# Dexterous Manipulation Transfer via Progressive Kinematic-Dynamic Alignment

**Wenbin Bai[1], Qiyu Chen[1], Xiangbo Lin[1]*, Jw L[2], Quancheng Li[1], Hejiang Pan[1], Yi Sun[1]**

[1]School of Information and Communication Engineering, Dalian University of Technology, China
[2]Shenyang Institute of Automation, Chinese Academy of Sciences, China
{baiwenbin6,20201071515,c,318546410}@mail.dlut.edu.cn, {linxbo,lslwf}@dlut.edu.cn,lijianwen@sia.cn

## Abstract

The inherent difficulty and limited scalability of collecting manipulation data using multi-fingered robot hand hardware platforms have resulted in severe data scarcity, impeding research on data-driven dexterous manipulation policy learning. To address this challenge, we present a hand-agnostic manipulation transfer system. It efficiently converts human hand manipulation sequences from demonstration videos into high-quality dexterous manipulation trajectories without requirements of massive training data. To tackle the multi-dimensional disparities between human hands and dexterous hands, as well as the challenges posed by high-degree-of-freedom coordinated control of dexterous hands, we design a progressive transfer framework: first, we establish primary control signals for dexterous hands based on kinematic matching; subsequently, we train residual policies with action space rescaling and thumb-guided initialization to dynamically optimize contact interactions under unified rewards; finally, we compute wrist control trajectories with the objective of preserving operational semantics. Using only human hand manipulation videos, our system automatically configures system parameters for different tasks, balancing kinematic matching and dynamic optimization across dexterous hands, object categories, and tasks. Extensive experimental results demonstrate that our framework can automatically generate smooth and semantically correct dexterous hand manipulation that faithfully reproduces human intentions, achieving high efficiency and strong generalizability with an average transfer success rate of 73%, providing an easily implementable and scalable method for collecting robot dexterous manipulation data. Refer to the arXiv version for the appendix.

## Introduction

Transferring human capabilities in flexible object manipulation to multi-fingered dexterous robotic hands has long been a core topic in robotics. Dexterous manipulation transfer is crucial for two reasons: (a) transferring human skills into robotic productivity, allows robots to use everyday tools to better assist human life; (b) generating manipulation data, facilitates the development of data-driven robotic systems. Compared to real-world transfer, simulation-based transfer offers advantages such as higher data efficiency, broader task scalability, and greater tolerance to failure. With the advent of high-fidelity physics simulators, models trained in simulation have demonstrated strong potential for real-world deployment (Wang et al. 2024b; Jiang et al. 2024), thereby fueling growing interest in simulator-based manipulation transfer (Chen et al. 2022b; Qin et al. 2022; Liu et al. 2023; Wang et al. 2023; Zhang et al. 2023; Guzey et al. 2024; Chen et al. 2024; Liu et al. 2024a, 2025). However, replicating human-level manual dexterity remains a significant challenge.

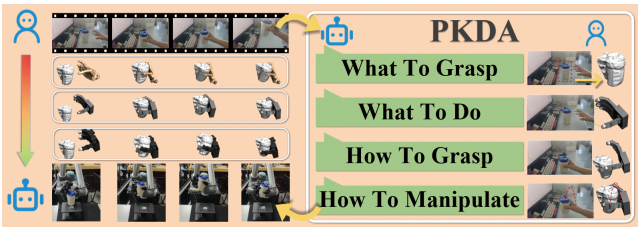


Figure 1: PKDA Overview: Starting from human video demonstrations, human manipulation is mapped to various dexterous hands. The system achieves perception-control integration for dexterous manipulation transfer by addressing four core questions: "what to grasp, what to do, how to grasp, and how to manipulate."

Three key challenges hinder dexterous manipulation transfer: 1) Structural differences between human and robotic hands pose significant challenges for accurate motion retargeting. 2) Complex hand-object contact dynamics impede behavior transfer (Liu et al. 2024a; Pang et al. 2023). 3) The diversity of manipulation tasks limits the generalization of optimization frameworks. (Arunachalam et al. 2023a,b; Park et al. 2025a; Wang et al. 2024a; Qin, Su, and Wang 2022; Handa et al. 2020; Qin et al. 2023; Sivakumar, Shaw, and Pathak 2022) utilize cameras or wearable devices to facilitate the real-time transition from human manipulation to dexterous hand movements with human visual feedback, but is costly and challenging to scale up. Offline transfer techniques, as referenced in (Qin et al. 2022; Chen et al. 2022b; Liu et al. 2023; Li et al. 2024a), only employing kinematics mapping between hands without considering hand-object contact dynamic optimization, fail to

*Corresponding author.

adhere to physical constraints. While reinforcement learning (RL) methods (Rajeswaran et al. 2017; Christen et al. 2021; Mandikal and Grauman 2022; Dasari, Gupta, and Kumar 2023; Guzey et al. 2024; Chen et al. 2024; Zhang et al. 2023) can explore interactions through autonomous trial and error, their inefficient exploration and task-specific reward design limit their generalizing ability.

To this end, we propose Dexterous Manipulation Transfer system via **P**rogressive **K**inematic- **D**ynamic **A**lignment (PKDA) framework, which is designed to be broadly adaptable to different dexterous hands, diverse manipulation tasks, and multiple object categories. As shown in Fig.1, the system only requires human manipulation RGB videos as input, and can automatically configure system parameters to transfer coherent control signals that drive various dexterous hands to complete diverse manipulation tasks.

Our main viewpoints are as follows: 1) The inherent coordination and interaction diversity existing in offline human demonstrations can yield valuable and reliable behavioral references. 2) The synergy of kinematics matching and RL can facilitate balancing the transfer efficiency and quality. Kinematic matching enhances RL by supplying high-quality initialization states and dependable exploration directions. Concurrently, RL investigates effective hand-object interactions under the guidance of kinematic matching and task-irrelevant rewards, forming a transfer path that is both kinematically aligned and dynamically optimized.

Based on the above perspectives, we model the dexterous manipulation transfer problem as a four-stage task: extracting hand-object priors, approaching the object, anthropomorphically grasping the object, and manipulating. Correspondingly, our PKDA framework contains four specially designed modules, as shown in Fig.2. **Interaction Perceptor** extracts hand-object interaction information. **Trajectory Proposer** maps human hand motion trajectories to the dexterous hands and generates primary control signals. **ContactAdapt Optimizer** employing RL, through thumb-guided pre-grasp initialization, action space rescaling, and unified reward, trains a residual policy, ensuring that the optimized trajectories meet both kinematic adaptability and physical interaction stability. **Wrist Trajectory Planner**, adjusts the wrist trajectory using the object trajectory as guidance, ensuring that the dexterous hand manipulation retains the task semantics. It is worth emphasizing that throughout the entire transfer process, it does not require task-specific parameter adjustment, enabling efficient adaptation to various manipulation tasks.

To validate the performance of our dexterous manipulation transfer system at different situations, we conducted extensive experiments in MuJoCo(Todorov, Erez, and Tassa 2012) across a wide variety of manipulation tasks, using three representative robotic hands: Adroit Hand, Allegro Hand, and Leap Hand. The proposed PKDA framework shows significant generalization ability, and achieves higher success rate, faster transfer speed than state-of-the-art methods.

Our contributions are summarized as follows:

- We present a new system for transferring human hand manipulation to multi-fingered dexterous robot hand from human demonstration video. It exhibits excellent transfer stability across different dexterous hand configurations, manipulation tasks, and object categories through leveraging their commonalities. It provides an easily implementable, efficient and scalable technical solution.
- We propose the PKDA manipulation policy learning framework featured by a novel synergistic optimization of kinematics mapping and contact dynamics. Kinematics mapping guides imitation and restricts exploration of RL. Action space rescaling and thumb-guided pre-grasp initialization improve the efficiency of dynamic deviations correction. It shortens the transfer time while also ensures the successful transfer of manipulation capabilities.

## Related Work

### Dexterous Manipulation Transfer

Humans develop adaptive grasping skills through practice, and efficiently transferring these skills to dexterous hands can avoid inefficient learning from scratch, offering a promising path to enhance robotic manipulation (Liu et al. 2020; Kadalagere Sampath et al. 2023). Many studies perform kinematic matching between human and robotic hands by establishing structural correspondence ( finger-wrist vectors (Qin et al. 2022, 2023), interfinger vectors (Handa et al. 2020)). However, due to the absence of contact dynamics optimization, such correspondences often require manual intervention to achieve successful physical interactions. Deep reinforcement learning (DRL) has shown strong capabilities in dynamic tasks like in-hand manipulation (Rajeswaran et al. 2017; OpenAI et al. 2019; Chen, Xu, and Agrawal 2021; Chen et al. 2022a), inspiring DRL-based transfer methods (Christen et al. 2021; Dasari, Gupta, and Kumar 2023; Wang et al. 2023), while effective, they suffer from task-dependent complex rewards design and inefficient exploration in high-dimensional space. Recent work seeks to unify kinematic and dynamic alignment (Chen et al. 2022b; Liu et al. 2023; Guzey et al. 2024; Liu et al. 2025, 2024a; Yin et al. 2025; Chen et al. 2024; Zhao et al. 2024; Lum et al. 2025; Park et al. 2025b; Li et al. 2025). Approaches like (Liu et al. 2023; Chen et al. 2022b) combine retargeting with correlated sampling and local trajectory optimization, but random exploration still risks physically implausible grasps. (Li et al. 2025) employs a two-stage RL pipeline but heavily depends on large, precisely labeled datasets and is vulnerable to noise. Differently, our method provides a new transfer scheme with the constrained RL exploration through multiple designs of action space rescaling, unified reward and thumb-guided pre-grasp initialization, improving efficiency, accuracy, and robust generalization.

### Demonstration Data Collection

Demonstration data has been shown to significantly accelerate manipulation policy learning and improve generalization. Precisely recorded human hand manipulation using multiple cameras or motion capture systems are important

data sources (Chao et al. 2021; Kwon et al. 2021; Fan et al. 2023; Li et al. 2024b), but anatomical and kinematic differences hinder direct transfer to robotic hands. A direct alternative is to collect data using the dexterous robot itself in physical environments (Cheng et al. 2024; Park et al. 2025a; Fang et al. 2023; Zitkovich et al. 2023; Khazatsky et al. 2024), but high hardware costs and complex setups limit scalability. Thanks to advances in high-fidelity simulation and Sim2Real transfer, simulation data has proven effective for real-world tasks (Wang et al. 2024b; Jiang et al. 2024; Handa et al. 2023). This has motivated the use of simulators for data collection, typical representatives including teleoperation (Qin, Su, and Wang 2022; Park et al. 2024) and imitation learning (Wang et al. 2024a). Our method provides an efficient way to generate dexterous manipulation data that aligns with human manipulation in both kinematic and dynamic terms without the need for online human involvement.

## Method

The PKDA pipeline of our system is illustrated in Fig.2, comprising four modules.

### Interaction Perceptor

Interaction Perceptor undertakes the task of collecting the dynamic hand-object interaction information from the raw human manipulation videos, including hand trajectories $H = \{h_1, \ldots, h_t, \ldots, h_T\}$, object trajectories $O = \{o_1, \ldots, o_t, \ldots, o_T\}$, with horizon length $T$, and $N$ contact points $C = \{c_1, c_2, \ldots, c_N\}$. $h_t \in R^{18}$, represents the spatial positions of the fingertips $(15D)$ and the palm orientation $(3D)$. $o_t = \{pos_t, ori_t\} \in R^6$, represents the pose of the object at the time t, where $pos_t$ represents the 3D position of the object's center of mass, and $ori_t$ represents the object's 3D orientation. $c_N \in R^{N\times 3}$, represents the 3D coordinates of the contact points, and $N \in \{2, 3, 4, 5\}$ represents the number of fingertip contacts.

For datasets with known object models (e.g., DexYCB (Chao et al. 2021), TACO (Liu et al. 2024b)), we estimate hand and object pose trajectories using HFL-Net (Lin et al. 2023), designed for monocular RGB images with shared low/high-level features, separate mid-level backbones, and self-/cross-attention to enhance estimation accuracy. Contact points are firstly identified by computing the minimum fingertip-to-object distances and then retained those beneath a 5cm threshold.

For raw videos without ground truth object models, we adopt Hold (Fan et al. 2024) to jointly reconstruct 3D hand-object geometry. To mitigate impacts of object mesh defects caused by occlusion, we apply convex decomposition optimization to perform effective collision detection. To ensure physical plausibility, we also lower the reconstructed object's center of mass to improve stability.

### Trajectory Proposer

The human hand has abilities of precise fingertip position control and adaptive postural adjustment. To mimic this, we employ a retargeting technique that maps a human manipulation trajectory $H$ to a dexterous hand joint angle sequence $Q = \{q_1, \ldots, q_t, \ldots q_T\}$, using fingertip positions and palm orientation as constraints. $q_t \in R^D$ represents joint angles at time $t$, and its dimension $D$ depends on hand type (Appendix A.1 for details).

Existing methods align fingertip-to-wrist vectors between human and robotic hands (Qin et al. 2023, 2022), emphasizing global pose imitation. However, due to hand size discrepancies, this approach introduces noticeable fingertip errors—especially problematic in autonomous manipulation without human feedback.

In our scheme, there is no human visual feedback at all. A decline in fingertip positioning precision alters contact points and contact directions on object surface, risking compromised grasp stability and consequently reducing grasp success rate. To address this, we use the fingertip position vectors in the world coordinate system as the main objective and the palm orientation as an auxiliary constraint for kinematics mapping. This decouples fingertip localization from wrist alignment, reducing sensitivity to morphological differences. Inspired by (Handa et al. 2020), we formulate the mapping as a nonlinear optimization problem that minimizes the position errors between human and robotic fingertips.

The objective function is defined as follows, including fingertip position constraint $E_f$, palm orientation constraint $E_o$, and temporal smoothing constraint$E_s$.

$$\min_{\mathbf{q}_t} \left(w_f E_f + w_o E_o + w_s E_s\right) \tag{1}$$

$$E_f = \sum_{i=1}^{K} \left\| \mathbf{v_i^H}\left(\mathbf{H_t}\right) - \mathbf{v_i^R}\left(\mathbf{q}_t\right) \right\|^2, \tag{2}$$

$$E_o = \mathbf{G}\left(\mathbf{M}_t^H, \mathbf{M}_t^R\right), \tag{3}$$

$$E_s = \left\| \mathbf{q_t} - \mathbf{q_{t-1}} \right\|^2, \tag{4}$$

$\mathbf{v_i^H}$ and $\mathbf{v_i^R}$ in Eq.2 represent the fingertip position vector of the $i$ th finger of the human hand and the corresponding finger of the dexterous hand, $\mathbf{v_i^R}$ is computed by the forward kinematic model of the dexterous hand. $\mathbf{G}$ in Eq.3 is the minimum geodesic distance(Huynh 2009) which is used for calculating the difference between the palm orientation of the human hand $\mathbf{M}_t^H$ and that of the dexterous hand $\mathbf{M}_t^R$. Eq.4 is a temporal smoothing term to penalize the occurrence of large joint angle changes in adjacent action frames. The obtained discrete joint angle sequence $Q$ is converted into the control sequence $A_{primary}$ using an inverse dynamics-based joint angle-control signal conversion method (Qin et al. 2022). For a dexterous hand with K fingers, the corresponding relation of fingers, as well as the definition of the palm orientation are presented in Appendix A.1.

### ContactAdapt Optimizer

Simple motion mapping can imitate grasp postures but fail to convey force closure and dynamic contact strategies, reducing grasp stability under disturbances. To address this, we use RL to optimize grasp dynamics, enabling human-like contact adaptation. However, task diversity complicates reward design, and large action spaces hinder policy efficiency. We propose: 1) RL-Configurator, a cross-task feature

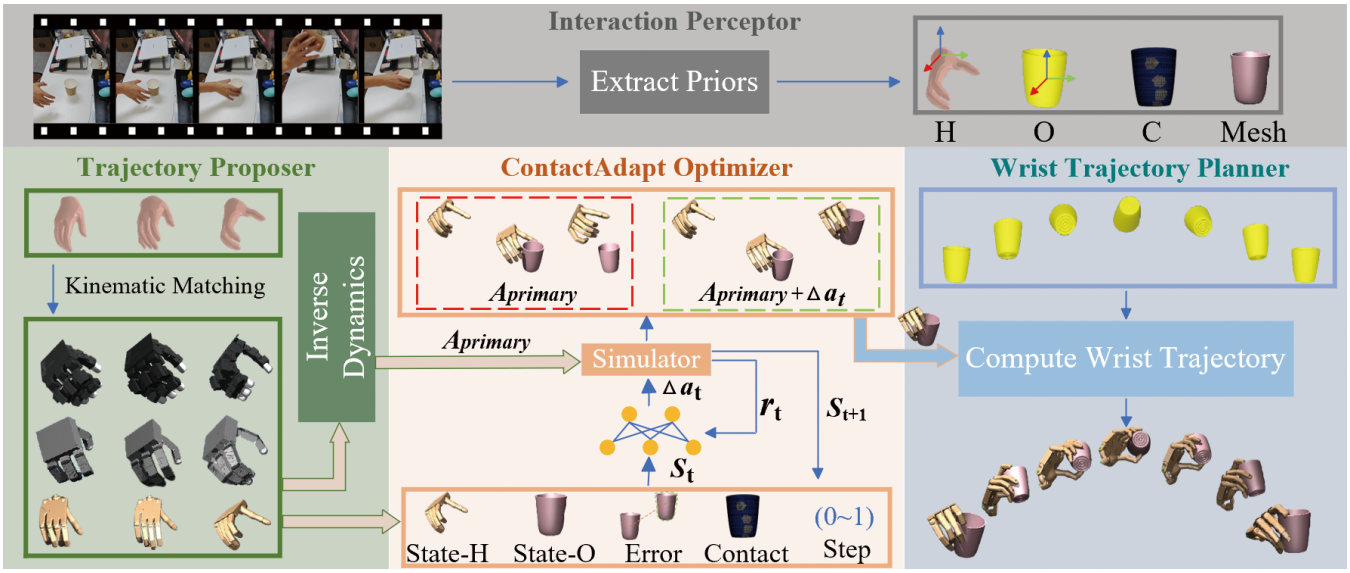


Figure 2: The PKDA system comprises four modules. Interaction Perceptor extracts key manipulation cues, including human hand posture H, object pose O, contact point C, and object mesh, from human demonstration videos. Trajectory Proposer retargets human hand movements into dexterous hand joint angle sequences, generating a primary control signal $A_{primary}$ to guide the primary trajectory. However, lacking dynamic adjustment, this trajectory often leads to failed grasps (see red box). To improve grasp stability, the ContactAdapt Optimizer employs RL, where a residual policy modifies $A_{primary}$ through $\Delta a_t$. Wrist Trajectory Planner combines object motion with relative hand-object constraints to synthesize wrist trajectories and generate the complete control signal.

extractor for unified reward design. 2) action space rescaling to mitigate ineffective exploration.

**RL-Configurator:** In cross-task settings, task specificity poses a significant challenge to the unified RL training paradigm. The core objective of RL-Configurator is extracting common features from object poses, primary trajectories of the dexterous hand, and fingertip contact points across different tasks, thereby establishing a unified RL configuration. As shown in Fig.3 (left), it regulates the configuration of the RL task from the following two main perspectives:

1. Initialize the pre-grasp state for RL based on $A_{primary}$. Considering the thumb's dominant role in the whole grasping process, making it approach the optimal contact position at the early stage of the interaction will significantly reduce the difficulty of policy exploration. Therefore, based on the real-time collision detection and the position detection of thumb tip, we choose the state where the hand-object are not in contact and the thumb tip is closest to the corresponding grasp point as the initial state, and we call this state as the pre-grasp state$(\hat{q}_{pre}, \dot{q}_{pre})$, where $\hat{q}_{pre}$ and $\dot{q}_{pre}$ represent the joint angle and joint velocity.

2. Set the goal for RL. Since most manipulations rely on grasping objects as a prerequisite, we abstract the initial phase of all manipulation tasks as pick-up. The objective is to bring the object to the target pose $o_{target}$ (the pose where the object deviates from its initial position by 0.1m for the first time in its trajectory).

**Action Space Rescaling:** The action space is delineated as $a \in R^D$, where the preceding six dimensions control wrist translation and rotation, while the remaining dimensions control the finger posture of the dexterous hand. To address the distinct learning characteristics and task requirements of wrist and finger movements, we employ action space rescaling during the RL process, as shown in Fig.3 (right). We compress the wrist joint movement range from the global workspace to a local neighborhood $\mathcal{N}(\hat{q}_{pre}, \rho)$ centered around the wrist component of the pre-grasp joint angle $\hat{q}_{pre}$, while maintaining the full range of motion for the finger joints. This constraint not only encourages diverse interaction attempts between the fingers and objects, reducing ineffective exploration of the policy, but also prevents overshooting behaviors caused by excessive wrist motor signal.

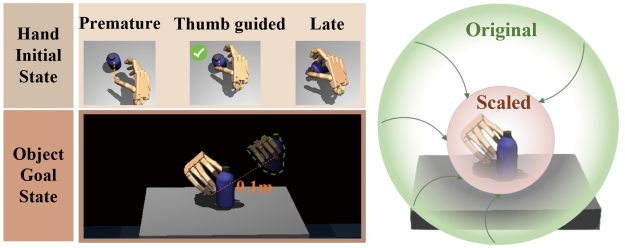


Figure 3: RL-Configurator (left) standardizes diverse tasks by configuring pre-grasp and object goal state. The Action Space Rescaling module (right) compresses wrist motion space into the neighborhood of the pre-grasp to enable efficient hand-object interaction exploration.

**Reward Design:** To effectively train grasping policies across diverse object types, we implement a hierarchical and unified reward with three key components: *1) Approach reward* ($r_{approach}$): It guides fingertips toward target contact points, providing positive feedback when fingertips approach their designated positions. *2) Grasp reward* ($r_{grasp}$): Activating when all fingertips are within the contact tolerance threshold ($\varepsilon = 0.06m$). It incorporates both contact reward $r_{con}$ and imitation reward $r_{sim}$. The contact reward encourages multi-point contact between the fingertips and the object through collision detection. The imitation reward promotes hand posture imitation by computing the cosine similarity between the current joint angles and the target joint angles obtained through retargeting. *3) Lifting reward* ($r_{lift}$): Activating when the thumb and one other finger make contact with the object. It guides the dexterous hand to maneuver the object to the target pose. The complete reward function combines these components (Appendix A.3 for more details). The RL training of the grasping policy is presented in Appendix A.4.

## Wrist Trajectory Planner

For dexterous manipulation transfer tasks, maintaining the consistency of action intentions before and after transfer is an important requirement. Taking the task of drinking water with a cup in hand as an example, the most important requirement of the manipulation transfer is that the dexterous hand can hold the cup to realize the dynamic process of "rise-tilt (drink) -put down". In order to meet such basic requirements, we designed a wrist trajectory planner guided by the dynamic pose change of the object. The interaction between the dexterous hand and the object is modeled as a low dynamic manipulation, that is, there is no relative sliding after the hand picks up the object. We extract the trajectory of the object in the manipulation stage $\{o_{grasp}, \ldots, o_t, \ldots, o_T\}$, and the pose of the wrist of the dexterous hand in the stable grasping $T_{grasp}$, where $o_{grasp}$ refers to the object pose corresponding to the end time step of RL. According to the computed wrist trajectory $T_t = o_t \cdot \left(T_{grasp}^{-1} \cdot o_{grasp}\right)^{-1}$ at each time step, the PD controller is used to control the motion of the dexterous wrist.

# Experiments

## Experiment Settings

To fully test our approach, we constructed test datasets from the following multiple scenarios.

- *Full-Information Scenario:* It enables access to accurate hand poses, object poses, and hand-object contact annotations. We manually selected about 600 trajectories involving single-hand interactions from GRAB dataset (Taheri et al. 2020) for evaluation.
- *Model-Known Visual Scenario:* Hand and object poses, as well as contact points, are estimated from monocular RGB videos, assuming the object's 3D model is known. 10 right-hand manipulations involving geometrically diverse objects were selected from each dataset ( DexYCB (Chao et al. 2021) and TACO (Liu et al. 2024b)) for evaluation.
- *Model-Unknown Visual Scenario:* Only RGB videos are available. This scenario reflects more actual situations. Videos are shot with our own color camera, featuring 5 diverse manipulation tasks using common household objects.

To quantitatively evaluate the performance of our method against the baseline, we adopt four key metrics: *1) SR Grasp(%)*: Grasping success rate. Successful grasping means the object can be held within 0.05m of the target position. *2) SR Follow(%)*: Following success rate. Successful following means dexterous hand can hold the object stably without dropping it throughout the entire manipulation procedure. *3) Er(°)*: Deviation of object rotation, computed by geodesic distance $G$, $Er = \frac{1}{T}\sum_{i=1}^{T} \mathbf{G}(ori_i, \hat{ori_i})$. *4) Ep(m)*: Deviation of object translation, computed by Euclidean distance $d(x, y) = \|\mathbf{x} - \mathbf{y}\|_2$, $Ep = \frac{1}{T}\sum_{i=1}^{T} d(pos_i, \hat{pos_i})$.

Furthermore, we introduce a new metric, *Transfer Success Rate (TSR(%))*, to assess intention-level consistency (Appendix B.1 for details).

## Comparative Experiments

We use the following methods as the baselines for comparison:

*1. Anyteleop* (Qin et al. 2023): Anyteleop achieves the alignment of the human hand and the dexterous hand through the hand pose retargeting.

*2. PGDM* (Dasari, Gupta, and Kumar 2023): PGDM uses pre-grasp poses from multiple sources (motion capture, expert teleoperation,manual annotations) to initialize the RL environment and designs rewards based on reference trajectory reproduction for precise tracking. In the comparative experiment, the manipulation trajectory is directly generated utilizing the pre-trained policy provided by the author.

*3. D-Grasp* (Christen et al. 2021): D-Grasp generates physically plausible dynamic interactions using single-frame grasping pose as reference and specially designed rewards. we re-implemented D-Grasp's reward design in MuJoCo (originally RaiSim) to eliminate the potential deviation caused by different simulator. Implementation details are in Appendix B.2.

Since the implementation of the PGDM method relies on the pre-extracted pose provided by the authors, which is only available on the TCDM benchmark they provided, we only evaluated the subset of tasks where the TCDM benchmark overlaps with our selected data from the GRAB dataset, totaling 40 manipulation tasks. The task list is provided in Appendix B.3.

As shown in Table 1, our method outperforms baselines on SR Grasp, SR Follow and TSR. Tracking accuracy is better than D-Grasp but lower than that of PGDM method. PGDM takes the object trajectory as a strong constraint, sacrificing transfer efficiency to exactly reproduce trajectories. We argue that prioritizing transfer of manipulation action intentions should be more reasonable in terms of practical goal and challenges. Hence we optimize only the grasp phase with RL and use a PD controller for the remaining motion. This yields markedly higher transfer efficiency (Fig.4). In

| | SR Grasp ↑ | SR Follow ↑ | Ep ↓ | Er ↓ | TSR↑ |
|---|---|---|---|---|---|
| *Anyteleop* | 12.5% | 7.5% | N/A | N/A | 7.5% |
| *PGDM* | 72.5% | 72.5% | 0.005 | 18.2 | 72.5% |
| *D-Grasp* | 62.5% | 60% | 0.067 | 35.5 | 57.5% |
| *PKDA-P* | 80% | 80% | 0.058 | 31.5 | 77.5% |
| *PKDA-F* | 84.2% | 77.6% | 0.060 | 34.8 | 73.3% |

Table 1: Results of different methods under the full information scenario using the Adroit Hand. PKDA-P represents the results tested on TCDM Task(40 sequences), and PKDA-F represents the results on GRAB (600 sequences).

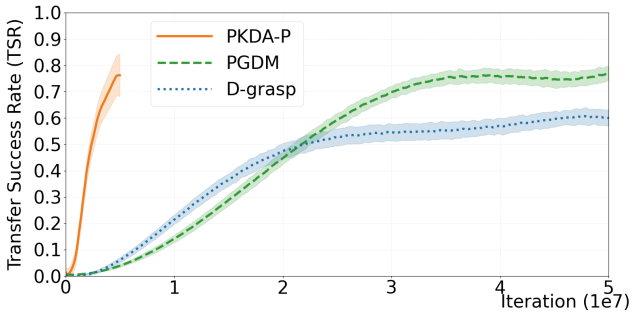


Figure 4: Learning efficiency comparison on the TCDM task using the Adroit Hand.

addition, PGDM only transfers a partial trajectory starting from the pre-grasp, whereas our method transfers a complete manipulation sequence, including approaching the object and putting the object back at the end of task. The Retarget-only method Anyteleop struggles to resist inertial disturbances, often resulting in "lifting and slipping down", it occasionally succeeds with objects like goblets due to favorable lifting postures, highlighting the need for dynamic optimization. D-Grasp, guided by static poses, excels at relocation but falters in repetitive tasks like knocking a nail (Fig.5). Overall, our approach offers both higher learning efficiency and superior transfer success.

## Robustness Evaluation

To assess robustness under non-ideal conditions with visual perception errors, we conducted experiments in both Model-Known and Model-Unknown Visual Scenarios. Two common visual perception inaccuracies occur, including pose estimate errors and object reconstruction defects. Fig.6 illustrates these shortcomings: (a) shows that pose estimation error yields unsatisfactory pre-grasp pose; (b) displays object reconstruction defects, where the reconstructed object model retains only a crude shape showing large error in details.

| | SR Grasp↑ | SR Follow↑ | Ep↓ | Er↓ | TSR↑ |
|---|---|---|---|---|---|
| **Model-Known** | 75% | 70% | 0.034 | 22.4 | 70% |
| **Model-Unknown** | 80% | 80% | 0.033 | 34.8 | 80% |

Table 2: Experimental results of PKDA in the scene with perception inaccuracies using the Adroit Hand.

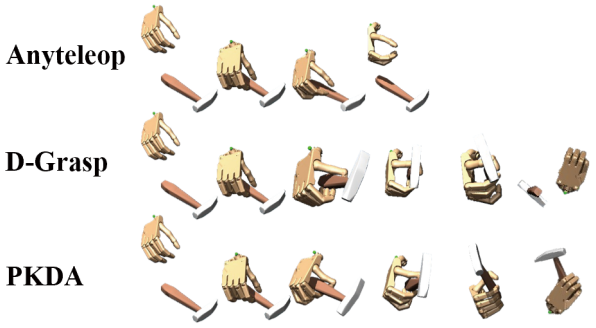


Figure 5: Comparison of the transfer quality on repetitive task of knocking a nail.

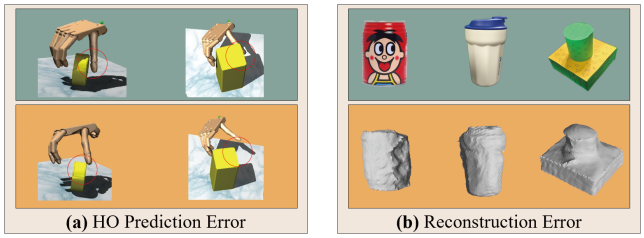


Figure 6: Perception inaccuracies: (a) hand pose (upper: ground truth, lower: estimation result), (b) reconstruction defects (upper: 2D image, lower: 3D mesh).

As shown in Table 2, the success rates are no less than 70% in the two scenarios. The results verify that the PKDA framework enables resistance to interference, so that a robust policy can still be learned under the condition of input uncertainty. The visualization results are presented in Fig.9 in Appendix B.4.

## Experiments on Different Dexterous Hand

To validate PKDA's cross-hand capability, we test three representative dexterous hands, with results in Table 3. Our method achieves effective transfer across diverse hand configurations. Adroit Hand performs best with a 77.5% transfer success rate. Larger hands like Allegro and Leap face challenges on small or slender objects (e.g., hammers), lowering their success rates. Despite significant differences in joint freedom, finger length, and kinematics, position errors (0.054–0.058) and rotation errors (31°–33°) remained consistent, demonstrating steady transfer quality. This consistency stems from preserving hand structural decoupling: fingertip mapping relies on spatial geometry rather than topology, and wrist trajectories are driven by relative hand-object relations. Our method prioritizes functional alignment over hardware specifics, enabling seamless adaptation to different hands by only mapping finger correspondences without adjusting overall parameters. The thumb-guided pre-grasp posture design ensures consistent contact strategies across hands, facilitating subsequent learning. In Appendix B.4, Fig.10 displays pre-grasp postures autonomously generated for Leap, Allegro, and Adroit, and Fig.11 displays manipulation sequence.

| | SR Grasp↑ | SR Follow↑ | Ep↓ | Er↓ | TSR↑ |
|---|---|---|---|---|---|
| Adroit | 80% | 80% | 0.0584 | 31.5 | 77.5% |
| Allegro | 77.5% | 72.5% | 0.0569 | 32.8 | 72.5% |
| Leap | 70% | 67.5% | 0.0544 | 31.7 | 67.5% |

Table 3: Result of PKDA-P on different dexterous hands.

| **Method** | *Finger-wrist Vector* | **Ours** |
|---|---|---|
| TSR↑ | 70% | 77.5% |

Table 4: TSR of PKDA-P with different retargeting methods using the Adroit Hand.

## Ablation Experiments

We construct the following ablation experiments to quantify the contribution of each core component.

*1. Retargeting method:* Aim to investigate the influence of retargeting method on the PKDA. We specifically investigated two retargeting methods: one guided by the finger-wrist vector and the other by the absolute position of the fingertip. The remaining components of the transfer procedure aligned with the PKDA.

*2. Pre-grasp pose selection strategy:* Explore two core aspects for obtaining high-quality pre-grasp poses:

(a) Guidance Condition: PKDA uses the thumb and its corresponding contact point. For comparison, we test index finger and middle finger guidance.

(b) Triggering moment: Under the premise of avoiding collisions between the hand and the object, PKDA selects the triggering moment that reaches the closest finger-to-object distance (Nearest). We also evaluate triggering at preset distance thresholds (0.05/0.1 m).

*3. Action space rescaling:* Performing diversified manipulation tasks with a unified model requires larger workspace, indicating that the dexterous hand has to spend more exploration time to seek for the correct action. To improve exploration efficiency, we propose action space rescaling mechanism in PKDA framework. The ablation experiment will validate its necessity.

Ablation results for the retargeting method (Table 4) show that using fingertip-wrist vectors reduces TSR by 7.5%. Table 5 reveals that index/middle finger guidance reduces TSR by 7.5%/10% versus thumb guidance under Nearest conditions. When using thumb guidance, increasing preset distance thresholds lowers TSR (e.g., dropping to 67.5% at 0.1m vs. Nearest). Crucially, action space rescaling has the most significant influence: Taking the "thumb-guided", "Nearest" triggering moment as an example, removing this mechanism resulted in obvious performance degradation, with the TSR falling to 37.5%. Observations during experiments reveal that without action space rescaling, the dexterous hand exhibites unintended overshooting behavior during grasping. These results validate that rescaling is essential for suppressing detrimental action space interference during policy learning.

| | TG+R | IG+R | MG+R | TG | IG | MG |
|---|---|---|---|---|---|---|
| Nearest | 77.5% | 70% | 67.5% | 37.5% | 32.5% | 25% |
| 0.05m | 72.5% | 70% | 65% | 42.5% | 42.5% | 30% |
| 0.1m | 67.5% | 65% | 67.5% | 45% | 40% | 30% |

Table 5: TSR of PKDA-P under different pre-grasp selection strategies, with and without action space rescaling. TG, IG, and MG represent Thumb-Guided, Index-Guided, and Middle-Guided strategies, respectively. "+R" denotes the use of action space rescaling. All experiments are conducted using the Adroit Hand.

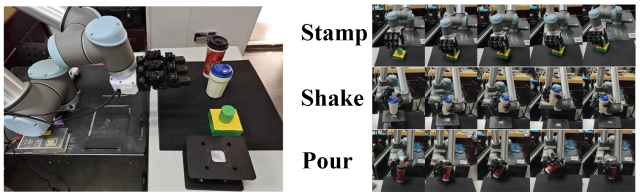


Figure 7: Real-world experiment. The left image shows the experimental scene configuration, while the right image displays the dexterous manipulation sequence deployed on the hardware.

## Real-World Experiments

To validate the feasibility of our generated manipulation trajectories, we conduct real-world experiments on a UR10 robot arm equipped with a Leap Hand (Fig.7), performing three representative tasks—Shake, Pour, and Stamp—using daily objects. The trajectories generated in simulation are directly executed in the real world via open-loop control, successfully reproducing the intended manipulations. Implementation details are provided in Appendix C.

## Summary and Limitations

We propose PKDA, a novel framework for transferring dexterous manipulation from human demonstration videos. Our method extracts reference trajectories from human videos using pose estimation and object reconstruction models, then maps them to dexterous hand trajectories via kinematic matching. To optimize the dynamics of hand-object contact, we train a residual policy with RL guided by a unified reward design. Finally, wrist trajectories are derived inversely from object motion trajectories to ensure the preservation of complete manipulation intentions. Systematic experiments across diverse dexterous hand platforms and manipulation scenarios validate PKDA's effectiveness. Results demonstrate that our method outperforms baseline models in both task success rate and learning efficiency.

Limitations: Although PKDA efficiently transfers most manipulation sequences without requiring large-scale pre-training, it mainly copes with relatively stable hand-object contact patterns. Future work will focus on exploring dexterous manipulation with frequent changes in multiple contact points.

## Acknowledgments

This work was supported by the National Nature Science Foundation of China.(62373075,61873046)

## References

Arunachalam, S. P.; Güzey, I.; Chintala, S.; and Pinto, L. 2023a. Holo-dex: Teaching dexterity with immersive mixed reality. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 5962–5969. IEEE.

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

Chao, Y.-W.; Yang, W.; Xiang, Y.; Molchanov, P.; Handa, A.; Tremblay, J.; Narang, Y. S.; Van Wyk, K.; Iqbal, U.; Birchfield, S.; et al. 2021. DexYCB: A benchmark for capturing hand grasping of objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9044–9053.

Chen, T.; Tippur, M. H.; Wu, S.; Kumar, V.; Adelson, E. H.; and Agrawal, P. 2022a. Visual dexterity: In-hand reorientation of novel and complex object shapes. *Science Robotics*, 8.

Chen, T.; Xu, J.; and Agrawal, P. 2021. A System for General In-Hand Object Re-Orientation. *ArXiv*, abs/2111.03043.

Chen, Y.; Wang, C.; Yang, Y.; and Liu, C. K. 2024. Object-centric dexterous manipulation from human motion data. *arXiv preprint arXiv:2411.04005*.

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

Cheng, X.; Li, J.; Yang, S.; Yang, G.; and Wang, X. 2024. Open-TeleVision: Teleoperation with Immersive Active Visual Feedback. *CoRR*.

Christen, S. J.; Kocabas, M.; Aksan, E.; Hwangbo, J.; Song, J.; and Hilliges, O. 2021. D-Grasp: Physically Plausible Dynamic Grasp Synthesis for Hand-Object Interactions. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 20545–20554.

Dasari, S.; Gupta, A.; and Kumar, V. 2023. Learning dexterous manipulation from exemplar object trajectories and pregrasps. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 3889–3896. IEEE.

Fan, Z.; Parelli, M.; Kadoglou, M. E.; Chen, X.; Kocabas, M.; Black, M. J.; and Hilliges, O. 2024. Hold: Category-agnostic 3d reconstruction of interacting hands and objects from video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 494–504.

Fan, Z.; Taheri, O.; Tzionas, D.; Kocabas, M.; Kaufmann, M.; Black, M. J.; and Hilliges, O. 2023. ARCTIC: A dataset for dexterous bimanual hand-object manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12943–12954.

Fang, H.-S.; Fang, H.; Tang, Z.; Liu, J.; Wang, C.; Wang, J.; Zhu, H.; and Lu, C. 2023. Rh20t: A comprehensive robotic dataset for learning diverse skills in one-shot. *arXiv preprint arXiv:2307.00595*.

Guzey, I.; Dai, Y.; Savva, G.; Bhirangi, R.; and Pinto, L. 2024. Bridging the Human to Robot Dexterity Gap through Object-Oriented Rewards. *arXiv preprint arXiv:2410.23289*.

Handa, A.; Allshire, A.; Makoviychuk, V.; Petrenko, A.; Singh, R.; Liu, J.; Makoviichuk, D.; Van Wyk, K.; Zhurkevich, A.; Sundaralingam, B.; et al. 2023. Dextreme: Transfer of agile in-hand manipulation from simulation to reality. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 5977–5984. IEEE.

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

Huynh, D. Q. 2009. Metrics for 3D rotations: Comparison and analysis. *Journal of Mathematical Imaging and Vision*, 35: 155–164.

Jiang, Z.; Xie, Y.; Lin, K.; Xu, Z.; Wan, W.; Mandlekar, A.; Fan, L.; and Zhu, Y. 2024. Dexmimicgen: Automated data generation for bimanual dexterous manipulation via imitation learning. *arXiv preprint arXiv:2410.24185*.

Kadalagere Sampath, S.; Wang, N.; Wu, H.; and Yang, C. 2023. Review on human-like robot manipulation using dexterous hands. *Cognitive Computation and Systems*, 5: 14–29.

Khazatsky, A.; Pertsch, K.; Nair, S.; Balakrishna, A.; Dasari, S.; Karamcheti, S.; Nasiriany, S.; Srirama, M. K.; Chen, L. Y.; Ellis, K.; et al. 2024. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*.

Kwon, T.; Tekin, B.; Stühmer, J.; Bogo, F.; and Pollefeys, M. 2021. H2o: Two hands manipulating objects for first person interaction recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, 10138–10148.

Li, J.; Zhu, Y.; Xie, Y.; Jiang, Z.; Seo, M.; Pavlakos, G.; and Zhu, Y. 2024a. OKAMI: Teaching Humanoid Robots Manipulation Skills through Single Video Imitation. *ArXiv*, abs/2410.11792.

Li, K.; Li, P.; Liu, T.; Li, Y.; and Huang, S. 2025. Maniptrans: Efficient dexterous bimanual manipulation transfer via residual learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 6991–7003.

Li, K.; Yang, L.; Lin, Z.; Xu, J.; Zhan, X.; Zhao, Y.; Zhu, P.; Kang, W.; Wu, K.; and Lu, C. 2024b. FAVOR: Full-body ar-driven virtual object rearrangement guided by instruction text. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 3136–3144.

Lin, Z.; Ding, C.; Yao, H.; Kuang, Z.; and Huang, S. 2023. Harmonious Feature Learning for Interactive Hand-Object Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 12989–12998.

Liu, Q.; Cui, Y.; Ye, Q.; Sun, Z.; Li, H.; Li, G.; Shao, L.; and Chen, J. 2023. Dexrepnet: Learning dexterous robotic grasping network with geometric and spatial hand-object representations. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3153–3160. IEEE.

Liu, X.; Adalibieke, J.; Han, Q.; Qin, Y.; and Yi, L. 2025. DexTrack: Towards Generalizable Neural Tracking Control for Dexterous Manipulation from Human References. *arXiv preprint arXiv:2502.09614*.

Liu, X.; Lyu, K.; Zhang, J.; Du, T.; and Yi, L. 2024a. Parameterized quasi-physical simulators for dexterous manipulations transfer. In *European Conference on Computer Vision*, 164–182. Springer.

Liu, Y.; Li, Z.; Liu, H.; and Kan, Z. 2020. Skill transfer learning for autonomous robots and human–robot cooperation: A survey. *Robotics and Autonomous Systems*, 128: 103515.

Liu, Y.; Yang, H.; Si, X.; Liu, L.; Li, Z.; Zhang, Y.; Liu, Y.; and Yi, L. 2024b. Taco: Benchmarking generalizable bimanual tool-action-object understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21740–21751.

Lum, T. G. W.; Lee, O. Y.; Liu, C. K.; and Bohg, J. 2025. Crossing the Human-Robot Embodiment Gap with Sim-to-Real RL using One Human Demonstration. *arXiv preprint arXiv:2504.12609*.

Mandikal, P.; and Grauman, K. 2022. Dexvip: Learning dexterous grasping with human hand pose priors from video. In *Conference on Robot Learning*, 651–661. PMLR.

OpenAI; Akkaya, I.; Andrychowicz, M.; Chociej, M.; teusz Litwin, M.; McGrew, B.; Petron, A.; Paino, A.; Plappert, M.; Powell, G.; Ribas, R.; Schneider, J.; Tezak, N. A.; Tworek, J.; Welinder, P.; Weng, L.; Yuan, Q.; Zaremba, W.; and Zhang, L. M. 2019. Solving Rubik's Cube with a Robot Hand. *ArXiv*, abs/1910.07113.

Pang, T.; Suh, H. T.; Yang, L.; and Tedrake, R. 2023. Global planning for contact-rich manipulation via local smoothing of quasi-dynamic contact models. *IEEE Transactions on robotics*, 39: 4691–4711.

Park, S.; Lee, S.; Choi, M.; Lee, J.; Kim, J.; Kim, J.; and Joo, H. 2025a. Learning to Transfer Human Hand Skills for Robot Manipulations. *ArXiv*, abs/2501.04169.

Park, S.; Lee, S.; Choi, M.; Lee, J.; Kim, J.; Kim, J.; and Joo, H. 2025b. Learning to Transfer Human Hand Skills for Robot Manipulations. *arXiv preprint arXiv:2501.04169*.

Park, Y.; Bhatia, J. S.; Ankile, L.; and Agrawal, P. 2024. Dexhub and dart: Towards internet scale robot data collection. *arXiv preprint arXiv:2411.02214*.

Qin, Y.; Su, H.; and Wang, X. 2022. From one hand to multiple hands: Imitation learning for dexterous manipulation from single-camera teleoperation. *IEEE Robotics and Automation Letters*, 7: 10873–10881.

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

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

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

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

Taheri, O.; Ghorbani, N.; Black, M. J.; and Tzionas, D. 2020. GRAB: A Dataset of Whole-Body Human Grasping of Objects. In *European Conference on Computer Vision*.

Todorov, E.; Erez, T.; and Tassa, Y. 2012. MuJoCo: A physics engine for model-based control. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 5026–5033.

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

Wang, J.; Qin, Y.; Kuang, K.; Korkmaz, Y.; Gurumoorthy, A.; Su, H.; and Wang, X. 2024b. Cyberdemo: Augmenting simulated human demonstration for real-world dexterous manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17952–17963.

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

Yin, Z.-H.; Wang, C.; Pineda, L.; Hogan, F.; Bodduluri, K.; Sharma, A.; Lancaster, P.; Prasad, I.; Kalakrishnan, M.; Malik, J.; et al. 2025. DexterityGen: Foundation Controller for Unprecedented Dexterity. *arXiv preprint arXiv:2502.04307*.

Zhang, Y.; Clegg, A.; Ha, S.; Turk, G.; and Ye, Y. 2023. Learning to Transfer In-Hand Manipulations Using a Greedy Shape Curriculum. In *Computer graphics forum*, volume 42, 25–36. Wiley Online Library.

Zhao, S.; Zhu, X.; Chen, Y.; Li, C.; Zhang, X.; Ding, M.; and Tomizuka, M. 2024. DexH2R: Task-oriented Dexterous Manipulation from Human to Robots. *arXiv preprint arXiv:2411.04428*.

Zitkovich, B.; Yu, T.; Xu, S.; Xu, P.; Xiao, T.; Xia, F.; Wu, J.; Wohlhart, P.; Welker, S.; Wahid, A.; et al. 2023. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, 2165–2183. PMLR.