2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

SynH2R: Synthesizing Hand-Object Motions for Learning Human-to-Robot Handovers

Sammy Christen[1*], Lan Feng[1*], Wei Yang[2], Yu-Wei Chao[2], Otmar Hilliges[1], Jie Song[1]

***Abstract*— Vision-based human-to-robot handover is an important and challenging task in human-robot interaction. Recent work has attempted to train robot policies by interacting with dynamic virtual humans in simulated environments, where the policies can later be transferred to the real world. However, a major bottleneck is the reliance on human motion capture data, which is expensive to acquire and difficult to scale to arbitrary objects and human grasping motions. In this paper, we introduce a framework that can generate plausible human grasping motions suitable for training the robot. To achieve this, we propose a hand-object synthesis method that is designed to generate handover-friendly motions similar to humans. This allows us to generate synthetic training and testing data with 100x more objects than previous work. In our experiments, we show that our method trained purely with synthetic data is competitive with state-of-the-art methods that rely on real human motion data both in simulation and on a real system. In addition, we can perform evaluations on a larger scale compared to prior work. With our newly introduced test set, we show that our model can better scale to a large variety of unseen objects and human motions compared to the baselines.**

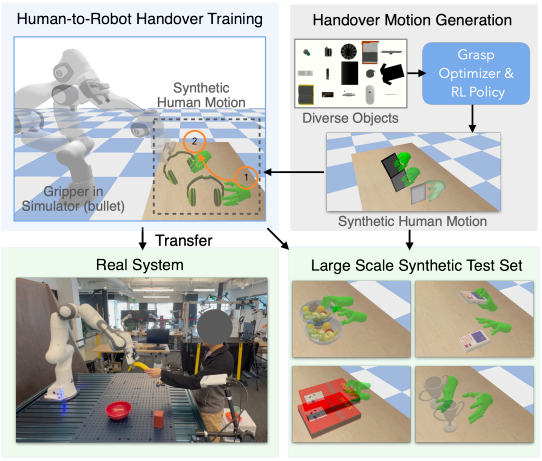

Fig. 1: Overview of our framework. We train a robot to perform human-to-robot handovers using synthetic human motions. We transfer to a real robot and evaluate on a large synthetic test set of unseen objects and human motions.

I. Introduction

Humans handing over objects to robots is a crucial task in human-robot interaction (HRI) [1]. Seamless human-to-robot handovers (H2R) will enable robots to assist humans in many domains, such as manufacturing settings, elderly homes, or rehabilitation. In unknown scenarios, robots will encounter objects and human behavior that they have not previously experienced. Therefore, robots should be flexible in handling unseen objects and human behavior.

Collecting training experiences for robots in the real world is prohibitively inefficient and unsafe for humans. Therefore, recent research on H2R handovers [2], [3] has trained robot policies in simulation by allowing the robot to interact with a simulated human partner, and later transfer the trained policies onto real-world platforms. While this improves the scalability of collecting training experiences for the robot, the pipeline for simulating the human counterpart remains challenging to scale. In order to simulate realistic human motions for handovers, prior work [2], [3], [4], [5] relies on motion capture data of hand-object interactions [6]. The simulated environment for training robots is thus bounded by the object instances and human motions pre-captured in the mocap dataset. To train on novel objects or human motions, a tedious re-capturing of data with a mocap setup is required. This begs the question: can we automatically synthesize human handover motions on arbitrary objects for robot handover training, and thereby fully leverage the blessing of scalability from training in simulation?

Fortunately, recent progress in hand-object interaction synthesis [7], [8] holds the promise to generate natural and physically plausible human grasping motions, which can potentially alleviate the need for expensive motion capture. For example, D-Grasp [8] generates hand motions that grasp an object and move it to a target pose using a reinforcement learning (RL) based policy. Despite their promise, these methods are still not readily applicable for human-to-robot handovers. For example, D-Grasp assumes a grasp pose reference as input and does not account for the handover-friendliness of such a grasp. For successful handovers, it is crucial to control the direction of approaching and the amount of free area for the robot to grasp the object.

In this paper, we combine human-to-robot handover training with hand-object motion synthesis. We build upon D-Grasp [8] and propose a method that can generate natural human grasping motions that are suited for training robots without requiring any high-quality motion capture data. The first question is how to generate grasp references. Current static grasp generation pipelines do not offer controllability with respect to the grasp direction. This makes them unsuitable for handover since humans tend to hand over objects in a direction toward the robot and leave free space on the opposite side for the robot to grasp the object. Besides, we empirically discovered that off-the-shelf learning-based grasp generation models often struggle to generalize to objects

[1]Department of Computer Science, ETH Zurich [2]NVIDIA
[*]Indicates equal contribution

beyond the training distribution. This limits their use on arbitrary object datasets without additional training. To this end, we propose an optimization-based grasp generation method that is conditioned on the approaching direction and incentivizes a stable human hand grasp that does not enclose an object fully. Since our grasp generation pipeline is non-learning based, it also does not suffer from generalization issues on unseen objects. We then generate hand pose references on a large set of objects, and pass them to D-Grasp to generate human grasping and handover motions. To improve the grasping of unseen objects, we also augment D-Grasp to condition on an object shape representation. With this pipeline, our method can synthesize diverse human motions for grasping unseen objects at a larger scale. This in turn allows us to leverage much more diverse human motions and objects in simulation to train the robot.

In our experiments, we first evaluate our approach on the HandoverSim benchmark [2]. We demonstrate that training our method from purely synthetic human motion data can achieve on-par performance with recent work that relies on high-quality motion capture data and uses the test objects during training. Furthermore, we introduce a new synthetic test set of 1174 unseen objects which exceeds the scale of previous benchmarks by 100x (see Tab. I). Our method outperforms the state-of-the-art baselines on this more challenging testbed. Lastly, we show that users do not recognize any significant differences between a policy trained on purely synthetic data versus a policy trained on real motion capture data, indicating the naturalness and plausibility of our generated human motions. This is an important insight that has implications for the training of robotic agents with simulated humans in the future.

To summarize, our contributions are as follows: i) A new framework to scale up human-to-robot handover training by generating large-scale synthetic human handover motions. ii) A method to generate natural human grasping motions that can scale to many objects and allow control of the direction of approach. iii) Experiments in simulation and on a real system showing our method can perform on par with baselines that use high-quality motion capture data for training. iv) A new synthetic test set that allows the evaluation of human-to-robot handovers on more than 1,000 unknown objects. Our evaluations show that our method outperforms baselines on this new benchmark.

II. Related Work

A. Grasp Synthesis for Dexterous Hands

The problem of dexterous grasp synthesis involves determining optimal grasping poses given an object's mesh or point cloud and is generally categorized into two techniques: *non-learning-based* and *learning-based*.

Learning-based methods employ neural networks to predict grasps, leveraging motion capture grasp datasets [9], [10], [11] or synthetic datasets generated by their non-learning-based counterparts [12], [13]. These methods predominantly rely on conditional variational autoencoder architectures, where the resulting grasps are stochastically sampled from latent space. To generalize to unseen shapes, the objects need to be close enough to the training distribution. In our approach, we use a non-learning based solution and hence do not suffer from generalization issues. *Non-learning-based* methods have been employed to generate extensive synthetic datasets [12], [14], [15], [16], [17]. [12] employs collision detection algorithms to formulate stable grasps. In a different vein, some approaches employ differentiable force closure estimation [14], [18] for grasp generation. Works such as [15], [16] exploit a differentiable simulation to synthesize grasps. In contrast to these works, our method can be conditioned on the grasp direction and does not rely on any simulator or force closure estimation.

TABLE I: Comparison between the number of objects and test configurations used in the most related works and ours.

Benchmark	Num Objects	Test Configurations
Sanchez-Matilla et al. [24]	4	288
Rosenberger et al. [25]	13	520
Yang et al. [26]	26	156
HandoverSim ('S0') [2]	18	144
Ours	**1174**	**4436**

Beyond static grasp synthesis, there are works [8], [19], [20], [21], [22], [23] that focus on the temporal aspect of hand-object synthesis. D-Grasp [8] introduces dynamic grasp synthesis to model hand-object interaction sequences, while [20] proposes a universal grasp policy generalizable to diverse objects. These methods utilize grasp reference poses to guide their RL-based policies. In our work, we present an RL policy that can be trained on a small set of YCB objects and generalize to unseen objects at inference time. We achieve this by combining grasp references from our non-learning based grasp optimization with an RL-based policy.

B. Human-to-Robot Handovers

Recent advances in human-to-robot (H2R) handover systems [2], [3], [4], [24], [25], [26], [27], [28] show the potential of creating robust human-to-robot interaction frameworks. This progress has been driven by the surge of hand-object interaction datasets [6], [11], [29], [30], [31], [32], [33], [34], [35], which allows studying H2R handovers as a grasp planning problem [12], [36], [37]. These approaches require the exact knowledge of the 3D object shape, and hence do not generalize to unseen objects. To mitigate this, recent works leverage learning-based grasp predictions from vision input [25], [26], [27], [28], [38]. Rosenberger et al. [25] use hand and object tracking and a grasp selection network to plan H2R handovers, which are executed in open-loop fashion. Yang et al. [26] propose a reactive H2R system that can generalize to unseen objects by selecting temporally consistent 6DoF grasps from GraspNet [39]. In [27], this work is improved by employing an MPC-based algorithm that adds reachability criteria to the motion planning. However, these methods either require the human hand to be stationary, complex hand-designed cost functions, and expertise in robot motion planning. Chao et al. [2] introduce HandoverSim, a benchmark to evaluate handover policies in simulation. GA-DDPG [40] propose a vision-based method

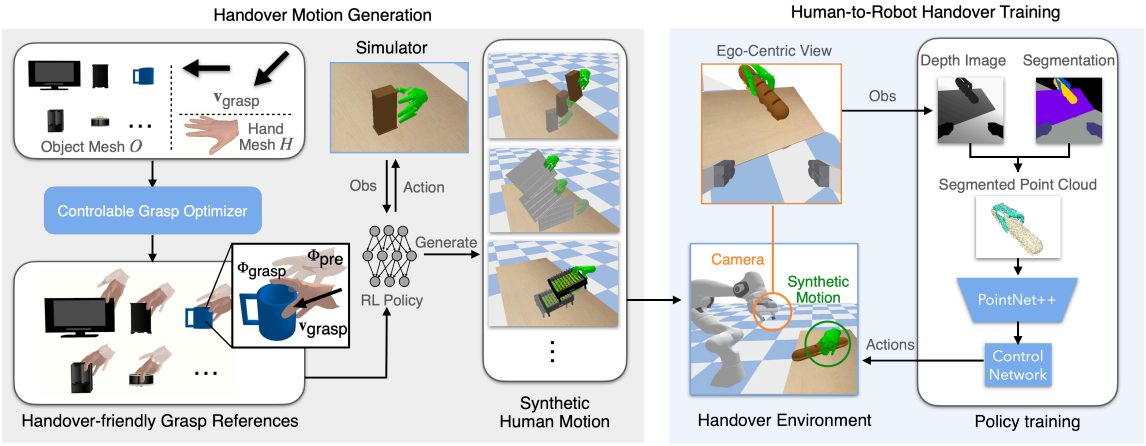

Fig. 2: **Method Overview.** Our framework contains a handover motion generation stage and a H2R handover training stage.

for grasping static objects, which can be deployed for H2R handovers. However, their method has difficulties in dynamic scenes with humans. Closest to our work, Christen et al. [3] propose a framework to learn vision-based handover policies by training with human grasping motions from the DexYCB dataset [6]. In contrast, our work uses synthetic handover motions generated by our method. Therefore, it does not require any real-world mocap data and allows scaling to a much more diverse set of training objects and motions.

III. Overview

The goal of this work is to teach a robot agent to perform human-to-robot handovers by training purely on synthetic human motion data. The simulation setting follows HandoverSim [2] and consists of a tabletop scene with different objects, a robot, and a simulated human hand. The robot comprises a 7-DoF Panda arm with a two-fingered gripper and a wrist-mounted RGB-D camera. The simulated hand replays human handover motions (either from motion capture or synthetic), i.e., grasping an object and moving it to a handover location. The goal of the robot is to grasp the object from the human, without collision or dropping, and move it to a designated goal location. Our framework comprises two stages, as shown in Fig. 2. In the handover motion generation stage (left), we generate synthetic human-object interaction data over a large set of different objects. In the human-to-robot handover training stage, we leverage the synthetic data to train a vision-based human-to-robot handover policy in simulation, which can be transferred to a real system.

IV. Synthetic Handover Motion Generation

To synthesize human handover motions (Fig. 2 left), we first generate handover-friendly static grasp poses and then utilize these grasps as references to guide an RL-based policy inspired by D-Grasp [8] to generate handover motions.

A. Grasp Reference Generation

Our controllable grasp optimizer takes as input an object mesh $\boldsymbol{O}$, a unified human hand mesh $\boldsymbol{H}$ given by the parametric MANO hand model [41], and a grasp direction $\mathbf{v}_{\text{grasp}}$ defined by a three-dimensional vector pointing from the wrist joint to the object center. The MANO hand model [41] is parameterized by a set of pose parameters $\boldsymbol{\theta} \in \mathbb{R}^{45}$, global wrist translation $\boldsymbol{\tau} \in \mathbb{R}^3$ and global wrist orientation $\boldsymbol{\phi} \in \mathbb{R}^3$. The pose parameters comprise 15 joints with 3DoF each. We use the lower dimensional Principal Component Analysis (PCA) space with 15 components to represent the pose parameters $\boldsymbol{\theta}$. As shown in Fig. 2 (left), we introduce a controllable grasp optimizer that outputs two keyframe hand poses—a *pre-grasp pose* $\boldsymbol{\Phi}_{\text{pre}} = [\boldsymbol{\tau}_{\text{pre}}, \boldsymbol{\phi}_{\text{pre}}, \boldsymbol{\theta}_{\text{pre}}]$ and a *grasp reference pose* $\boldsymbol{\Phi}_{\text{grasp}} = [\boldsymbol{\tau}_{\text{grasp}}, \boldsymbol{\phi}_{\text{grasp}}, \boldsymbol{\theta}_{\text{grasp}}]$. We first generate the pre-grasp pose, which is crucial to achieving stable grasping [42], and then optimize the grasp reference pose based on it. Both the pre-grasp pose and the grasp reference pose should adhere to the grasp direction $\mathbf{v}_{\text{grasp}}$.

1) Pre-Grasp Pose $\boldsymbol{\Phi}_{\text{pre}}$. We separately optimize the 6D global pre-grasp pose, which includes the global wrist translation $\boldsymbol{\tau}_{\text{pre}}$ and wrist orientation $\boldsymbol{\phi}_{\text{pre}}$, and the local finger pre-grasp pose $\boldsymbol{\theta}_{\text{pre}}$. As shown in Fig. 3, we define the line connecting the middle fingertip to the thumb tip as the *grasp axis*. Similarly, the vector originating from the wrist joint and pointing towards the midpoint of this connecting line is termed the hand's *heading*.

- **Gripper-like Finger Pose** $\boldsymbol{\theta}_{\text{pre}}$. A two-fingered gripper typically has two adjustable fingers designed to grasp an object firmly from both sides. As illustrated in Fig. 3, this characteristic is emulated by considering the thumb as one finger of the gripper and grouping the other fingers as the second finger. The gripper-like finger pose $\boldsymbol{\theta}_{\text{pre}}$ is derived by maximizing the separation between the thumb tip and the palm plane, i.e., the grasp axis. Specifically, the MANO hand is initialized in the flat hand pose $\boldsymbol{\theta}_{\text{flat}}$ without any rotation or translation. By default, the flat hand is positioned in the xz-plane, with the palm oriented in the -y direction. The y-coordinate of the thumb tip, $\mathbf{p}^{y}_{\text{thumb}}(\boldsymbol{\theta})$, is minimized to maximize its separation from the other fingers, resulting in gripper-like finger poses:

$$\boldsymbol{\theta}_{\text{pre}} = \arg\min_{\boldsymbol{\theta}} \mathbf{p}^{y}_{\text{thumb}}(\boldsymbol{\theta}; \boldsymbol{\theta}_{\text{flat}}). \tag{1}$$

We use the Adam optimizer [43] with a learning rate set at

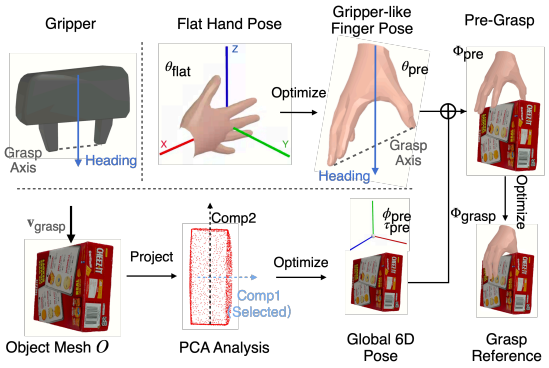

Fig. 3: **Grasp Reference Generation.** Given a grasp direction $\mathbf{v}_{\text{grasp}}$ and an object mesh $\boldsymbol{O}$, our method generates a pre-grasp pose $\boldsymbol{\Phi}_{\text{pre}}$ and grasp reference pose $\boldsymbol{\Phi}_{\text{grasp}}$.

0.003 for 300 iterations. Since we optimize in PCA space, the remaining four fingers converge to a natural pose, even though the objective focuses on the thumb's position.

- **6D Global Pre-Grasp Pose** $(\boldsymbol{\tau}_{\text{pre}}, \boldsymbol{\phi}_{\text{pre}})$**.** We visualize the process of determining the global 6D wrist pre-grasp pose in Fig. 3. We begin by sampling 3,000 points from the object mesh and identify the sampled point furthest from the object center along the grasp direction. The pre-grasp translation $\boldsymbol{\tau}_{\text{pre}}$ is computed as the distance between the object center and the furthest point with an added offset, ensuring there is no collision with the object. To get the pre-grasp global orientation $\boldsymbol{\phi}_{\text{pre}}$, we first align the hand's heading with the grasp direction $\mathbf{v}_{\text{grasp}}$ (cf. Fig. 3). We then rotate the hand to grasp the slimmest part of the object. To this end, we project the object points onto a 2D plane orthogonal to the given grasp direction. We run PCA analysis on the projected 2D point set, which yields two principal components. We choose the component with the lower variance that corresponds to the object's narrowest width. Subsequently, the hand is rotated such that the grasp axis aligns with this narrowest segment.

2) Grasp Reference Pose $\boldsymbol{\Phi}_{\text{grasp}}$. The second part of our optimization generates a grasp reference pose. It takes as input the hand mesh $\boldsymbol{H}$, the object mesh $\boldsymbol{O}$, and the grasp direction $\mathbf{v}_{\text{grasp}}$. We initialize the hand in the pre-grasp pose $\boldsymbol{\Phi}_{\text{pre}}$ from the previous stage. Then, multiple losses are optimized to determine the grasp reference pose $\boldsymbol{\Phi}_{\text{grasp}}$:

$$\begin{aligned} \boldsymbol{\Phi}_{\text{grasp}} &= \arg\min_{\boldsymbol{\Phi}} \mathcal{L}(\boldsymbol{H}(\boldsymbol{\Phi}; \boldsymbol{\Phi}_{\text{pre}}), \boldsymbol{O}, \mathbf{v}_{\text{grasp}}), \\ \mathcal{L} &= \alpha\mathcal{L}_{\text{DP}} + \beta\mathcal{L}_{\text{C}} + \gamma\mathcal{L}_{\text{DF}} + \delta\mathcal{L}_{\text{Ctrl}}. \end{aligned} \tag{2}$$

In the following computations, the object and hand vertices are sampled from the mesh surfaces only. The loss function $\mathcal{L}$ comprises the following components:

- **Dual Penetration Loss** $\mathcal{L}_{\text{DP}}$. We employ a hand-centric penetration loss to minimize penetration during the optimization, similar to [9]. We identifiy the object vertices that are situated inside the hand mesh and compute the penetration loss as the sum of the distances between these vertices and their nearest hand surface vertices:

$$\mathcal{L}_{\text{DP}} = \sum_i \mathcal{L}^2(\mathbf{o}_i^{\text{in}}, \mathbf{h}_i^{\text{closest}}) + \sum_j \mathcal{L}^2(\mathbf{h}_j^{\text{in}}, \mathbf{o}_j^{\text{closest}}), \tag{3}$$

where $\mathbf{o}_i^{\text{in}} \in \boldsymbol{O}$ is the i-th object point inside the hand mesh and $\mathbf{h}_i^{\text{closest}} \in \boldsymbol{H}$ is its closest vertex on the hand surface. While the hand-centric loss mitigates penetration by urging the object vertices toward their closest hand vertices, it is insufficient when entire fingertips are immersed within the object. To address this limitation, we additionally introduce an object-centric penetration loss, implemented in a symmetric manner (second term in Eq. (3)).

- **Contact Loss** $\mathcal{L}_{\text{C}}$. The contact loss ensures that the hand closely approaches and establishes substantial contact with the object, resulting in a stable grasp. It measures the distance between the hand vertices and their closest corresponding object surface vertices:

$$\mathcal{L}_{\text{C}} = \sum_k \mathcal{L}^2(\mathbf{h}_k, \mathbf{o}_k^{\text{closest}}), \tag{4}$$

where $\mathbf{h}_k \in \boldsymbol{H}$ is the k-th hand vertex and $\mathbf{o}_k^{\text{closest}} \in \boldsymbol{O}$ is its closest point on the object mesh.

- **Dynamic Fingertip Loss** $\mathcal{L}_{\text{DF}}$. This loss mimics the human grasping process. It is calculated based on the distance between the thumb tip and the other four fingertips:

$$\mathcal{L}_{\text{DF}} = k \sum_l \mathcal{L}^2(\mathbf{p}_{\text{thumb}}, \mathbf{p}_l). \tag{5}$$

$\mathbf{p}_{\text{thumb}}$ is the 3D joint position of the thumb tip and $\mathbf{p}_l$ represent 3D joint positions of the other four fingertips. k is the dynamic coefficient, which is negative in the early stages of the optimization (step<100) to keep the hand open. In later stages, the coefficient is positive to close the hand towards a stable grasp.

- **Control Loss** $\mathcal{L}_{\text{Ctrl}}$. This loss is designed to ensure that the grasp direction will not deviate from the pre-defined direction during the optimization process. We compute it as the cosine similarity between the wrist vector $\mathbf{v}_{\text{wrist}}$, i.e., the vector pointing from the wrist to the object center, and the grasp direction $\mathbf{v}_{\text{grasp}}$:

$$\mathcal{L}_{\text{Ctrl}} = 1 - \frac{\mathbf{v}_{\text{wrist}} \cdot \mathbf{v}_{\text{grasp}}}{|\mathbf{v}_{\text{wrist}}||\mathbf{v}_{\text{grasp}}|}. \tag{6}$$

The learning rate is initially set to 0.003 and decays by 10% every 100 steps. The entire optimization process takes 500 steps to obtain the final grasping pose. We set the coefficients α, β, γ, δ to 1.5, 3, 0.1, and 1, respectively.

B. Handover Motion Generation

To generate handover motions, we pass the grasp reference pose $\boldsymbol{\Phi}_{\text{grasp}}$ to our improved variant of D-Grasp [8] and initialize the hand in the pre-grasp pose $\boldsymbol{\Phi}_{\text{pre}}$. The D-Grasp model takes as input the grasp reference pose and a target 6D object pose. It then generates human motions that approach, grasp, and bring the object into the target pose. In contrast to vanilla D-Grasp, we augment the observation space with information about the object shape to make it more generalizable to unseen objects. Specifically, we compute the signed-distance information [7] by sampling the object's signed-distance field for each hand joint, which we add to D-Grasp by concatenating it to the original observation space [8].

We generate a training set of grasp reference poses with our optimization on the DexYCB [6] object set, which we use

TABLE II: Benchmark Evaluation on HandoverSim and on the Synthetic Test Set.

			HandoverSim Test Set (DexYCB)				Our Synthetic Test Set			
	Method	Training Data	success (%)	failure (%)			success (%)	failure (%)		
				contact	drop	timeout		contact	drop	timeout
w/ hold	GA-DDPG [40]	ShapeNet [44]	50.00	**4.86**	19.44	25.69	31.22	16.39	43.85	**8.54**
	GA-DDPG finetuned [3]	DexYCB [6]	57.18	6.48	27.08	9.26	40.67	15.70	36.24	7.38
	Christen et. al [3]	DexYCB [6]	**75.23**	9.26	**13.43**	**2.08**	46.57	13.90	28.68	10.85
	Christen et. al [3]	Our Synthetic	71.51	7.87	15.30	5.32	**56.25**	**8.06**	**23.50**	12.17
w/o hold	GA-DDPG [40]	ShapeNet [44]	36.81	9.03	25.00	29.17	23.90	18.26	48.67	9.17
	GA-DDPG finetuned [3]	DexYCB [6]	54.86	**6.71**	26.39	12.04	37.53	12.32	35.03	16.44
	Christen et. al [3]	DexYCB [6]	68.75	8.80	17.82	**4.63**	43.20	10.92	32.84	13.02
	Christen et. al [3]	Our Synthetic	**70.60**	7.18	**16.67**	5.56	**55.94**	**7.53**	**25.89**	**10.63**

as guidance to train D-Grasp. After training the model, we generate grasp reference poses on a larger variety of objects. We synthesize human motions by passing these grasp pose references to the trained D-Grasp model. As our optimization allows control of the approaching direction, we sample grasp directions that are pointing towards the robot. Furthermore, we sample random target object 6D poses within the robot's workspace which serve as handover locations. Lastly, we filter out sequences that fail to grasp the object and reach the target 6D object pose.

V. Augmenting Handover Training

To train the robot, we follow the framework in [3]. Instead of training with trajectories from the DexYCB dataset [6], we simulate the humans in the training environment using our synthetic data. The synthetic human motions are replayed in the simulation during training, following the HandoverSim procedure [2]. Our method takes as input egocentric RGB-D images, from which we compute a segmented point cloud (see Fig. 2). If the hand is occluded by the object, we use the last frame where the hand was visible. We then pass the point cloud through PointNet++ [45] to compute a feature that serves as input to our control policy. The control policy is a neural network that predicts actions that are applied to the robot. Given the updated state, the new point cloud is computed and passed to our policy. The training follows a two-stage procedure. In the pre-training stage, we train in a setting where the human has come to a stop before the robot starts moving. This allows us to leverage expert trajectories from motion and grasp planning [46], which uses ACRONYM [47] to select grasps. To avoid collisions between the robot and the human, we sample grasps that are opposed to the input direction used in the static grasp generation (cf. Section IV-A). In the fine-tuning stage, we train the robot in a setting where the human and robot move simultaneously. Since we cannot use open-loop motion and grasp planning in this setting, we utilize a frozen version of the pre-trained policy as expert [3]. Our control policy is trained in actor-critic fashion using a mix of RL-based, behavior cloning, and auxiliary losses as proposed in [40]. We refer the reader to [3] for more details about the overall training procedure and the definition of the losses.

VI. Experiments

A. Experimental Details

We generate a train and test set of human handover motions using our method on a subset of ShapeNet objects [44]. We adjust the size of the objects based on the dimensions specified in ACRONYM [47]. To eliminate objects that are too large to grasp for the gripper, we exclude those with a minimal width exceeding 0.15m along the grasp direction. Our train set comprises 1175 objects and a total of 2230 right-handed handover motions, whereas our test set contains 1174 objects and 4436 handover motions. The test set also includes left-handed motions, which we generate by mirroring the synthesized right-handed motions. As target object 6DoF handover poses, we randomly sample position offsets from the object's initial position within a range of $[-15, 15]$ cm in x- and y-directions and $[10, 35]$ cm in z-direction. For a fair comparison between training on real motion capture and synthetic motions, we use the same training procedure and hyperparameters from [3] for both variants. We use a single NVIDIA V100 GPU for training and observe similar convergence times in both data settings.

B. Baselines

We experiment with two relevant grasping policies [3], [40]. GA-DDPG [40] is a method for vision-based grasping of rigid objects. Christen et al. [3] is a learning-based method for human-to-robot handovers from point clouds. We use their pre-trained models for evaluation. For [3], we also train on our synthetic data as described in Section V. Furthermore, we include the version of GA-DDPG which was trained in the HandoverSim environment following [3].

C. Metrics

We follow the efficacy metrics in HandoverSim [2]. We report the overall success rate (*success*). A handover is considered a success if the robot grasps the object and moves it to a goal location without dropping or colliding with the human. We distinguish between the three failure cases of human collision (*contact*), object dropping (*drop*), and timeout if the object is not reached (*timeout*). Since we do not focus on improving the efficiency of handovers in this paper, we omit the efficiency metrics from the experiments.

D. Benchmark Evaluation

In this experiment, we compare our framework (Christen et al. [3] trained with synthetic data) against baselines on the HandoverSim [2] test split (i.e., with real human motions from DexYCB [6]). Furthermore, we conduct evaluations on our new synthetic test set to assess generalization to unseen objects and human motions at a larger scale. We report the results in Tab. II and indicate the dataset each model was

TABLE III: Ablation of our framework.

Synthetic Test Set	success (%)	failure (%)		
		contact	drop	timeout
w/ GraspTTA [9]	50.33	**6.12**	31.89	11.64
Ours - 25%	47.23	8.08	25.75	18.93
Ours - 50%	49.21	10.75	27.22	12.80
Ours	**55.94**	7.53	25.89	**10.63**

trained on. We differ between the *w/ hold* setup, where the robot only starts moving once the human has stopped, and the *w/o hold* setup, where the robot and the human move at the same time. Please see our supplementary video for qualitative examples of our method and the baselines.

HandoverSim Our method outperforms the GA-DDPG [40] baselines, and reaches comparable performance with Christen et al. [3] trained on DexYCB, e.g., a success rate of 70.60% for our data and 68.75% for DexYCB data in the *w/o hold* setting. This result is important, as it shows that using purely synthetic human motion data can match the baseline trained on real human motion data. There is a slight drop in performance for synthetic training data compared to DexYCB in the *w/ hold* setting (from 75.23% to 71.51%), which we hypothesize is because the HandoverSim test objects are included in the DexYCB train set, whereas our synthetic data does not contain any of the test objects.

Synthetic Test Set We compare against baselines on the new synthetic test set that includes 1174 unseen objects (Tab. II right). Notably, the success rate of the baselines drops when evaluated on a large set of unseen objects. In contrast, training on our synthetic data has significantly higher success rates in both the *w/ hold* and the *w/o hold* setting (e.g., a 20% relative increase in success rate over the most related baseline [3]). This indicates that our synthetic training set improves generalization to unseen objects. The decrease in success rate for all methods is expected, as the test set includes unseen objects and hence a much wider variety of different shapes. While the human-robot collisions (*contact*) remain relatively low on the synthetic test set, the object drop rate increases the most, e.g., in the *w/o hold* setting from 17.82% on HandoverSim to 32.84% on the synthetic test set for [3] trained on DexYCB. This shows that the methods struggle to find feasible grasps on unseen objects.

E. Ablations

We ablate our synthetic data generation pipeline by comparing it against a variant where we use GraspTTA [9] (*w/ GraspTTA)* instead of our method to generate grasp references for D-Grasp. Furthermore, we analyze the influence of the size of the synthetic train set on the test performance. We report the results on the synthetic test set in Tab. III. We find that a larger training set of synthetic data helps with generalization to unseen objects and human motions, as shown by the decreased performance when only 50% or 25% of the synthetic training set are used for training. Our grasp generator can generate more suitable grasps for handovers than GraspTTA, as shown by the relative increase of 10% in success rate. This implies that the conditioning on the grasping direction is favorable for handover policy training. Note that the ShapeNet objects used are within the training distribution of GraspTTA, as it is trained on Obman [17], and it is likely to perform worse on unseen objects.

F. Sim-to-Real Evaluation

Finally, we transfer the policy trained with our synthetic dataset onto a real robotic platform (system *A*) and compare it with the policy trained on DexYCB from [3] (system *B*). We seek to answer the question: *Can a person differentiate these two systems from interacting with them?* To answer this, we run a human evaluation with 8 participants. For each participant, the experiment consists of two phases. In the first phase, we let the participant hand over three YCB objects, each to both systems once. After each handover, we inform the participant which system is used (*A* or *B*). In the second phase, we use the 10 household objects selected in [26] (see Fig. 6 in [26]) and ask each participant to hand over each object to the robot just once. We randomly sample a system for each object and equally distribute the choices of the two systems (i.e., *A* for 5 objects and *B* for the other 5). In this phase, we do not disclose the chosen system to the participant. After each handover, we ask the participant to make a guess of the chosen system based on the interaction. In the end, we found the two systems both performing competently, exhibiting over 85% handover success rate (45/50 for *A* and 48/50 for *B*). The classification accuracies from the participants are: (8/10, 6/10, 4/10, 10/10, 10/10, 6/10, 6/10, 10/10) (random guessing is expected to get 5/10). Four of them have an accuracy less or equal to 6/10, struggling to tell apart the two systems. Among the other four, two of them answered "felt equal" in a forced choice question between "preferred *A*", "preferred *B*", and "felt equal". They also commented that the systems can be distinguished due to their subtly different tendencies in the approaching direction (the baseline policy tends to have a slight left tilt before handover). This may have resulted from the randomness in training. Overall, this result suggests that our system trained purely on synthetic data is performing closely to a system trained on real data.

VII. Conclusion

We have introduced a framework to generate synthetic human motions for handover training. Our method combines a non-learning based grasp optimizer with an RL-based policy. We have generated a synthetic training set and demonstrated that training with our generated motions reaches a similar performance to training with motion capture data, both in simulation and on a real system. Moreover, we have shown that training with our synthetic data generalizes better to unseen objects on a large-scale synthetic test set. While this work focuses on generalizability to new objects, future work can investigate the generalizability to different human motions or robot morphologies. For example, it is interesting to explore the integration of full-body synthetic humans [22] or more challenging human-robot interactions such as two-handed handovers and articulated objects [23].

REFERENCES

[1] V. Ortenzi, A. Cosgun, T. Pardi, W. P. Chan, E. Croft, and D. Kulić, "Object handovers: A review for robotics," *IEEE Transactions on Robotics (T-RO)*, 2021.

[2] Y.-W. Chao, C. Paxton, Y. Xiang, W. Yang, B. Sundaralingam, T. Chen, A. Murali, M. Cakmak, and D. Fox, "HandoverSim: A simulation framework and benchmark for human-to-robot object handovers," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

[3] S. Christen, W. Yang, C. Pérez-D'Arpino, O. Hilliges, D. Fox, and Y.-W. Chao, "Learning human-to-robot handovers from point clouds," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[4] Y. L. Pang, A. Xompero, C. Oh, and A. Cavallaro, "Towards safe human-to-robot handovers of unknown containers," in *IEEE International Conference on Robot & Human Interactive Communication (RO-MAN)*, 2021.

[5] S. Christen, S. Stevsic, and O. Hilliges, "Guided deep reinforcement learning of control policies for dexterous human-robot interaction," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2019.

[6] Y.-W. Chao, W. Yang, Y. Xiang, P. Molchanov, A. Handa, J. Tremblay, Y. S. Narang, K. Van Wyk, U. Iqbal, S. Birchfield, J. Kautz, and D. Fox, "DexYCB: A benchmark for capturing hand grasping of objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[7] H. Zhang, Y. Ye, T. Shiratori, and T. Komura, "Manipnet: neural manipulation synthesis with a hand-object spatial representation," *ACM Trans. Graph.*, vol. 40, pp. 121:1–121:14, 2021.

[8] S. Christen, M. Kocabas, E. Aksan, J. Hwangbo, J. Song, and O. Hilliges, "D-grasp: Physically plausible dynamic grasp synthesis for hand-object interactions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[9] H. Jiang, S. Liu, J. Wang, and X. Wang, "Hand-object contact consistency reasoning for human grasps generation," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11 087–11 096, 2021.

[10] K. Karunratanakul, J. Yang, Y. Zhang, M. J. Black, K. Muandet, and S. Tang, "Grasping field: Learning implicit representations for human grasps," *2020 International Conference on 3D Vision (3DV)*, pp. 333–344, 2020.

[11] O. Taheri, N. Ghorbani, M. J. Black, and D. Tzionas, "GRAB: A dataset of whole-body human grasping of objects," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[12] A. T. Miller and P. K. Allen, "Graspit! a versatile simulator for robotic grasping," *IEEE Robotics & Automation Magazine*, vol. 11, pp. 110–122, 2004.

[13] Y. Hasson, G. Varol, D. Tzionas, I. Kalevatykh, M. J. Black, I. Laptev, and C. Schmid, "Learning joint reconstruction of hands and manipulated objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[14] R. Wang, J. Zhang, J. Chen, Y. Xu, P. Li, T. Liu, and H. Wang, "Dexgraspnet: A large-scale robotic dexterous grasp dataset for general objects based on simulation," *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 11 359–11 366, 2022.

[15] D. Turpin, L. Wang, E. Heiden, Y.-C. Chen, M. Macklin, S. Tsogkas, S. Dickinson, and A. Garg, "Grasp'd: Differentiable contact-rich grasp synthesis for multi-fingered hands," in *European Conference on Computer Vision*. Springer, 2022, pp. 201–221.

[16] D. Turpin, T. Zhong, S. Zhang, G. Zhu, J. Liu, R. Singh, E. Heiden, M. Macklin, S. Tsogkas, S. Dickinson, *et al.*, "Fast-grasp'd: Dexterous multi-finger grasp generation through differentiable simulation," *arXiv preprint arXiv:2306.08132*, 2023.

[17] Y. Hasson, G. Varol, D. Tzionas, I. Kalevatykh, M. J. Black, I. Laptev, and C. Schmid, "Learning joint reconstruction of hands and manipulated objects," in *CVPR*, 2019.

[18] T. Liu, Z. Liu, Z. Jiao, Y. Zhu, and S.-C. Zhu, "Synthesizing diverse and physically stable grasps with arbitrary hand structures using differentiable force closure estimator," *IEEE Robotics and Automation Letters*, vol. 7, pp. 470–477, 2021.

[19] K. Zhou, B. L. Bhatnagar, J. E. Lenssen, and G. Pons-Moll, "Toch: Spatio-temporal object-to-hand correspondence for motion refinement," in *European Conference on Computer Vision*. Springer, 2022, pp. 1–19.

[20] Y. Xu, W. Wan, J. Zhang, H. Liu, Z. Shan, H. Shen, R. Wang, H. Geng, Y. Weng, J. Chen, T. Liu, L. Yi, and H. Wang, "Unidexgrasp: Universal robotic dexterous grasping via learning diverse proposal generation and goal-conditioned policy," *ArXiv*, vol. abs/2303.00938, 2023.

[21] Y.-H. Wu, J. Wang, and X. Wang, "Learning generalizable dexterous manipulation from human grasp affordance," in *Conference on Robot Learning*, 2022.

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

[23] H. Zhang, S. Christen, Z. Fan, L. Zheng, J. Hwangbo, J. Song, and O. Hilliges, "ArtiGrasp: Physically plausible synthesis of bi-manual dexterous grasping and articulation," in *International Conference on 3D Vision (3DV)*, 2024.

[24] R. Sanchez-Matilla, K. Chatzilygeroudis, A. Modas, N. F. Duarte, A. Xompero, P. Frossard, A. Billard, and A. Cavallaro, "Benchmark for human-to-robot handovers of unseen containers with unknown filling," *IEEE Robotics and Automation Letters (RA-L)*, 2020.

[25] P. Rosenberger, A. Cosgun, R. Newbury, J. Kwan, V. Ortenzi, P. Corke, and M. Grafinger, "Object-independent human-to-robot handovers using real time robotic vision," *IEEE Robotics and Automation Letters (RA-L)*, 2021.

[26] W. Yang, C. Paxton, A. Mousavian, Y.-W. Chao, M. Cakmak, and D. Fox, "Reactive human-to-robot handovers of arbitrary objects," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[27] W. Yang, B. Sundaralingam, C. Paxton, I. Akinola, Y.-W. Chao, M. Cakmak, and D. Fox, "Model predictive control for fluid human-to-robot handovers," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

[28] H. Duan, P. Wang, Y. Li, D. Li, and W. Wei, "Learning human-to-robot dexterous handovers for anthropomorphic hand," *IEEE Transactions on Cognitive and Developmental Systems (TCDS)*, 2022.

[29] Y. Liu, Y. Liu, C. Jiang, K. Lyu, W. Wan, H. Shen, B. Liang, Z. Fu, H. Wang, and L. Yi, "HOI4D: A 4D egocentric dataset for category-level human-object interaction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[30] G. Garcia-Hernando, S. Yuan, S. Baek, and T.-K. Kim, "First-person hand action benchmark with RGB-D videos and 3D hand pose annotations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[31] S. Hampali, M. Rad, M. Oberweger, and V. Lepetit, "HOnnotate: A method for 3D annotation of hand and object poses," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[32] S. Brahmbhatt, C. Tang, C. D. Twigg, C. C. Kemp, and J. Hays, "ContactPose: A dataset of grasps with object contact and hand pose," in *Proceedings of the European Conference on Computer Vision (ECCV)*, August 2020.

[33] G. Moon, S.-I. Yu, H. Wen, T. Shiratori, and K. M. Lee, "InterHand2.6M: A dataset and baseline for 3D interacting hand pose estimation from a single RGB image," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[34] R. Ye, W. Xu, Z. Xue, T. Tang, Y. Wang, and C. Lu, "H2O: A benchmark for visual human-human object handover analysis," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[35] Z. Fan, O. Taheri, D. Tzionas, M. Kocabas, M. Kaufmann, M. J. Black, and O. Hilliges, "ARCTIC: A dataset for dexterous bimanual hand-object manipulation," in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[36] A. Bicchi and V. Kumar, "Robotic grasping and contact: A review," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2000.

[37] J. Bohg, A. Morales, T. Asfour, and D. Kragic, "Data-driven grasp synthesis—a survey," *IEEE Transactions on Robotics (T-RO)*, 2013.

[38] N. Marturi, M. Kopicki, A. Rastegarpanah, V. Rajasekaran, M. Adjigble, R. Stolkin, A. Leonardis, and Y. Bekiroglu, "Dynamic grasp and trajectory planning for moving objects," *Autonomous Robots*, 2019.

[39] A. Mousavian, C. Eppner, and D. Fox, "6-DOF GraspNet: Variational grasp generation for object manipulation," in *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[40] L. Wang, Y. Xiang, W. Yang, A. Mousavian, and D. Fox, "Goal-

auxiliary actor-critic for 6D robotic grasping with point clouds," in *Conference on Robot Learning (CoRL)*, 2021.

[41] J. Romero, D. Tzionas, and M. J. Black, "Embodied hands: Modeling and capturing hands and bodies together," *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, vol. 36, no. 6, Nov. 2017.

[42] 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)*, 2023, pp. 3889–3896.

[43] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, Y. Bengio and Y. LeCun, Eds., 2015.

[44] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, *et al.*, "Shapenet: An information-rich 3d model repository," *arXiv preprint arXiv:1512.03012*, 2015.

[45] C. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," in *NIPS*, 2017.

[46] L. Wang, Y. Xiang, and D. Fox, "Manipulation trajectory optimization with online grasp synthesis and selection," in *Proceedings of Robotics: Science and Systems (RSS)*, 2020.

[47] C. Eppner, A. Mousavian, and D. Fox, "ACRONYM: A large-scale grasp dataset based on simulation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.