. Zhang, Z. Temel, and O. Kroemer, "Tilde: Teleoperation for dexterous in-hand manipulation learning with a deltahand," *arXiv preprint arXiv:2405.18804*, 2024.

[29] S. Li, J. Jiang, P. Ruppel, H. Liang, X. Ma, N. Hendrich, F. Sun, and J. Zhang, "A mobile robot hand-arm teleoperation system by vision and imu," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 10 900–10 906.

[30] S. Li, X. Ma, H. Liang, M. Görner, P. Ruppel, B. Fang, F. Sun, and J. Zhang, "Vision-based teleoperation of shadow dexterous hand using end-to-end deep neural network," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 416–422.

[31] S. Li, N. Hendrich, H. Liang, P. Ruppel, C. Zhang, and J. Zhang, "A dexterous hand-arm teleoperation system based on hand pose estimation and active vision," *IEEE Transactions on Cybernetics*, vol. 54, no. 3, pp. 1417–1428, 2022.

[32] R. M. Aronson and H. Admoni, "Gaze complements control input for goal prediction during assisted teleoperation," in *Robotics science and systems*, 2022.

[33] T. Zhang, Z. McCarthy, O. Jow, D. Lee, X. Chen, K. Goldberg, and P. Abbeel, "Deep imitation learning for complex manipulation tasks from virtual reality teleoperation," in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 5628–5635.

[34] H. Liu, X. Xie, M. Millar, M. Edmonds, F. Gao, Y. Zhu, V. J. Santos, B. Rothrock, and S.-C. Zhu, "A glove-based system for studying hand-object manipulation via joint pose and force sensing," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 6617–6624.

[35] H. Liu, Z. Zhang, X. Xie, Y. Zhu, Y. Liu, Y. Wang, and S.-C. Zhu, "High-fidelity grasping in virtual reality using a glove-based system," in *2019 international conference on robotics and automation (icra)*. IEEE, 2019, pp. 5180–5186.

[36] S. Liu, H. Jiang, J. Xu, S. Liu, and X. Wang, "Semi-supervised 3d hand-object poses estimation with interactions in time," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 687–14 697.

[37] S. P. Arunachalam, I. Güzey, S. Chintala, and L. Pinto, "Holo-dex: Teaching dexterity with immersive mixed reality," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 5962–5969.

[38] J. I. Lipton, A. J. Fay, and D. Rus, "Baxter's homunculus: Virtual reality spaces for teleoperation in manufacturing," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 179–186, 2017.

[39] P. Ponomareva, D. Trinitatova, A. Fedoseev, I. Kalinov, and D. Tsetserukou, "Grasplook: a vr-based telemanipulation system with r-cnn-driven augmentation of virtual environment," in *2021 20th International Conference on Advanced Robotics (ICAR)*. IEEE, 2021, pp. 166–171.

[40] A. Handa, K. Van Wyk, W. Yang, J. Liang, Y.-W. Chao, Q. Wan, S. Birchfield, N. Ratliff, and D. Fox, "Dexpilot: Vision-based teleoperation of dexterous robotic hand-arm system," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9164–9170.

[41] D. Antotsiou, G. Garcia-Hernando, and T.-K. Kim, "Task-oriented hand motion retargeting for dexterous manipulation imitation," in

Proceedings of the European conference on computer vision (ECCV) workshops, 2018, pp. 0–0.

[42] S. P. Arunachalam, S. Silwal, B. Evans, and L. Pinto, "Dexterous imitation made easy: A learning-based framework for efficient dexterous manipulation," in *2023 ieee international conference on robotics and automation (icra)*. IEEE, 2023, pp. 5954–5961.

[43] Z. Chen, S. Kiami, A. Gupta, and V. Kumar, "Genaug: Retargeting behaviors to unseen situations via generative augmentation," *arXiv preprint arXiv:2302.06671*, 2023.

[44] T. Wu, M. Wu, J. Zhang, Y. Gan, and H. Dong, "Graspgf: learning score-based grasping primitive for human-assisting dexterous grasping," in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 2023, pp. 22 132–22 150.

[45] H. Zhu, A. Gupta, A. Rajeswaran, S. Levine, and V. Kumar, "Dexterous manipulation with deep reinforcement learning: Efficient, general, and low-cost," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 3651–3657.

[46] G. Khandate, S. Shang, E. T. Chang, T. L. Saidi, Y. Liu, S. M. Dennis, J. Adams, and M. Ciocarlie, "Sampling-based exploration for reinforcement learning of dexterous manipulation," *arXiv preprint arXiv:2303.03486*, 2023.

[47] Q. She, R. Hu, J. Xu, M. Liu, K. Xu, and H. Huang, "Learning high-dof reaching-and-grasping via dynamic representation of gripper-object interaction," *arXiv preprint arXiv:2204.13998*, 2022.

[48] A. Nagabandi, K. Konolige, S. Levine, and V. Kumar, "Deep dynamics models for learning dexterous manipulation," in *Conference on Robot Learning*. PMLR, 2020, pp. 1101–1112.

[49] J. Bae, J. Won, D. Lim, C.-H. Min, and Y. M. Kim, "Pmp: Learning to physically interact with environments using part-wise motion priors," in *ACM SIGGRAPH 2023 Conference Proceedings*, 2023, pp. 1–10.

[50] Y. Wang, J. Lin, A. Zeng, Z. Luo, J. Zhang, and L. Zhang, "Physhoi: Physics-based imitation of dynamic human-object interaction," *arXiv preprint arXiv:2312.04393*, 2023.

[51] J. Braun, S. Christen, M. Kocabas, E. Aksan, and O. Hilliges, "Physically plausible full-body hand-object interaction synthesis," in *2024 International Conference on 3D Vision (3DV)*. IEEE, 2024, pp. 464–473.

[52] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and autonomous systems*, vol. 57, no. 5, pp. 469–483, 2009.

[53] P. Englert and M. Toussaint, "Learning manipulation skills from a single demonstration," *The International Journal of Robotics Research*, vol. 37, no. 1, pp. 137–154, 2018.

[54] C. Wang, L. Fan, J. Sun, R. Zhang, L. Fei-Fei, D. Xu, Y. Zhu, and A. Anandkumar, "Mimicplay: Long-horizon imitation learning by watching human play," *arXiv preprint arXiv:2302.12422*, 2023.

[55] A. Sivakumar, K. Shaw, and D. Pathak, "Robotic telekinesis: Learning a robotic hand imitator by watching humans on youtube," *arXiv preprint arXiv:2202.10448*, 2022.

[56] H. Bharadhwaj, A. Gupta, V. Kumar, and S. Tulsiani, "Towards generalizable zero-shot manipulation via translating human interaction plans," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 6904–6911.

[57] C. Wang, H. Shi, W. Wang, R. Zhang, L. Fei-Fei, and C. K. Liu, "Dexcap: Scalable and portable mocap data collection system for dexterous manipulation," *arXiv preprint arXiv:2403.07788*, 2024.

[58] Z. Q. Chen, K. Van Wyk, Y.-W. Chao, W. Yang, A. Mousavian, A. Gupta, and D. Fox, "Dextransfer: Real world multi-fingered dexterous grasping with minimal human demonstrations," *arXiv preprint arXiv:2209.14284*, 2022.

[59] J. Gu, S. Kirmani, P. Wohlhart, Y. Lu, M. G. Arenas, K. Rao, W. Yu, C. Fu, K. Gopalakrishnan, Z. Xu, *et al.*, "Rt-trajectory: Robotic task generalization via hindsight trajectory sketches," *arXiv preprint arXiv:2311.01977*, 2023.

[60] M. Xu, Z. Xu, C. Chi, M. Veloso, and S. Song, "Xskill: Cross embodiment skill discovery," in *Conference on Robot Learning*. PMLR, 2023, pp. 3536–3555.

[61] Z. Luo, J. Cao, S. Christen, A. Winkler, K. Kitani, and W. Xu, "Grasping diverse objects with simulated humanoids," *arXiv preprint arXiv:2407.11385*, 2024.

[62] J. Romero, D. Tzionas, and M. J. Black, "Embodied hands: Modeling and capturing hands and bodies together," *arXiv preprint arXiv:2201.02610*, 2022.

[63] Y.-W. Chao, W. Yang, Y. Xiang, P. Molchanov, A. Handa, J. Tremblay, Y. S. Narang, K. Van Wyk, U. Iqbal, S. Birchfield, *et al.*, "Dexycb: A benchmark for capturing hand grasping of objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9044–9053.

[64] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, *et al.*, "Isaac gym: High performance gpu-based physics simulation for robot learning," *arXiv preprint arXiv:2108.10470*, 2021.

[65] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[66] G. Qian, Y. Li, H. Peng, J. Mai, H. Hammoud, M. Elhoseiny, and B. Ghanem, "Pointnext: Revisiting pointnet++ with improved training and scaling strategies," *Advances in neural information processing systems*, vol. 35, pp. 23 192–23 204, 2022.

[67] B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar, "The ycb object and model set: Towards common benchmarks for manipulation research," in *2015 international conference on advanced robotics (ICAR)*. IEEE, 2015, pp. 510–517.