UniDexGrasp: Universal Robotic Dexterous Grasping via Learning Diverse Proposal Generation and Goal-Conditioned Policy

Yinzhen Xu*[1,2,3], Weikang Wan*[1,2], Jialiang Zhang*[1,2], Haoran Liu*[1,2], Zikang Shan[1], Hao Shen[1], Ruicheng Wang[1], Haoran Geng[1,2], Yijia Weng[4], Jiayi Chen[1], Tengyu Liu[3], Li Yi[5], He Wang†[1,2]

[1] Center on Frontiers of Computing Studies, Peking University [2] School of EECS, Peking University
[3] Beijing Institute for General AI [4] Stanford University [5] Tsinghua University
https://pku-epic.github.io/UniDexGrasp/

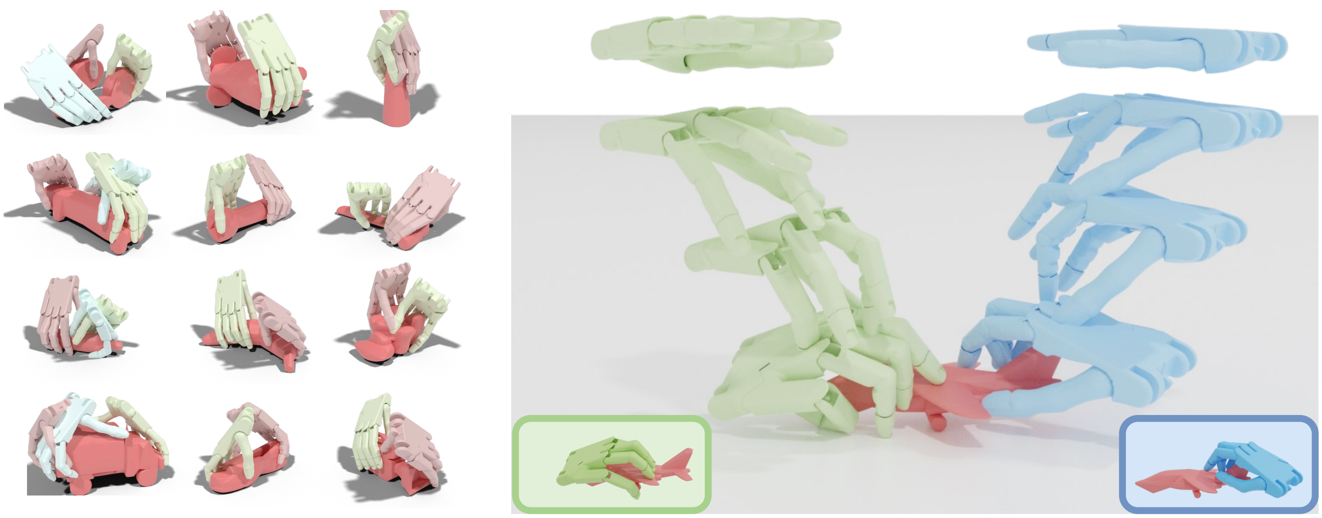


Figure 1. **UniDexGrasp via grasp proposal generation and goal-conditioned execution.** Left (*grasp proposals*): each figure demonstrates two or three diverse and high-quality grasp proposals that vary greatly in rotation, translation, and joint angles; right (*grasp execution*): given a grasp goal pose, our highly generalizable goal-conditioned grasping policy can grasp the object in the way specified by the goal, as shown in the green and blue trajectories and their corresponding goals.

Abstract

In this work, we tackle the problem of learning universal robotic dexterous grasping from a point cloud observation under a table-top setting. The goal is to grasp and lift up objects in high-quality and diverse ways and generalize across hundreds of categories and even the unseen. Inspired by successful pipelines used in parallel gripper grasping, we split the task into two stages: 1) grasp proposal (pose) generation and 2) goal-conditioned grasp execution. For the first stage, we propose a novel probabilistic model of grasp pose conditioned on the point cloud observation that factorizes rotation from translation and articulation. Trained on our synthesized large-scale dexterous grasp dataset, this model enables us to sample diverse and high-quality dexterous grasp poses for the object point cloud. For the second stage, we propose to replace the motion planning used in parallel gripper grasping with a goal-conditioned grasp policy, due to the complexity involved in dexterous grasping execution. Note that it is very challenging to learn this highly generalizable grasp policy that only takes realistic inputs without oracle states. We thus propose several important innovations, including state canonicalization, object curriculum, and teacher-student distillation. Integrating the two stages, our final pipeline becomes the first to achieve universal generalization for dexterous grasping, demonstrating an average success rate of more than 60% on thousands of object instances, which significantly out-

*Equal contribution.
†Corresponding author.

performs all baselines, meanwhile showing only a minimal generalization gap.

1. Introduction

Robotic grasping is a fundamental capability for an agent to interact with the environment and serves as a prerequisite to manipulation, which has been extensively studied for decades. Recent years have witnessed great progress in developing grasping algorithms for parallel grippers [8, 16, 17, 20, 49, 50] that carry high success rate on universally grasping unknown objects. However, one fundamental limitation of parallel grasping is its low dexterity which limits its usage to complex and functional object manipulation.

Dexterous grasping provides a more diverse way to grasp objects and thus is of vital importance to robotics for functional and fine-grained object manipulation [2, 29, 38, 40, 52]. However, the high dimensionality of the actuation space of a dexterous hand is both the advantage that endows it with such versatility and the major cause of the difficulty in executing a successful grasp. As a widely used five-finger robotic dexterous hand, ShadowHand [1] amounts to 26 degrees of freedom (DoF), in contrast with 7 DoF for a typical parallel gripper. Such high dimensionality magnifies the difficulty in both generating valid grasp poses and planning the execution trajectories, and thus distinguishes the dexterous grasping task from its counterpart for parallel grippers. Several works have tackled the grasping pose synthesis problem [6, 28, 33, 49], however, they all assume oracle inputs (full object geometry and states). Very few works [9, 38] tackle dexterous grasping in a realistic robotic setting, but so far no work yet can demonstrate universal and diverse dexterous grasping that can well generalize to unseen objects.

In this work, we tackle this very challenging task: learning universal dexterous grasping skills that can generalize well across hundreds of seen and unseen object categories in a realistic robotic setting and only allow us to access depth observations and robot proprioception information. Our dataset contains more than one million grasps for 5519 object instances from 133 object categories, which is the largest robotic dexterous grasping benchmark to evaluate universal dexterous grasping.

Inspired by the successful pipelines from parallel grippers, we propose to decompose this challenging task into two stages: 1) **dexterous grasp proposal generation**, in which we predict diverse grasp poses given the point cloud observations; and 2) **goal-conditioned grasp execution**, in which we take one grasp goal pose predicted by stage 1 as a condition and generates physically correct motion trajectories that comply with the goal pose. Note that both of these two stages are indeed very challenging, for each of which we contribute several innovations, as explained below.

For dexterous grasp proposal generation, we devise a novel conditional grasp pose generative model that takes point cloud observations and is trained on our synthesized large-scale table-top dataset. Here our approach emphasizes the diversity in grasp pose generation, since the way we humans manipulate objects can vary in many different ways and thus correspond to different grasping poses. Without diversity, it is impossible for the grasping pose generation to comply with the demand of later dexterous manipulation. Previous works [22] leverages CVAE to jointly model hand rotation, translation, and articulations and we observe that such CVAE suffers from severe mode collapse and can't generate diverse grasp poses, owing to its limited expressivity when compared to conditional normalizing flows [12, 13, 15, 23, 35] and conditional diffusion models [5, 43, 46]. However, no works have developed normalizing flows and diffusion models that work for the grasp pose space, which is a Cartesian product of $\mathbf{SO(3)}$ of hand rotation and a Euclidean space of the translation and joint angles. We thus propose to decompose this conditional generative model into two conditional generative models: a conditional rotation generative model, *namely* GraspIPDF, leveraging ImplicitPDF [34] (in short, IPDF) and a conditional normalizing flow, *namely* GraspGlow, leveraging Glow [23]. Combining these two modules, we can sample diverse grasping poses and even select what we need according to language descriptions. The sampled grasps can be further refined to be more physically plausible via ContactNet, as done in [22].

For our grasp execution stage, we learn a goal-conditioned policy that can grasp any object in the way specified by the grasp goal pose and only takes realistic inputs: the point cloud observation and robot proprioception information, as required by real robot experiments. Note that reinforcement learning (RL) algorithms usually have difficulties with learning such a highly generalizable policy, especially when the inputs are visual signals without ground truth states. To tackle this challenge, we leverage a teacher-student learning framework that first learns an oracle teacher model that can access the oracle state inputs and then distill it to a student model that only takes realistic inputs. Even though the teacher policy gains access to oracle information, making it successful in grasping thousands of different objects paired with diverse grasp goals is still formidable for RL. We thus introduce two critical innovations: a canonicalization step that ensures $\mathbf{SO(2)}$ equivariance to ease the policy learning; and an object curriculum that first learns to grasp one object with different goals, then one category, then many categories, and finally all categories.

Extensive experiments demonstrate the remarkable performance of our pipelines. In the grasp proposal generation stage, our pipeline is the only method that exhibits high diversity while maintaining the highest grasping quality. The whole dexterous grasping pipeline, from the vision to policy, again achieves impressive performance in our simulation environment and, for the first time, demonstrates a universal grasping policy with more than 60% success rate and remarkably outperforms all the baselines. We will make the dataset and the code publicly available to facilitate future research.

2. Related Work

Dexterous Grasp Synthesis Dexterous grasp synthesis, a task aiming to generate valid grasping poses given the object mesh or point cloud, falls into two categories. First, non-learning methods serve to generate large synthetic datasets. Among which, *GraspIt!* [33] is a classical tool that synthesizes stable grasps by collision detection, commonly adopted by early works [7, 26, 27]. Recently, an optimization-based method [28] greatly improves grasp diversity with its proposed differentiable force closure estimator, enabling datasets of higher quality [25, 51]. Second, learning-based methods [7,11,21,22,26,44] learn from these datasets to predict grasps with feed-forward pipelines, but struggle to possess quality and diversity at the same time. To tackle this problem, we propose to build a conditional generative model that decouples rotation from translation and articulation.

Probabilistic Modeling on $\mathbf{SO(3)} \times \mathbb{R}^n$ One way to generate diverse grasping poses is to learn the distribution of plausible poses using ground truth poses. GraspTTA [22] uses conditional VAE and suffers from severe model collapse which leads to limited diversity. In contrast, normalizing flow is capable of modeling highly complex distributions as it uses negative log-likelihood (NLL) as loss, suiting our needs for grasping proposal generation. Building normalizing flow (NF) in the Euclidean space has been well studied [12, 13, 23]. But unfortunately, hand pose consists of a $\mathbf{SO(3)}$ part (the rotation) and a $\mathbb{R}^n$ part (the translation and joint angles), and using normalizing flow in $\mathbf{SO(3)}$ is hard due to its special topological structure. Relie [15] and ProHMR [24] both perform NF in the Euclidean space and then map it into $\mathbf{SO(3)}$. As these two spaces are not topologically equivalent, they suffer from discontinuity and infinite-to-one mapping respectively. A more detailed analysis of these phenomena is in Sec. B.1.2 of our supp. In comparison, IPDF [34] uses a neural network to output unnormalized log probability and then normalize it with uniform samples or grids on $\mathbf{SO(3)}$ and also uses NLL as loss. As IPDF is insensitive to topological structure, it is a better choice to model distributions on $\mathbf{SO(3)}$ than existing NFs. Therefore, our grasp proposal module decouples rotation from translation and joint angles and models these distributions separately with IPDF [34] and Glow [23].

Dexterous Grasp Execution Executing a dexterous grasp requires an agent to perform a complete trajectory, rather than a static grasping pose. Previous approaches have used analytical methods [3,4,14] to model hand and object kinematics and dynamics, and then optimized trajectories for robot control. However, these methods typically require simplifications such as using simple finger and object geometries to make planning tractable. More recently, reinforcement and imitation learning techniques have shown promise for dexterous grasping [10,32,40,42,48,52]. However, these methods rely on omniscient knowledge of the object mesh and struggle to handle realistic task settings, making them unsuitable for deployment in the real world. To address this issue, recent works have explored using raw RGB images [31, 32] or 3D point clouds [39] as policy inputs. However, none of these methods have been able to generalize to a large number of objects under raw vision input. In contrast, our goal-conditioned grasp execution method achieves universal generalization on thousands of object instances by leveraging a teacher-student distillation trick, object curriculum learning, and state canonicalization.

3. Method

We propose UniDexGrasp, a two-stage pipeline for generalizable dexterous grasping. We divide the task into two phases: 1) grasp proposal generation (Sec. 3.2), 2) goal-conditioned grasp execution (Sec. 3.3). First, the grasp proposal generation module takes the object point cloud and samples a grasp proposal. Then, the goal-conditioned grasping policy takes this proposal as goal pose, and executes the grasp in the Isaac Gym simulator [30], taking only point cloud observations and robot proprioception as input in every time step t.

3.1. Problem Settings and Method Overview

The grasp proposal generation module, shown on the left part of Fig. 2, takes the object and table cloud $X_0 \in \mathbb{R}^{N\times 3}$ as input, and samples a grasp proposal $\boldsymbol{g} = (R, \boldsymbol{t}, \boldsymbol{q})$ out of a distribution, where $R \in \mathbf{SO(3)}, \boldsymbol{t} \in \mathbb{R}^3, \boldsymbol{q} \in \mathbb{R}^K$ represent the root rotation, root translation, and joint angles of the dexterous hand, and K is the total degree-of-freedom of the hand articulations. The proposed grasp $\boldsymbol{g}$ will be the goal pose of the next module, shown on the right part of Fig. 2.

The final goal-conditioned grasp execution module is a vision-based policy that runs in the IsaacGym [30] physics simulator. In each time step t, the policy takes the goal pose $\boldsymbol{g}$ from the previous module, object and the scene point cloud X_t, and robot proprioception $\boldsymbol{s}^r_t$ as observation, and outputs an action $\boldsymbol{a}_t$. The policy should work across different object categories and even unseen categories. To simplify the problem, we initialize the hand with an initial translation $\boldsymbol{t}_0 = (0, 0, h_0)$ and an initial rotation $R_0 = (\frac{\pi}{2}, 0, \phi_0)$, where h_0 is a fixed height and the hand

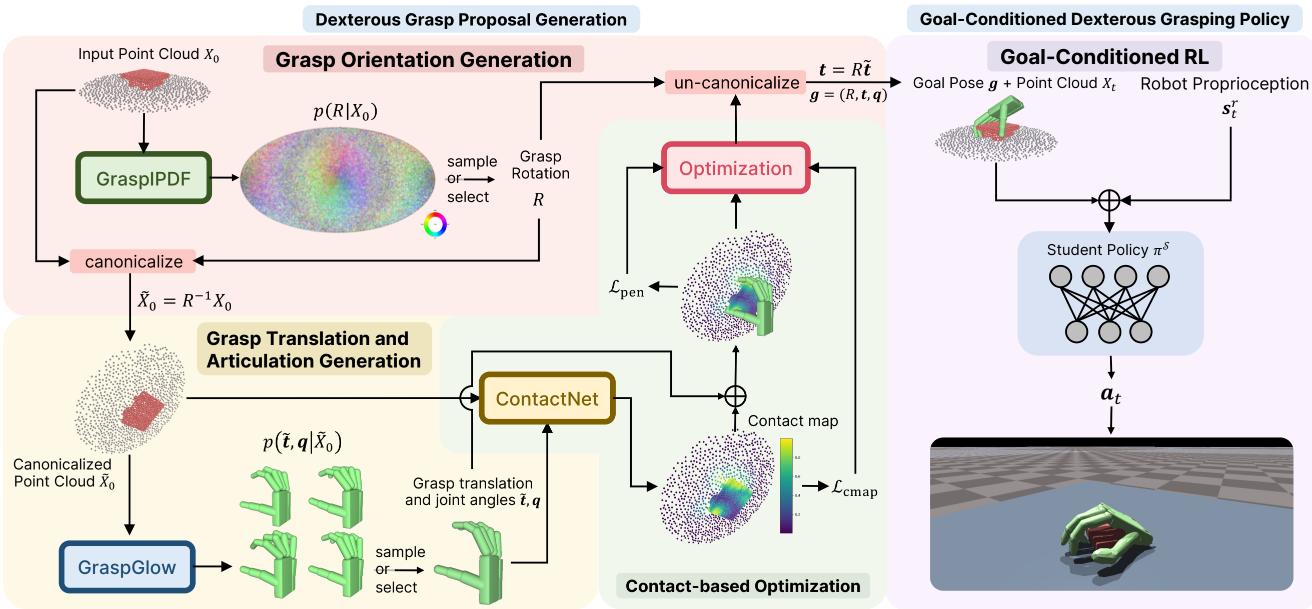


Figure 2. **Method overview**. The left part is the first stage, which generates a dexterous grasp proposal. The input is the object point cloud at time step 0, X_0, fused from depth images, with ground truth segmentation of the table and the object. A rotation R is sampled from the distribution implied by the GraspIPDF, and the point cloud will be canonicalized by R^{-1} to $\tilde{X}_0$. The GraspGlow then samples the translation $\tilde{\boldsymbol{t}}$ and joint angles $\boldsymbol{q}$. Next, the ContactNet takes $\tilde{X}_0$ and a point cloud $\tilde{X}_H$ sampled from the hand to predict the ideal contact map $\boldsymbol{c}$ on the object. Then, the predicted hand pose is optimized based on the contact information. The final goal pose is transformed by R to align with the original visual observation. The right part is the second stage, the goal-conditioned dexterous grasping policy that takes the goal $\boldsymbol{g}$, point cloud X_t and robot proprioception $\boldsymbol{s}_t^r$ to take actions accordingly.

rotation is initialized so that the hand palm faces down and its $\phi_0 = \phi$. The joint angles of the hand are set to zero. The task is to grasp the object as specified by the goal grasp label g and lift it to a certain height. The task is successful if the position difference between the object and the target point is smaller than the threshold value $\boldsymbol{t}_0$.

Since directly training such a vision-based policy using reinforcement learning is challenging, we use the idea of *teacher-student learning*. We first use a popular on-policy RL algorithm, PPO [47], with our proposed object curriculum learning and state canonicalization, to learn an oracle teacher policy that can access the ground-truth states of the environment (*e.g.* object poses, velocities and object full point cloud). This information is very useful to the task and available in the simulator but not in the real world. Once the teacher finishes training, we use an imitation learning algorithm, DAgger [45], to distill this policy to a student policy that can only access realistic inputs.

3.2. Dexterous Grasp Proposal Generation

In this subsection, we introduce how we model the conditional probability distribution $p(\boldsymbol{g}|X_0) : \mathbf{SO(3)} \times \mathbb{R}^{3+K} \rightarrow \mathbb{R}$. By factorizing $p(\boldsymbol{g}|X_0)$ into two parts $p(R|X_0) \cdot p(\boldsymbol{t}, \boldsymbol{q}|X_0, R)$, we propose a three-stage pipeline: 1) given the point cloud observation, predict the conditional distribution of hand root rotation $p(R|X_0)$ using GraspIPDF (see Sec. 3.2.1) and then sample a single root rotation; 2) given the point cloud observation and the root rotation, predict the conditional distribution of the hand root translation and joint angles $p(\boldsymbol{t}, \boldsymbol{q}|X_0, R) = p(\tilde{\boldsymbol{t}}, \boldsymbol{q}|\tilde{X}_0)$ using GraspGlow (see Sec. 3.2.2) and then sample a single proposal; 3) optimize the sampled grasp pose g with ContactNet to improve physical plausibility (see Sec. 3.2.4).

3.2.1 GraspIPDF: Grasp Orientation Generation

Inspired by IPDF [34], a probabilistic model over $\mathbf{SO(3)}$, we propose GraspIPDF $f(X_0, R)$ to predict the conditional probability distribution $p(R|X_0)$ of the hand root rotation R given the point cloud observation X_0. This model takes X_0 and R as inputs, extracts the geometric features with a PointNet++ [37] backbone, and outputs an unnormalized joint log probability density $f(X_0, R) = \alpha \log(p(X_0, R))$, where α is a normalization constant. The normalized probability density is recovered by computing:

$$p(R|X_0) = \frac{p(X_0, R)}{p(X_0)} \approx \frac{1}{V} \frac{\exp(f(X_0, R))}{\sum_i^M \exp(f(X_0, R_i))} \quad (1)$$

where M is the number of volume partitions and $V = \frac{\pi^2}{M}$ is the volume of partition.

During train time, GraspIPDF is supervised by an NLL loss $\mathcal{L} = -\log(p(R_0|X_0))$, where R_0 is a ground-truth hand root rotation. During test time, we generate an equivolumetric grid on $\mathbf{SO(3)}$ as in [19, 34, 53] and sample rotations according to their queried probabilities.

3.2.2 GraspGlow: Grasp Translation and Articulation Generation given Orientation

To condition a probabilistic model on X_0 and R simultaneously, we propose to canonicalize the point cloud to $\tilde{X}_0 = R^{-1}X_0$. This trick simplifies the task from predicting valid grasps for observation X_0 with R as the hand root rotation, to predicting it for $\tilde{X}_0$ with identity as hand rotation. Then, we use a PointNet [36] to extract the features of $\tilde{X}_0$, and model the conditional probability distribution $p(\boldsymbol{t}, \boldsymbol{q}|X_0, R) = p(\tilde{\boldsymbol{t}}, \boldsymbol{q}|\tilde{X}_0)$ where $\tilde{\boldsymbol{t}} = R^{-1}\boldsymbol{t}$ with Glow [23], a popular normalizing flow model that handles probabilistic modeling over Euclidean spaces.

During train time, the model is supervised by an NLL loss $\mathcal{L}_{\text{NLL}} = -\log(p(\tilde{\boldsymbol{t}}_{\text{gt}}, \boldsymbol{q}_{\text{gt}}|\tilde{X}_0))$ using ground truth grasp data tuples $(X_0, R_{\text{gt}}, \boldsymbol{t}_{\text{gt}}, \boldsymbol{q}_{\text{gt}})$. During test time, samples are drawn from the base distribution of Glow, and reconstructed into grasp poses using the bijection of the normalizing flow.

3.2.3 End-to-End Training with ContactNet

Inspired by [22], we use ContactNet to model a mapping from flawed raw grasp pose predictions to ideal contact patterns between the hand and the object. The input of the ContactNet is the canonicalized object point cloud $\tilde{X}_0$ and the sampled hand point cloud $\tilde{X}_H$; the output is the contact heat $c_i \in [0, 1]$ predicted at each point $\boldsymbol{p}_i \in \tilde{X}_0$. The ground truth of contact heat is given by $c_i = f(D_i(\tilde{X}_H)) = 2 - 2 \cdot (\text{sigmoid}(\beta\, D_i(\tilde{X}_H)))$, where β is a coefficient to help map the distance to $[0, 1]$, and $D_i(\tilde{X}_H) = \min_j \|\boldsymbol{p}_i - \boldsymbol{p}_j\|_2, \boldsymbol{p}_j \in \tilde{X}_H$.

Leveraging ContactNet, we construct a self-supervised task to improve the sample quality of GraspGlow by training it end-to-end with RotationNet. To be specific, in this stage, we first sample rotations with GraspIPDF, use them to canonicalize point clouds, then feed the point clouds to GraspGlow to get translations and joint angles samples. Next, ContactNet takes the grasp samples, and outputs ideal contact maps. Here we use four additional loss terms: 1) $\mathcal{L}_{\text{cmap}}$: MSE between current and target contact map; 2) $\mathcal{L}_{\text{pen}}$: Total penetration from object point cloud to hand mesh calculated using signed squared distance function; 3)$\mathcal{L}_{\text{tpen}}$: Total penetration depth from some chosen hand key points to the plane; 4)$\mathcal{L}_{\text{spen}}$: Self penetration term inspired by [54]. Then the joint loss becomes $\mathcal{L}_{\text{joint}} = \mathcal{L}_{\text{NLL}} + \mathcal{L}_{\text{add}}$ where $\mathcal{L}_{\text{add}}$ is defined as:

$$\mathcal{L}_{\text{add}} = \lambda_{\text{cmap}}\mathcal{L}_{\text{cmap}} + \lambda_{\text{pen}}\mathcal{L}_{\text{pen}} + \lambda_{\text{tpen}}\mathcal{L}_{\text{tpen}} + \lambda_{\text{spen}}\mathcal{L}_{\text{spen}} \quad (2)$$

In this stage, we freeze RotationNet as experiments demonstrate that it learns quite well in its own stage.

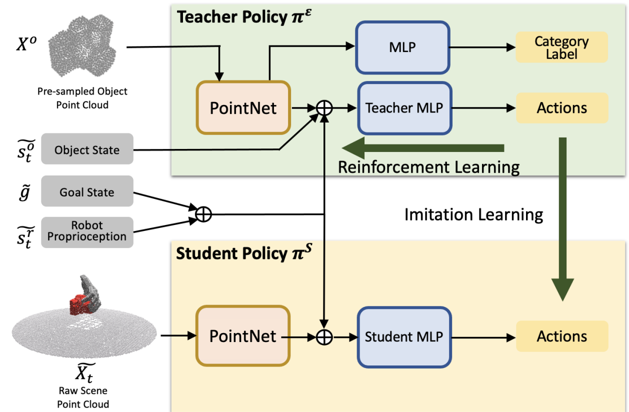


Figure 3. **The goal-conditioned dexterous grasping policy pipeline**. $\widetilde{\mathcal{S}_t^{\mathcal{E}}} = (\widetilde{\boldsymbol{s}_t^r}, \widetilde{\boldsymbol{s}_t^o}, X^O, \widetilde{g})$ and $\widetilde{\mathcal{S}_t^{\mathcal{S}}} = (\widetilde{\boldsymbol{s}_t^r}, \widetilde{X_t}, \widetilde{g})$ denote the input state of the teacher policy and student policy after state canonicalization, respectively; $\oplus$ denotes concatenation.

3.2.4 Test-Time Contact-based Optimization

Since the randomness of the flow sometimes leads to small artifacts, the raw outputs of GraspGlow may contain slight penetration and inexact contact. So we use ContactNet to construct a self-supervised optimization task for test-time adaptation to adjust the imperfect grasp as in [22].

When GraspGlow predicts a grasp, ContactNet takes the scene and hand point cloud, then outputs a target contact map on the scene point cloud. Next, the grasp pose is optimized for 300 steps to match this contact pattern. The total energy E_{TTA} consists of the four additional loss term described in Sec. 3.2.3:

$$\lambda_{\text{cmap}}^{\text{TTA}} E_{\text{cmap}} + \lambda_{\text{pen}}^{\text{TTA}} E_{\text{pen}} + \lambda_{\text{tpen}}^{\text{TTA}} E_{\text{tpen}} + \lambda_{\text{spen}}^{\text{TTA}} E_{\text{spen}} \quad (3)$$

For network structures, hyperparameters, and other implementation details, please refer to Sec. B.1.1 of our supp.

3.3. Goal-Conditioned Dexterous Grasping Policy

In this section, we will introduce our *goal-conditioned* grasp policy learning. We introduce our proposed state canonicalization, object curriculum learning and other method details for training the teacher policy in Sec. 3.3.1. We then introduce the vision-based student policy training in Sec. 3.3.2. The teacher policy $\pi^{\mathcal{E}}$ has a state space $\mathcal{S}_t^{\mathcal{E}} = (\boldsymbol{s}_t^r, \boldsymbol{s}_t^o, X^O, \boldsymbol{g})$ where $\boldsymbol{s}_t^r$ is the robot hand proprioception state, $\boldsymbol{s}_t^o$ is the object state, X^O is the pre-sampled object point cloud and $\boldsymbol{g}$ is the goal grasp label. The state of the student is defined as $\mathcal{S}^{\mathcal{S}} = (\boldsymbol{s}_t^r, X_t, \boldsymbol{g})$ where X_t is the raw scene point cloud. Details about the state and action space are provided in Sec. C.1 of our supp.

3.3.1 Learning Teacher Policy

We use a model-free RL framework to learn the oracle teacher policy. The goal of the goal-conditioned RL is to maximize the expected reward $\mathbb{E}\left[\sum_{t=0}^{T-1}\gamma^t\mathcal{R}(\boldsymbol{s}_t, \boldsymbol{a}_t, g)\right]$ with $\pi^{\mathcal{E}}$. We use PPO [47] for policy updating in our method. Inspired by ILAD [52], we use a PointNet [36] to extract the geometry feature of the object. ILAD pre-trains the PointNet using behavior cloning from the motion planning demonstrations and jointly trains the PointNet using behavior cloning from the RL demonstrations during policy learning. Although ILAD performs very well in one single category (96% success rate [52]), we find that directly using this method cannot get good results under the setting of *goal-conditioned* and *cross-category*. We then propose several techniques on top of it. First, we do state canonicalization according to the initial object pose which improves the sample efficiency of the RL with diverse goal inputs. Second, we design a novel goal-conditioned function. We then do 3-stage curriculum policy learning which significantly improves the performance under *cross-category* setting. Additionally, we find that doing object category classification when joint training the PointNet using behavior cloning which is proposed in [52] can also improve the performance under *cross-category* setting.

State Canonicalization Ideally, this goal-conditioned grasping policy should be $\mathbf{SO(2)}$ equivariant, that is, when we rotate the whole scene and the goal grasp label with the same angle ϕ about the z axis (gravity axis), the grasping trajectory generated by the policy should rotate in the same way. To ensure this $\mathbf{SO(2)}$ equivariance, we define a static reference frame (denoted by $\widetilde{\cdot}$) according to the initial hand pose: the origin of the reference frame is at the initial hand translation $(0, 0, h_0)$ and the Euler angle of the reference frame is $(0, 0, \phi)$ so that the initial Euler angle of the hand in this reference frame is always a fixed value $\widetilde{R_0} = (\frac{\pi}{2}, 0, 0)$. Before we input the states to the policy, we transfer the states from the world frame to this reference frame: $\widetilde{\mathcal{S}_t^{\mathcal{E}}} = (\widetilde{\boldsymbol{s}_t^r}, \widetilde{\boldsymbol{s}_t^o}, X^O, \widetilde{g})$. Thus, the system is $\mathbf{SO(2)}$ equivariant to ϕ. This improves the sample efficiency of the goal-conditioned RL.

Object Curriculum Learning Since it's difficult to train the grasping policy *cross-category* due to the topological and geometric variations in different categories, we propose to build an object curriculum learning method to learn $\pi^{\mathcal{E}}$. We find that $\pi^{\mathcal{E}}$ can already perform very well when training on one single object but fails when training directly on different category objects simultaneously. We apply curriculum learning techniques. Our curriculum learning technique is constructed as follows: first train the policy on one single object and then on different objects in one category, several representative categories, and finally on all the categories. We find that this 3-stage curriculum learning significantly boosts the success rates. More details about the ablations on the curriculum stages are in Sec. 4.3 and Tab. 4.

Pre-training and Joint Training PointNet with Classification ILAD [52] proposed an important technique that jointly does geometric representation learning using behavior cloning when doing policy learning. We use this technique in our method and add additional object category classification objectives to update the PointNet.

Goal-conditioned Reward Function We are supposed to conquer the dexterous manipulation problem with RL, therefore the reward design is crucial. Here is our novel goal-conditioned reward function which can guide the robot to grasp and lift the object by the standard of the goal grasp label: $r = r_{\text{goal}} + r_{\text{reach}} + r_{\text{lift}} + r_{\text{move}}$.

The goal reward r_{goal} punished distance between the current hand configuration and the goal hand configuration. The reaching reward r_{reach} encourages the robot fingers to reach the object. The lifting reward r_{lift} encourages the robot hand to lift the object. It's **non-zero if and only if** the goal grasp label is reached within a threshold. The moving reward r_{move} encourages the object to reach the target.

3.3.2 Distilling to the Vision-based Student Policy

We then distill the teacher policy $\pi^{\mathcal{E}}$ into the student policy $\pi^{\mathcal{S}}$ using DAgger [45] which is an imitation method that overcomes the covariate shift problem of behavior cloning. We optimize $\pi^{\mathcal{S}}$ by: $\pi^{\mathcal{S}} = \arg\min_{\pi^{\mathcal{S}}} \|\pi^{\mathcal{E}}(\mathcal{S}_t^{\mathcal{E}}) - \pi^{\mathcal{S}}(\mathcal{S}_t^{\mathcal{S}})\|_2$. We also use state canonicalization but this time ϕ is the initial Euler angle of the robot hand root around the z axis in the world frame because we don't know the object pose in the student input states. Similarly, we transfer the states from the world frame to this reference frame: $\widetilde{\mathcal{S}_t^{\mathcal{S}}} = (\widetilde{\boldsymbol{s}_t^r}, \widetilde{X_t}, \widetilde{g})$. The pipeline/network architecture is shown in Fig. 3. For implementation details, please refer to Sec. B.2.1 of our supp.

4. Experimentals

4.1. Data Generation and Statistics

We used a similar method from [51] to synthesize grasps.

Data Generation First, we randomly select an object from the pool of our training instances and let it fall randomly onto the table from a high place. Next, we randomly initialize a dexterous hand and optimize it into a plausible grasp. The optimization is guided by an energy function proposed by [51]. We add an energy term on top of this to punish penetration between the hand and the table. Finally, the grasps are filtered by penetration depth and simulation success in Isaac. Please refers to Sec. A of our supp.

Statistics We generated 1.12 million valid grasps for 5519 object instances in 133 categories. These objects are split into three sets: training instances (3251), seen category unseen instances (754), unseen category instances (1514).

4.2. Results on Grasp Proposal Generation

Baselines [22] takes point cloud as input, generates grasps with CVAE, then performs test-time adaptation with

Method	seen cat		unseen cat		$\sigma_R \uparrow$ (degree)	$\sigma_{T\vert R} \uparrow$ (cm)	$\sigma_{\theta\vert R} \uparrow$ (degree)	$\sigma_{\text{keypoints}} \uparrow$ (cm)
	$Q_1 \uparrow$	obj. pen.$\downarrow$	$Q_1 \uparrow$	obj. pen.$\downarrow$				
GraspTTA [22] (C + T)	0.0269	0.354	0.0239	0.363	4.9	/	/	2.909
DDG [26]	0.0357	0.319	0.0223	0.338	0.0	/	/	0.000
R + C + T	0.0362	0.251	**0.0336**	0.235	**128.0**	0.095	0.227	5.982
ReLie [15] + T	0.0190	0.219	0.0191	0.225	109.9	/	/	**6.698**
ProHMR [24] + T	0.0210	**0.202**	0.0221	**0.192**	88.4	/	/	5.837
ours (R + GL + T)	**0.0423**	0.205	0.0322	0.220	127.6	**1.143**	**5.806**	6.389

Table 1. **Results on grasp goal generation.** R: GraspIPDF, C: CVAE, T: test-time adaptation, GL: GraspGlow, and obj. pen. is the penetration between the hand and the object.

ContactNet. Apart from this baseline, we also designed two ablations to verify the two key designs in our pipeline. First, we substituted GraspGlow for the same CVAE from [22] to demonstrate the severe mode collapse of CVAE. Second, we substituted GraspIPDF with ReLie to demonstrate the problem of discontinuity, as described in Sec. 2.

Metrics We use some analytical metrics to evaluate quality and diversity. 1) Q_1 [18]. The smallest wrench needed to make a grasp unstable. 2) Object penetration depth(cm). Maximal penetration from object point cloud to hand mesh. 3) $\sigma^2_{R/\text{keypoints}}$. Variance of rotation or keypoints. 4) $\sigma^2_{T/\theta|R}$. Variance of translation or joint angles with fixed rotation.

Ablation 1: Decoupling. Tab. 1 shows that the Q_1 of the fourth row and the fifth row is significantly lower than the last row, implying that the normalizing flow on $\mathbf{SO(3)} \times \mathbb{R}^{3+22}$ failed to learn a good distribution. On the other hand, our model can produce grasps with much higher quality. We argue that this improvement is contributed by rotation factorization, which allows us to use IPDF and GLOW to model the distributions on $\mathbf{SO(3)}$ and $\mathbb{R}^{22}$ separately. However, as shown in Tab. 2, further decoupling translation and joint angles will result in worse performance as it is less end-to-end.

Ablation 2: GLOW vs CVAE. The reason we favored GLOW over CVAE can be interpreted from the last two columns of Tab. 1. If the input object point cloud is fixed, then no matter what the latent z vector is, CVAE will always collapse to a single mode. However, GLOW can propose diverse results. Fig. 4 shows a typical case.

Ablation 3: TTA. As discussed in Sec. 3.2.4 and shown in Tab. 2, poses sampled from flow are usually imperfect and need TTA to make them plausible.

4.3. Grasp Execution

Environment Setup and Data We use a subset of the train split proposed in Sec. 4.1 as our training data. For the state-based policy evaluation, we use the grasp proposals sampled by our grasp proposal generation module. For the final vision-based policy evaluation, we use both the training data ("GT" in Tab. 4) and grasp proposals sampled by our grasp proposal generation module ("pred" in Tab. 4) as

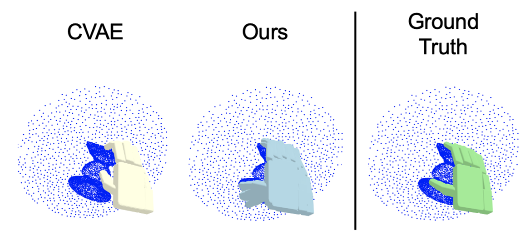


Figure 4. **Comparison of diversity in grasp translation and articulation given the rotation.** Left: 8 outputs of CVAE (completely collapsed to one pose); Middle: 8 outputs of GraspGLOW; Right: a ground truth grasp.

Method		decoup.T	w/o TTA	ours
seen cat.	$Q_1 \uparrow$	0.0003	0.0013	**0.0423**
	pen.$\downarrow$	0.862	0.744	**0.205**
unseen cat.	$Q_1 \uparrow$	0.0061	0.0000	**0.0322**
	pen.$\downarrow$	0.843	0.764	**0.220**

Table 2. **Ablation study on decoupling translation and joint angles and TTA.** decoup.T: decouple translation, w/o TTA: without TTA and pen. is the penetration between the hand and the object.

the goal of our policy to do testing on the train object set and test object set. Details are in Sec. C.2 of our supp.

Baselines and Compared Methods We adopt PPO [47] as our RL baseline and DAPG [42] as our Imitation Learning (IL) baseline. We also compared our method with ILAD [52] which can reach a very high success rate in one category and can generalize to novel object instances within the same category. We further do ablations on our proposed techniques. All methods are compared under the same setting of teacher policy. Once we get the teacher policy with the highest success rate, we distill it to the student policy.

Main Results The top half of Tab. 3 shows that our method outperforms baselines by a large margin. The PPO (RL) and DAPG (IL) baseline only achieve an average success rate of 14% and 13% on train set. Our teacher method achieves an average success rate of 74% on train set, 69% on test set, which is about **49%** and **47%** improvement over ILAD. Since our method has a teacher policy with the high-

Model	Train	Test	
		unseen obj seen cat	unseen cat
MP	0.12±0.01	0.02±0.00	0.02±0.01
PPO [47]	0.14±0.06	0.11±0.04	0.09±0.06
DAPG [42]	0.13±0.05	0.13±0.08	0.11±0.05
ILAD [52]	0.25±0.03	0.22±0.04	0.20±0.05
Ours	**0.74±0.07**	**0.71±0.05**	**0.66±0.06**
Ours(w/o SC)	0.59±0.06	0.54±0.07	0.51±0.04
Ours(w/o cls)	0.65±0.05	0.64±0.06	0.60±0.07
Ours(w/o OCL)	0.31±0.07	0.23±0.06	0.21±0.04
Ours(1-stage OCL)	0.58±0.07	0.55±0.03	0.55±0.05
Ours(2-stage OCL)	0.68±0.06	0.67±0.07	0.62±0.05

Table 3. **The success rate of state-based policy**. The experiment is evaluated with three different random seeds. "MP": motion planning;"SC": state canonicalization; "cls"; joint learning object classification; "OCL": object curriculum learning.

est success rate, we distill this policy to a vision-based student policy and the success rate decreases by 6% and 7% on train and test set respectively (the first line of Tab. 4). This indicates that the vision-based, goal-conditional, and cross-category setting is difficult and the student's performance is limited by the teacher's performance.

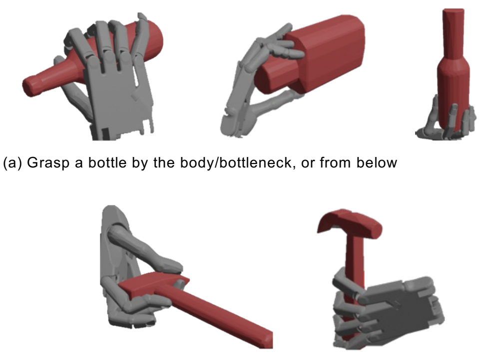
Ablation Results We evaluate our method without the state canonicalization (w/o SC), without the object classification (w/o classification) and without object curriculum learning (w/o OCL). Tab. 3 shows that each technique yields considerable performance improvement. We do more specific ablations on stages of object curriculum learning (OCL). We stipulate—**a**: train on one object; **b**: train on one category; **c**: train on 3 representative categories; **d**: train on all the categories. We formulate the experiment as follows: w/o OCL: **d**; 1-stage OCL: **a** $\rightarrow$ **d**; 2-stage OCL: **a** $\rightarrow$ **b** $\rightarrow$ **d**; 3-stage OCL: **a** $\rightarrow$ **b** $\rightarrow$ **c** $\rightarrow$ **d**; As shown in Tab. 3, without OCL, the policy seldom succeeds both during training and testing. However, as the total stages of the curriculum increase, the success rate improves significantly. In Tab. 4, we conduct the robustness test for our vision-based policy by jittering our predicted grasp poses to cause small penetration (a), large penetration (b), and no contact (c) and observe our grasp execution policy is robust to such errors.

4.4. Language-guided Dexterous Grasping

A natural downstream task for our method is to introduce specific semantic meaning parsing, *e.g.* "grasping a hammer by the handle". Thanks to the great diversity of grasp proposals, this can be easily done by adding a filtering module using CLIP [41], a large text-image pre-train model which shows significant generalizability for computing the similarity between text and images. Combined with the goal-conditioned policy network, the robot is endowed with the ability to grasp an object according to human instructions. One of the most promising applications is for the functional grasping of certain tools, *e.g.* bottles and hammers. In detail, we select images rendered from grasp proposals with the highest image-text similarity following the user command (*e.g.* "A robot hand grasps a hammer by the handle."). As shown in Fig. 5, with only 10 minutes of fine-tuning, the model can achieve around 90% accuracy in the bottle and hammer categories. This validates the feasibility of generating grasps with specific semantic meanings using our generation pipeline together with the CLIP model.

Penetration (cm)	Train	Test	
		unseen obj seen cat	unseen cat
0.117 (GT)	0.68±0.06	0.65±0.05	0.63±0.04
0.208 (pred)	0.66±0.04	0.59±0.04	0.58±0.05
0.512 (a)	0.63±0.05	0.54±0.05	0.57±0.04
1.058 (b)	0.47±0.04	0.37±0.05	0.39±0.04
-0.309 (c)	0.50±0.03	0.38±0.02	0.35±0.03

Table 4. **The success rate of our vision-based policy**. We test our trained vision-based policy with jittered goal grasp on the train and test set.



(a) Grasp a bottle by the body/bottleneck, or from below

(b) Grasp a hammer by the head/handle

Figure 5. **Qualitative results of language-guided grasp proposal selection.** CLIP can select proposals complying with the language instruction, allowing the goal-conditioned policy to execute potentially functional grasps.

5. Conclusions and Discussions

In our work, we propose a novel two-stage pipeline composed of grasp proposal generation and goal-conditioned grasp execution. The whole pipeline for the first time demonstrates universal dexterous grasping over thousand of objects under a realistic robotic setting and thus has the potential to transfer to real-world settings. The limitation is that we only tackle the grasping of rigid objects, in contrast to articulated objects like scissors. Furthermore, functional dexterous grasping still remains a challenging but promising field to explore.

References

[1] ShadowRobot. URL https://www.shadowrobot.com/dexterous-hand-series/, 2005. 2

[2] Ilge Akkaya, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael Ribas, et al. Solving rubik's cube with a robot hand. *arXiv preprint arXiv:1910.07113*, 2019. 2

[3] Sheldon Andrews and Paul G Kry. Goal directed multi-finger manipulation: Control policies and analysis. *Computers & Graphics*, 37(7):830–839, 2013. 3

[4] Yunfei Bai and C. Karen Liu. Dexterous manipulation using both palm and fingers. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1560–1565, 2014. 3

[5] Georgios Batzolis, Jan Stanczuk, Carola-Bibiane Schönlieb, and Christian Etmann. Conditional image generation with score-based diffusion models. *arXiv preprint arXiv:2111.13606*, 2021. 2

[6] Samarth Brahmbhatt, Cusuh Ham, Charles C. Kemp, and James Hays. ContactDB: Analyzing and predicting grasp contact via thermal imaging. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6 2019. 2

[7] Samarth Brahmbhatt, Ankur Handa, James Hays, and Dieter Fox. ContactGrasp: Functional Multi-finger Grasp Synthesis from Contact. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019. 3

[8] Michel Breyer, Jen Jen Chung, Lionel Ott, Roland Siegwart, and Juan Nieto. Volumetric grasping network: Real-time 6 dof grasp detection in clutter. *arXiv preprint arXiv:2101.01132*, 2021. 2

[9] Qiuyu Chen, Karl Van Wyk, Yu-Wei Chao, Wei Yang, Arsalan Mousavian, Abhishek Gupta, and Dieter Fox. Learning robust real-world dexterous grasping policies via implicit shape augmentation. In *6th Annual Conference on Robot Learning*. 2

[10] Sammy Christen, Muhammed Kocabas, Emre Aksan, Jemin Hwangbo, Jie Song, and Otmar 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. 3

[11] Enric Corona, Albert Pumarola, Guillem Alenya, Francesc Moreno-Noguer, and Grégory Rogez. Ganhand: Predicting human grasp affordances in multi-object scenes. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5031–5041, 2020. 3

[12] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014. 2, 3

[13] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016. 2, 3

[14] Mehmet R. Dogar and Siddhartha S. Srinivasa. Push-grasping with dexterous hands: Mechanics and a method. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2123–2130, 2010. 3

[15] Luca Falorsi, Pim de Haan, Tim R Davidson, and Patrick Forré. Reparameterizing distributions on lie groups. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3244–3253. PMLR, 2019. 2, 3, 7

[16] Hongjie Fang, Hao-Shu Fang, Sheng Xu, and Cewu Lu. Transcg: A large-scale real-world dataset for transparent object depth completion and a grasping baseline. *IEEE Robotics and Automation Letters*, pages 1–8, 2022. 2

[17] Hao-Shu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. Graspnet-1billion: A large-scale benchmark for general object grasping. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11444–11453, 2020. 2

[18] Carlo Ferrari and John F Canny. Planning optimal grasps. In *ICRA*, volume 3, page 6, 1992. 7

[19] K. M. Gorski, E. Hivon, A. J. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelmann. HEALPix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622(2):759–771, apr 2005. 5

[20] Minghao Gou, Hao-Shu Fang, Zhanda Zhu, Sheng Xu, Chenxi Wang, and Cewu Lu. Rgb matters: Learning 7-dof grasp poses on monocular rgbd images. In *Proceedings of the International Conference on Robotics and Automation (ICRA)*, 2021. 2

[21] Patrick Grady, Chengcheng Tang, Christopher D. Twigg, Minh Vo, Samarth Brahmbhatt, and Charles C. Kemp. ContactOpt: Optimizing contact to improve grasps. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[22] Hanwen Jiang, Shaowei Liu, Jiashun Wang, and Xiaolong Wang. Hand-object contact consistency reasoning for human grasps generation. In *Proceedings of the International Conference on Computer Vision*, 2021. 2, 3, 5, 6, 7

[23] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018. 2, 3, 5

[24] Nikos Kolotouros, Georgios Pavlakos, Dinesh Jayaraman, and Kostas Daniilidis. Probabilistic modeling for human mesh recovery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11605–11614, 2021. 3, 7

[25] Puhao Li, Tengyu Liu, Yuyang Li, Yiran Geng, Yixin Zhu, Yaodong Yang, and Siyuan Huang. Gendexgrasp: Generalizable dexterous grasping. *arXiv preprint arXiv:2210.00722*, 2022. 3

[26] Min Liu, Zherong Pan, Kai Xu, Kanishka Ganguly, and Dinesh Manocha. Deep differentiable grasp planner for high-dof grippers. *arXiv preprint arXiv:2002.01530*, 2020. 3, 7

[27] Min Liu, Zherong Pan, Kai Xu, Kanishka Ganguly, and Dinesh Manocha. Deep differentiable grasp planner for high-dof grippers. *CoRR*, abs/2002.01530, 2020. 3

[28] Tengyu Liu, Zeyu Liu, Ziyuan Jiao, Yixin Zhu, and Song-Chun Zhu. Synthesizing diverse and physically stable grasps with arbitrary hand structures by differentiable force closure estimation. *CoRR*, abs/2104.09194, 2021. 2, 3

[29] Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li 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*, pages 21013–21022, 2022. 2

[30] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, and Gavriel State. Isaac gym: High performance gpu-based physics simulation for robot learning, 2021. 3

[31] Priyanka Mandikal and Kristen Grauman. Dexvip: Learning dexterous grasping with human hand pose priors from video. In *Conference on Robot Learning (CoRL)*, 2021. 3

[32] Priyanka Mandikal and Kristen Grauman. Learning dexterous grasping with object-centric visual affordances. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2021. 3

[33] A.T. Miller and P.K. Allen. Graspit! a versatile simulator for robotic grasping. *Robotics Automation Magazine, IEEE*, 11(4):110 – 122, dec. 2004. 2, 3

[34] Kieran A Murphy, Carlos Esteves, Varun Jampani, Srikumar Ramalingam, and Ameesh Makadia. Implicit-pdf: Non-parametric representation of probability distributions on the rotation manifold. In *Proceedings of the 38th International Conference on Machine Learning*, pages 7882–7893, 2021. 2, 3, 4, 5

[35] George Papamakarios, Eric T Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *J. Mach. Learn. Res.*, 22(57):1–64, 2021. 2

[36] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *arXiv preprint arXiv:1612.00593*, 2016. 5, 6

[37] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*, 2017. 4

[38] Yuzhe Qin, Binghao Huang, Zhao-Heng Yin, Hao Su, and Xiaolong Wang. Generalizable point cloud policy learning for sim-to-real dexterous manipulation. In *6th Annual Conference on Robot Learning*. 2

[39] Yuzhe Qin, Binghao Huang, Zhao-Heng Yin, Hao Su, and Xiaolong Wang. Dexpoint: Generalizable point cloud reinforcement learning for sim-to-real dexterous manipulation. *arXiv preprint arXiv:2211.09423*, 2022. 3

[40] Yuzhe Qin, Yueh-Hua, Shaowei Liu, Hanwen Jiang, Rui Yang, Yang Fu, and Xiaolong Wang. Dexmv: Imitation learn for dexterous manipulation from human videos, 2021. 2, 3

[41] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 8

[42] Aravind Rajeswaran, Vikash Kumar, Abhishek Gupta, Giulia Vezzani, John Schulman, Emanuel Todorov, and Sergey Levine. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. *arXiv preprint arXiv:1709.10087*, 2017. 3, 7, 8

[43] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 2

[44] Carlos Rosales, Lluís Ros, Josep M. Porta, and Raúl Suárez. Synthesizing grasp configurations with specified contact regions. *The International Journal of Robotics Research*, 30(4):431–443, 2011. 3

[45] Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635. JMLR Workshop and Conference Proceedings, 2011. 4, 6

[46] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 2

[47] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 4, 6, 7, 8

[48] Qijin She, Ruizhen Hu, Juzhan Xu, Min Liu, Kai Xu, and Hui Huang. Learning high-dof reaching-and-grasping via dynamic representation of gripper-object interaction. *arXiv preprint arXiv:2204.13998*, 2022. 3

[49] Martin Sundermeyer, Arsalan Mousavian, Rudolph Triebel, and Dieter Fox. Contact-graspnet: Efficient 6-dof grasp generation in cluttered scenes. 2021. 2

[50] Chenxi Wang, Hao-Shu Fang, Minghao Gou, Hongjie Fang, Jin Gao, and Cewu Lu. Graspness discovery in clutters for fast and accurate grasp detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15964–15973, October 2021. 2

[51] Ruicheng Wang, Jialiang Zhang, Jiayi Chen, Yinzhen Xu, Puhao Li, Tengyu Liu, and He Wang. Dexgraspnet: A large-scale robotic dexterous grasp dataset for general objects based on simulation. *arXiv preprint arXiv:2210.02697*, 2022. 3, 6

[52] Yueh-Hua Wu, Jiashun Wang, and Xiaolong Wang. Learning generalizable dexterous manipulation from human grasp affordance. *arXiv preprint arXiv:2204.02320*, 2022. 2, 3, 6, 7, 8

[53] Anna Yershova, Swati Jain, Steven M. Lavalle, and Julie C. Mitchell. Generating uniform incremental grids on so(3) using the hopf fibration. *International Journal of Robotics Research*, 29(7):801–812, June 2010. 5

[54] Tianqiang Zhu, Rina Wu, Xiangbo Lin, and Yi Sun. Toward human-like grasp: Dexterous grasping via semantic representation of object-hand. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15741–15751, 2021. 5