

UltraDexGrasp: Learning Universal Dexterous Grasping for Bimanual Robots with Synthetic Data

Sizhe Yang^{1,2} Yiman Xie^{1,3} Zhixuan Liang^{1,4} Yang Tian^{1,5} Jia Zeng¹

Dahua Lin^{1,2} Jiangmiao Pang¹

¹Shanghai AI Laboratory ²The Chinese University of Hong Kong

³Zhejiang University ⁴The University of Hong Kong ⁵Peking University

Project page: <https://yangsizhe.github.io/ultradexgrasp/>

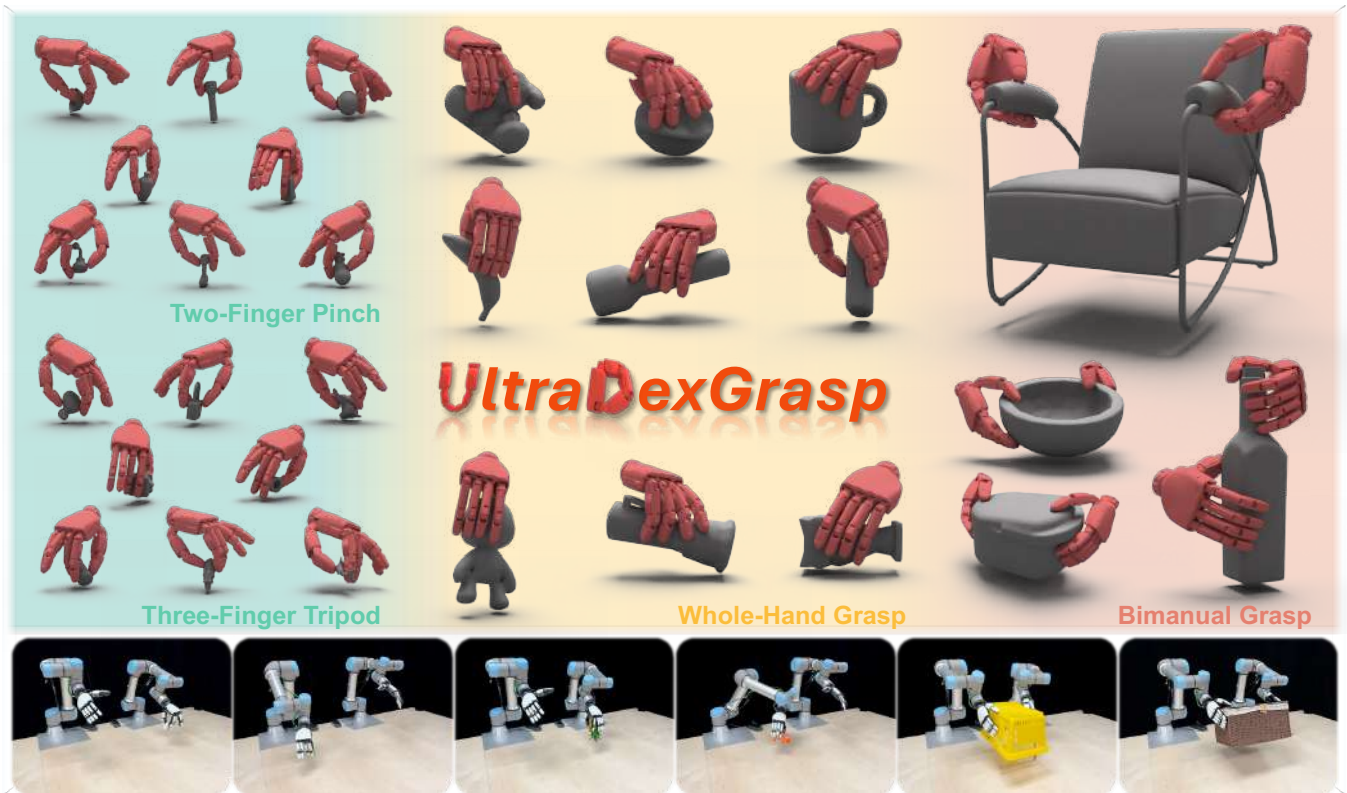


Fig. 1: **Overview.** UltraDexGrasp is a framework for universal dexterous grasping with bimanual robots. The proposed data generation pipeline integrates an optimization-based grasp synthesizer with a planning-based demonstration generation module, and supports multiple grasp strategies, including two-finger pinch, three-finger tripod, whole-hand grasp, and bimanual grasp. Trained on data produced by this pipeline, the policy demonstrates robust zero-shot sim-to-real transfer and strong generalization to novel objects with varied shapes, sizes, and weights.

Abstract—Grasping is a fundamental capability for robots to interact with the physical world. Humans, equipped with two hands, autonomously select appropriate grasp strategies based on the shape, size, and weight of objects, enabling robust grasping and subsequent manipulation. In contrast, current robotic grasping remains limited, particularly in multi-strategy settings. Although substantial efforts have targeted parallel-gripper and single-hand grasping, dexterous grasping for bimanual robots remains underexplored, with data being a primary bottleneck. Achieving physically plausible and geometrically conforming grasps that can withstand external wrenches poses significant challenges. To address these issues,

we introduce *UltraDexGrasp*, a framework for universal dexterous grasping with bimanual robots. The proposed data-generation pipeline integrates optimization-based grasp synthesis with planning-based demonstration generation, yielding high-quality and diverse trajectories across multiple grasp strategies. With this framework, we curate *UltraDexGrasp-20M*, a large-scale, multi-strategy grasp dataset comprising 20 million frames across 1,000 objects. Based on UltraDexGrasp-20M, we further develop a simple yet effective grasp policy that takes point clouds as input, aggregates scene features via unidirectional attention, and predicts control commands. Trained exclusively on synthetic data, the policy achieves robust

zero-shot sim-to-real transfer and consistently succeeds on novel objects with varied shapes, sizes, and weights, attaining an average success rate of 81.2% in real-world universal dexterous grasping. To facilitate future research on grasping with bimanual robots, we open-source the data generation pipeline at <https://github.com/InternRobotics/UltraDexGrasp>.

I. INTRODUCTION

Dexterous grasping is a critical step toward enabling robots to manipulate objects with human-level proficiency. As the first stage of many manipulation skills, it bridges the transition from non-contact to contact interaction. In daily life, humans autonomously select grasping strategies that accommodate variations in the shape, size, and weight of objects: large, heavy objects generally require a coordinated bimanual grasp to maintain balance; medium-sized objects can be stably grasped by a single hand using all five fingers; and small objects, which provide insufficient surface area, are best handled with two-finger pinch or three-finger tripod. Humans can also adapt hand postures to object geometry, maximizing surface conformity to ensure robust grasping and subsequent manipulation. In contrast, current robotic grasping capabilities fall far short of human performance. Although the burgeoning humanoid-robot market has accelerated the development of dual-arm platforms and dexterous hands, data and algorithms remain the principal bottleneck. Considerable efforts [1]–[7] have been devoted to parallel-gripper and single-hand grasping, yet universal dexterous grasping for bimanual robots remains underexplored.

A central difficulty of universal dexterous grasping is producing high-quality data. Existing data generation paradigms for dexterous grasping fall into three strands: 1) reinforcement learning (RL) with privileged information that trains experts for grasping [8]–[11]; 2) optimization-based synthesis that minimizes grasp-related costs [3], [12]–[14]; and 3) learning-based synthesis that trains generative models to predict diverse grasps [15]–[17]. However, RL-trained experts are typically deterministic after training, thus mapping a given observation to a single action, which leads to repetitive postures and a lack of diversity. Optimization- and learning-based synthesis are largely open-loop, struggle in dynamic real-world scenarios and typically ignore arm kinematics. Moreover, previous approaches are predominantly limited to single-hand settings. Universal dexterous grasping for dual-arm robots poses exceptional challenges for data generation, due to the extensive degrees of freedom, requirements for bimanual coordination, and the multitude of possible grasping strategies. Whether data are collected via real-world teleoperation or generated in simulation, producing physically plausible and geometrically conforming grasps that resist external wrenches, is essential though difficult.

To address these challenges, we propose **UltraDexGrasp**, a framework for universal dexterous grasping with bimanual robots. The proposed data-generation pipeline integrates an optimization-based grasp synthesizer with a planning-based demonstration generation module for coordinated dual-arm manipulation. This integration yields kinematically feasible

and natural closed-loop motions while preserving data diversity. Notably, the data generation supports multiple grasp strategies, including holding large objects with both hands, whole-hand grasp for medium-sized objects, and two-finger pinch or three-finger tripod for small objects. With this pipeline, we curate **UltraDexGrasp-20M**, the first large-scale multi-strategy dexterous grasp dataset for bimanual robots, comprising 20 million frames over 1,000 objects.

Building upon this, we develop a simple yet effective grasp policy that takes point clouds as input, aggregates scene features via unidirectional attention, and predicts control commands, enabling multiple grasp strategies and improving generalization across diverse objects. We conduct comprehensive experiments in both simulation and the real world to evaluate the robustness of the proposed policy trained on UltraDexGrasp-20M. Testing in simulation on 600 objects—including both seen and unseen objects during training—that vary widely in shape, weight (ranging from 5 g to 1,000 g), and size (from objects with a longest bounding box edge less than 0.03 m to those with a shortest edge exceeding 0.5 m), our policy attains an 84.0% average success rate, exceeding the next best baseline by 25.2 percentage points (approximately 43% relative improvement). The policy, trained exclusively on synthetic data, is further deployed in real-world scenarios. It adapts grasp strategies to different objects and successfully handles diverse items, achieving an 81.2% success rate across all objects.

In summary, our paper makes three contributions. First, we present UltraDexGrasp-20M and its data-generation pipeline that integrates optimization-based grasp synthesis with planning-based demonstration generation, producing high-quality and diverse trajectories across multiple grasp strategies. Second, we introduce a novel grasp policy for universal dexterous grasping with bimanual robots that enhances generalization to diverse objects. Third, we demonstrate that our policy, trained solely on synthetic data, enables robust zero-shot sim-to-real transfer and strong generalization to novel objects with varied shapes, sizes, and weights.

II. RELATED WORK

Dexterous grasp synthesis and dataset. Grasp synthesis is crucial for the advancement of dexterous grasping, and existing approaches can be broadly categorized into three types: sampling-based, optimization-based, and learning-based methods. Sampling-based methods [18] typically require simplification of the search space, which leads to limited grasp diversity. Many studies focus on optimization-based methods [3], [12]–[14], [19], [20], with some utilizing differentiable force closure to optimize grasp poses [3], [12], [21]. [22] formulates grasp synthesis as a bilevel optimization problem to generate high-quality grasps. Supervised learning-based methods [15]–[17], [23] require a small amount of grasp data to train generative models that can produce a large number of grasps for novel objects. Additionally, some works employ reinforcement learning to train experts for grasp generation [8]–[10]. Effective grasp synthesis and teleoperation have enabled the creation of

several dexterous grasp datasets [16], [24]–[27]. Recently, [28] proposed a bimanual grasp generation algorithm, but it imposes a simplified contact model, does not consider dual-arm coordination, and has not explored closed-loop control policies for universal grasping. Although [29] explores bimanual dexterous grasping, the RL-trained expert can only grasp limited objects, such as boxes. Scaling to a wider variety of objects is costly and generalization remains challenging. In contrast, our work proposes a framework for generating grasps for arbitrary objects and trains policies capable of handling novel and diverse objects.

Generalizable policy for robotic grasping. Grasping is a crucial skill for robotic manipulation. Several studies focus on data-efficient learning for grasping using real-world datasets [1], [4], [6], [30]. Some other works train and evaluate grasping policies in simulated environments, demonstrating high grasp success rates across large numbers of objects [8], [15], [31]. Some research targets functional grasping, emphasizing object part functionality and subsequent tasks [32], [33]. Recently, sim-to-real transfer of grasping policies has shown promising results. Methods such as [5], [16], [34], [35] generate large-scale grasping datasets using optimization-based or rule-based pipelines. Other approaches, including [9], [10], [29], [36], [37], employ reinforcement learning to train expert agents for grasp data generation, achieving strong real-world performance. Our proposed bimanual grasp policy is trained on data produced through the complementary integration of optimization-based and rule-based approaches, achieving robust real-world bimanual dexterous grasping.

III. PRELIMINARIES

A. Definition of Grasp Pose for Bimanual Robots

A grasp is defined as the act of restraining the motion of an object by applying forces and torques at a set of contact points. The grasp pose represents the critical configuration of the hands required to achieve a stable grasp. Specifically, we parameterize the grasp pose for bimanual robots as a tuple:

$$g = \{(\mathbf{t}_h, \mathbf{R}_h, \mathbf{q}_h) \mid h = 0, 1\}, \quad (1)$$

where $h \in \{0, 1\}$ denotes the index of the hand, $\mathbf{t}_h \in \mathbb{R}^3$ is the translation of the h -th hand, $\mathbf{R}_h \in SO(3)$ represents the rotation of the hand, and $\mathbf{q}_h \in \mathbb{R}^n$ denotes joint positions for the hand, with n being the number of joints of the hand.

B. Basics of Grasp Modeling

Grasping focuses on restraining the motion of an object through application of forces and torques at a set of contact points. We use point-on-plane contacts as the contact type, and hard finger model as the contact model, which is commonly used in force closure grasps. In the hard finger model, forces of arbitrary magnitude and direction can be applied within the limits defined by the friction cone:

$$\mathcal{F} = \{\mathbf{f} \mid \|\mathbf{f}_{\text{tan}}\| \leq \mu \|\mathbf{f}_n\|, f_z \geq 0\}, \quad (2)$$

where μ is the static friction coefficient of the object, $\mathbf{f}_n = [0, 0, f_z]$ is the vector component along the normal direction,

and $\mathbf{f}_{\text{tan}} = [f_x, f_y, 0]$ is the vector component tangent to the surface. Each contact point applies a wrench to the manipulated object:

$$\mathbf{w}_i = \begin{bmatrix} \mathbf{f}_i \\ \alpha(\mathbf{d}_i \times \mathbf{f}_i) \end{bmatrix}, \quad (3)$$

where $\mathbf{f}_i \in \mathbb{R}^3$ is the force, $\alpha(\mathbf{d}_i \times \mathbf{f}_i)$ is the torque, $\alpha \in \mathbb{R}$ is an arbitrary constant, and \mathbf{d}_i is the relative position of the i -th contact point with respect to the center of mass of the object. \mathbf{w}_i can be represented as $\mathbf{G}_i \mathbf{f}_i$, where \mathbf{G}_i is a wrench basis matrix for the i -th contact. For hard finger model, the grasp map \mathbf{G}_i is defined as:

$$\mathbf{G}_i = \begin{bmatrix} \mathbf{I}_{3 \times 3} \\ (\mathbf{p}_i - \mathbf{m})_{\times} \end{bmatrix} \mathbf{O}_i, \quad (4)$$

where $\mathbf{I}_{3 \times 3}$ is the 3×3 identity matrix, $(\mathbf{p}_i - \mathbf{m})_{\times}$ represents the cross product with $(\mathbf{p}_i - \mathbf{m})$, where \mathbf{p}_i denotes the contact position, \mathbf{m} represents the center of mass of the object, and \mathbf{O}_i is the rotation matrix from the local contact frame to the frame of the object.

Thus, the grasp wrench space \mathcal{W} for a grasp with k contact points is defined as the set of possible wrenches w that can be applied to the object:

$$\mathcal{W} := \left\{ w \mid w = \sum_{i=1}^k \mathbf{G}_i \mathbf{f}_i, \mathbf{f}_i \in \mathcal{F}_i, i = 1, \dots, k \right\}. \quad (5)$$

The grasp wrench space \mathcal{W} is expected to be sufficiently large to resist external wrenches.

IV. UNIVERSAL DEXTEROUS GRASP DATASET

As illustrated in Fig. 2, the pipeline begins with scene initialization, where object assets and the robot URDF are imported into the simulation environment. We select 1,000 distinct objects from DexGraspNet [3] to construct our object assets. Camera pose and joint impedance are randomized to reduce the sim-to-real gap. Given the meshes and poses of both the table and object, the grasp synthesizer generates a batch of feasible grasps, which are subsequently filtered and ranked to select the best candidate. We employ bimanual motion planning to compute collision-free and coordinated trajectories, which are then executed in simulation. Utilizing this scalable data generation pipeline, we curate the UltraDexGrasp-20M dataset.

A. Grasp Synthesis

Grasp synthesis involves two main stages. First, the hands are initialized near the object, and then, through optimization, physically plausible and geometrically conforming grasps are generated. In the initialization stage, we first obtain the convex hull of the object’s mesh. For unimanual grasps, such as whole-hand, two-finger pinch, and three-finger tripod, we sample a point on the surface of the object’s convex hull. For bimanual grasps, two points are sampled on the convex hull surface, positioned on opposite sides of the object’s center. The hand is positioned along the normal vector at the sampled point, with the palm facing the target object.

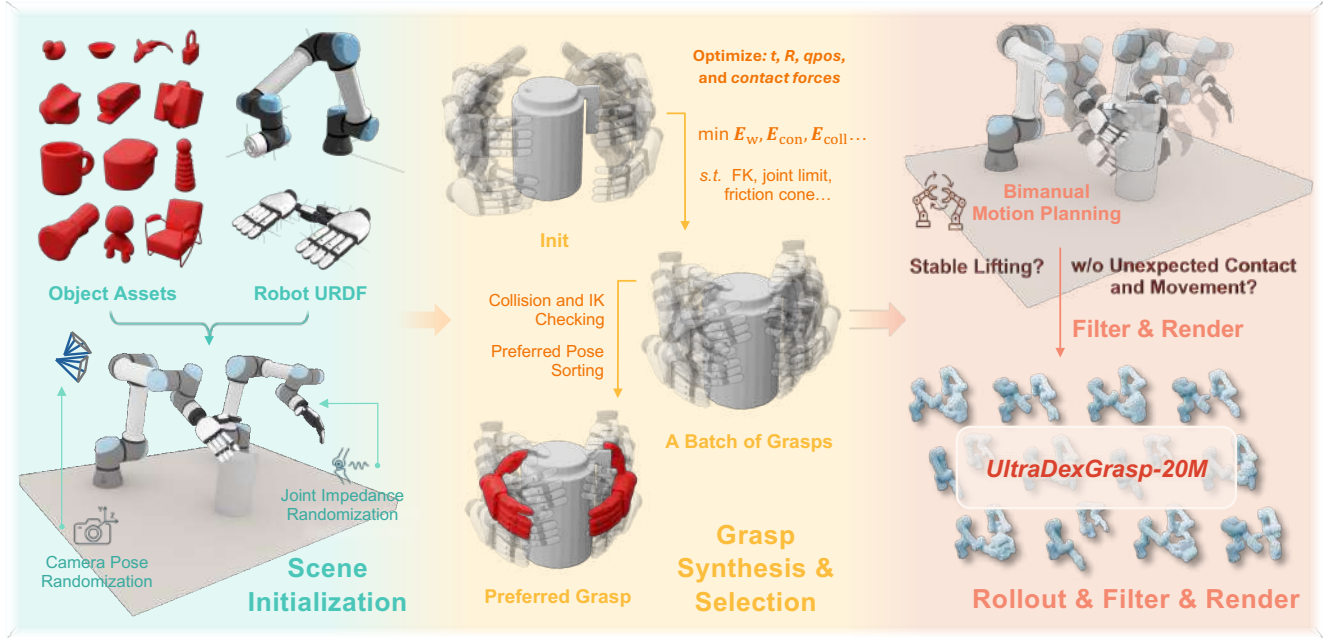


Fig. 2: **Overview of data generation pipeline.** We first collect diverse object assets and import the objects and the robot URDF files into the simulator. An optimization-based grasp synthesizer is then used to generate feasible grasps, from which the preferred grasp is selected. Finally, motion planning is employed to generate demonstration trajectories.

Given the object mesh M with center of mass m , two hands $h \in \{0, 1\}$ with known kinematics, candidate contact sets \bar{C}_h , a strategy-selected active contact set $\mathcal{C} \subset \bar{C}_0 \cup \bar{C}_1$, and the contact point $c \in \mathcal{C}$, the decision variables are the bimanual grasp pose $\mathbf{g} = \{(\mathbf{t}_h, \mathbf{R}_h, \mathbf{q}_h) \mid h = 0, 1\}$ and the contact forces $\{\mathbf{f}_c \in \mathbb{R}^3\}_{c \in \mathcal{C}}$. Let $\mathbf{p}_c = \mathbf{p}_c(\mathbf{g}, c)$ and $\mathbf{O}_c = \mathbf{O}_c(\mathbf{g})$ denote the contact position and orientation obtained via forward kinematics, and define the grasp map $\mathbf{G}_c = \begin{bmatrix} \mathbf{I} \\ (\mathbf{p}_c - \mathbf{m})_{\times} \end{bmatrix} \mathbf{O}_c$. Let $d_M(\cdot)$ be the signed distance to the object surface. Given target wrenches $\{\mathbf{w}_j\}_{j=1}^J$, a scaling factor $\lambda > 0$, and weights $\kappa_{\bullet} > 0$, grasp synthesis for bimanual robots is formulated as:

$$\min_{\mathbf{g}, \{\mathbf{f}_c\}} \kappa_w \sum_{j=1}^J \left\| \lambda \mathbf{w}_j - \sum_{c \in \mathcal{C}} \mathbf{G}_c(\mathbf{g}) \mathbf{f}_c \right\|_2^2 + \kappa_{\text{con}} \sum_{c \in \mathcal{C}} \psi(d_M(\mathbf{p}_c)) + \kappa_{\text{coll}} \Phi_M(\mathbf{g}) + \kappa_{\text{hh}} \Phi_{\text{hh}}(\mathbf{g}) \quad (6)$$

$$\text{s.t. } \mathbf{q}_{h, \min} \leq \mathbf{q}_h \leq \mathbf{q}_{h, \max}, \quad h \in \{0, 1\}, \quad (7)$$

$$\mathbf{f}_c \in \mathcal{F}, \quad c \in \mathcal{C}, \quad (8)$$

$$\mathbf{R}_h \in SO(3), \quad h \in \{0, 1\}, \quad (9)$$

where $\psi(d)$ is a distance energy that measures distance between contact points on the hands and object surface, $\Phi_M(\mathbf{g})$ is a hand-object collision energy based on signed-distance penalties, and $\Phi_{\text{hh}}(\mathbf{g})$ is an inter-hand penetration energy derived from hand-hand signed distances.

Following BODex [22], we consider the above as a nonlinear bilevel program, where the lower level quadratic programming optimizes contact forces for each contact point

to realize the target wrench and the upper level updates the hand pose to reduce the error between target and achievable wrenches via gradient descent. cuRobo [38] and a GPU-based QP solver are utilized to solve the problem efficiently.

The optimization program is unified for various grasp strategies. To adapt to a specific strategy, we select different contact points on the dexterous hands, as illustrated in Fig. 3. A gallery of synthesized grasps is presented in Fig. 1.

For each object placed in the scene, we generate 500 candidate grasps. Through physical validation, we filter out physically implausible grasps. Next, we use cuRobo for inverse kinematics analysis to determine whether these grasps are reachable by bimanual robots, and perform collision checking to exclude grasps that result in collisions with objects other than the target. Finally, we calculate the $SE(3)$ distance between the poses of the remaining grasps and the current end-effector pose of the bimanual robots, ranking them accordingly and selecting the grasp with the shortest distance as the preferred grasp. Specifically, the $SE(3)$ distance between two poses \mathbf{T}_1 and \mathbf{T}_2 is defined as:

$$d(\mathbf{T}_1, \mathbf{T}_2) = \|\mathbf{t}_1 - \mathbf{t}_2\|_2 + \lambda \cdot d_{\text{rot}}(\mathbf{R}_1, \mathbf{R}_2), \quad (10)$$

where $\mathbf{t}_1, \mathbf{t}_2$ are the translation vectors, $\mathbf{R}_1, \mathbf{R}_2$ are the rotation matrices, and λ is a weighting factor. The rotation distance d_{rot} is given by:

$$d_{\text{rot}}(\mathbf{R}_1, \mathbf{R}_2) = \arccos \left(\frac{\text{trace}(\mathbf{R}_1^{-1} \mathbf{R}_2) - 1}{2} \right). \quad (11)$$

This selection strategy favors grasps that require minimal motion from the current configuration, which not only improves the efficiency of the grasping process but also leads to more natural and smooth robot movements.

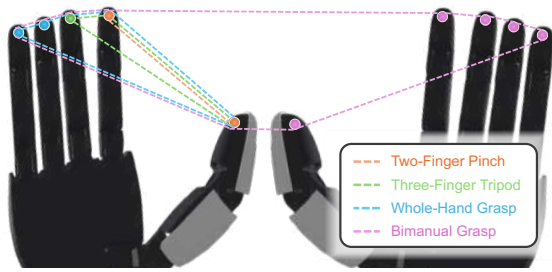


Fig. 3: **Hand contact points for various grasp strategies.** Different grasp strategies select distinct fingertip contact points, which are used to compute energy terms in the optimization process of grasp synthesis.

B. Demonstration Generation

After obtaining the preferred grasp, we divide the entire grasping process into four stages: pregrasp—the end-effector moves to a position 0.1 m away from the generated grasp pose in the direction opposite to the palm to avoid unintended collisions during approach; grasp—the hand reaches the preferred grasp pose; squeeze—the fingers apply pressure to the object to achieve a stable grasp; and lift—the object is lifted by 0.2 m. We employ bimanual motion planning to generate collision-free coordinated trajectories. To prevent policies trained on such data from exhibiting hesitation, we merge adjacent steps with negligible movements. In simulation, the robot executes the planned trajectory and grasp using PD control, and we verify whether the object is stably lifted. Specifically, the object must be raised at least 0.17 m above its initial pose and remain elevated for at least one second without being dropped. If the object is successfully lifted and no unexpected contacts or movements are detected, the trajectory is recorded and rendered. By generating demonstrations across a diverse set of objects and various grasp strategies, we construct UltraDexGrasp-20M, a large-scale, multi-strategy grasp dataset for bimanual robots, comprising 20 million frames over 1,000 objects.

It is noteworthy that, similar to [39], we supplement the robot’s point cloud during the rendering process with an additional imaged point cloud. During real-world deployment, the robot’s joint positions are known, enabling us to also use simulation to generate an imaged point cloud. This approach significantly reduces the sim-to-real gap.

V. UNIVERSAL DEXTEROUS GRASP POLICY

A. Overall Architecture

To accommodate multiple grasp strategies and improve generalization across a wide range of objects, we propose a universal dexterous grasp policy, as illustrated in Fig. 4. With simplicity and effectiveness in mind, we design the policy without redundant structures or auxiliary tasks. The policy receives scene point clouds as input, which are encoded into point features using a point encoder. The point features are processed by a decoder-only transformer architecture. Finally, action decoders predict control commands.

B. Point Cloud Encoding

We first downsample the point cloud using farthest point sampling (FPS) [40] to adjust the point cloud density. In practice, we use 2,048 points to balance computational cost and the granularity of scene representation. We then encode the resulting point cloud using a point encoder based on the PointNet++ [41] architecture. Specifically, we employ two set abstraction layers. The first layer does not perform downsampling and thus maintains 2,048 points. For each point, a group is formed by identifying its 32 nearest neighbors (including the point itself) using k-nearest neighbors algorithm (k-NN). Local features for each group are extracted via a series of 1×1 convolution, batch normalization, ReLU activation function, and max pooling. The second set abstraction layer has a similar structure but downsamples the 2,048 points to 256, thereby capturing higher-level features for action prediction.

C. Action Prediction

To read out action latents, a chunk of learnable action query tokens is fed into the transformer backbone, where they integrate scene information through unidirectional attention to the point features. Finally, a multi-layer perceptron (MLP) transforms the resulting action latents into action vectors to be executed by the robot. Rather than directly regressing the action vectors, our decoder predicts a bounded Gaussian distribution over actions via a truncated normal parameterization and optimizes the negative log-likelihood of the ground truth actions. Modeling actions probabilistically yields more stable training and enhances overall performance.

VI. EXPERIMENTS

We conduct comprehensive experiments in both simulation and real-world environments to evaluate the effectiveness of our data-generation pipeline for universal dexterous grasping, as well as the proposed universal dexterous grasp policy. Specifically, we aim to answer the following questions:

- 1) Does training on UltraDexGrasp-20M yield a policy with universal grasping capabilities and strong generalization across diverse objects?
- 2) How does the universal dexterous grasp policy perform compared to previous grasping methods and closed-loop control policies?
- 3) How does the policy perform as the amount of training data increases?
- 4) Do the key design components of the proposed policy effectively improve grasp success rates?
- 5) How does the policy trained on UltraDexGrasp-20M perform in real-world scenarios?

A. Simulation Experiments

1) *Experimental Setup:* We construct a dual-arm, dual-hand system composed of two 6-DoF UR5e robots and two 12-DoF XHand in simulation. The test set consists of 600 objects, including both seen and unseen categories during training. These objects exhibit significant variation in shape, weight, and size. The weight of the objects ranges from 5 g

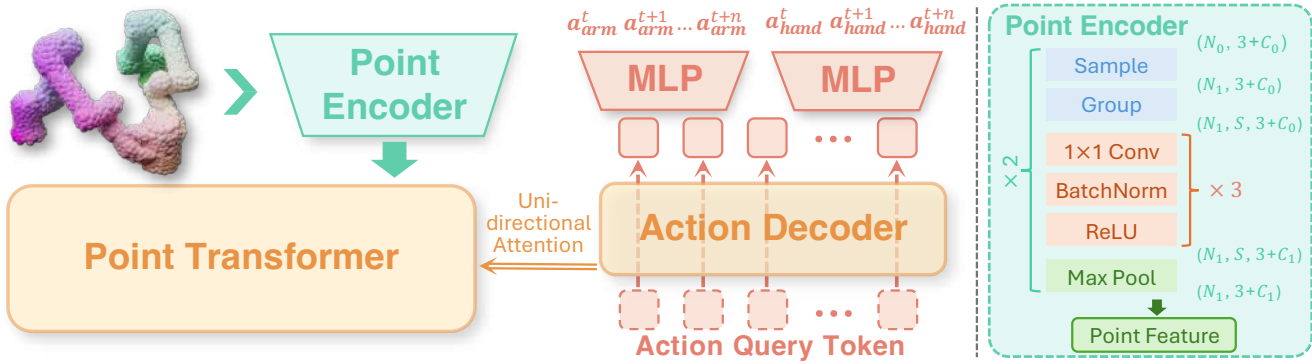


Fig. 4: **Overview of the policy architecture.** The proposed grasp policy takes point clouds as input, encodes them using a point encoder, aggregates scene features via unidirectional attention, and predicts control commands. The policy supports multiple grasp strategies and improves generalization across diverse objects.

to 1,000 g. The smallest object has a bounding box whose longest edge is less than 0.03 m, while the largest object has a bounding box whose shortest edge is greater than 0.5 m.

2) *Baselines:* To demonstrate the effectiveness of the proposed universal dexterous grasp policy, we select two strong baselines, DP3 [42] and DexGraspNet [3], for comparison. DP3 is a diffusion policy that takes point clouds and robot state as input and has shown strong performance in dexterous manipulation tasks. DexGraspNet takes the complete object mesh as input to generate grasp poses, and motion planning is utilized to obtain the full execution trajectory.

3) *Evaluation Metrics:* Each policy is evaluated on 600 objects, with 10 trials conducted for each object. The objects are categorized into three groups based on size: small, medium, and large. For small and medium items, placements are randomized within a 0.8 m × 0.2 m area, while large items are randomly placed in a 0.15 m × 0.16 m region to ensure reachability for the robot’s end-effectors. The success rate of each policy is reported.

TABLE I: **Results on simulation benchmarks.** The success rates (%) of each policy are reported. The proposed grasp policy trained on UltraDexGrasp-20M demonstrates strong generalization to diverse objects and consistently outperforms the baselines. The best results are highlighted in **bold**.

Benchmark	Object Size	DP3	DexGraspNet	Ours
Seen Objects	Small	41.7	45.6	78.8
	Medium	54.3	72.0	84.3
	Large	48.5	-	90.4
Unseen Objects	Small	37.4	45.6	76.9
	Medium	50.1	72.0	85.8
	Large	48.1	-	87.5
Average		46.7	58.8	84.0

4) *Main Results:* To answer Questions 1 and 2, we conduct experiments on the simulation benchmark. Both DP3 and our policy are trained on the UltraDexGrasp-20M dataset, while DexGraspNet, an optimization-based method, does not require training. As shown in Table I, our policy achieves an average success rate of 84.0%. It effectively

handles small, medium, and large objects. Notably, it demonstrates strong generalization with an 83.4% success rate on unseen objects. These results indicate that training on UltraDexGrasp-20M yields a policy with universal grasping capabilities and robust generalization across diverse objects. When trained on the same dataset, our policy outperforms DP3 by 37.3 percentage points, highlighting the superiority of the proposed grasp policy. We attribute this improvement to the carefully designed point encoder, which enables the policy to capture fine-grained object geometry, as well as the effective use of a unidirectional attention mechanism to aggregate scene features. DexGraspNet achieves an average success rate of 58.8% on small and medium objects, with significantly lower performance on small objects compared to medium ones. Additionally, DexGraspNet, which can only synthesize unimanual grasps, cannot handle large objects. These findings underscore the importance of multi-strategy dexterous grasping for bimanual robots.

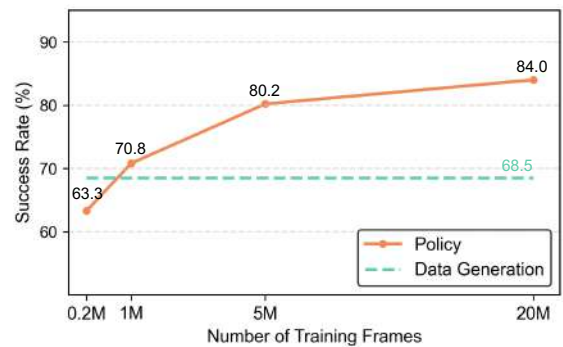


Fig. 5: **The variation in performance with growing amounts of training data.** The performance of the policy consistently improves as the volume of data increases.

5) *Scaling with Training Data:* To answer Question 3, we compare the average success rates of our policy across both seen and unseen objects, trained with different amounts of data. As shown in Fig. 5, the overall performance of the policy consistently improves as the amount of training data increases. For comparison, the average success rate of

grasping data generation is 68.5%. When training frames exceed 1M, the performance of the learned policy significantly surpasses that of data generation.

TABLE II: **Ablation study on policy design choices.** The success rates (%) are reported. Both the bounded Gaussian distribution prediction and the unidirectional attention mechanism significantly improve performance.

	w/o Dist. Pred.	w/o Uni. Attn.	Ours
Success Rate (%)	73.5	68.2	84.0

6) *Effectiveness of Design Choices:* To answer Question 4, we conduct ablation studies on key policy design choices. Specifically, we compare our policy with two ablated versions: one without bounded Gaussian distribution prediction (w/o Dist. Pred.) and another without the unidirectional attention mechanism (w/o Uni. Attn.), where all tokens input to the transformer decoder can freely attend to each other. As reported in Tab. II, our policy achieves more than a 10% improvement in success rate over both ablation baselines, demonstrating the effectiveness of our design choices.

B. Real-World Experiments

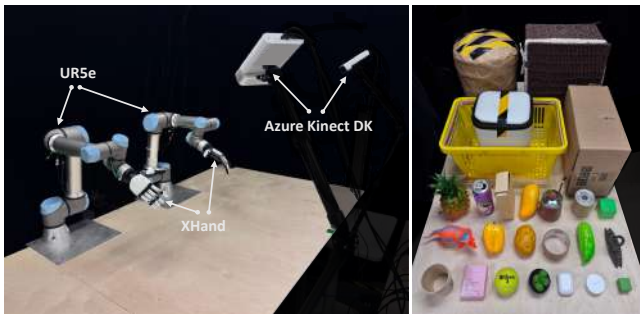


Fig. 6: **Real-world experiment setup.** We employ two UR5e robots, two XHand, and two eye-on-base Azure Kinect DK cameras. A variety of objects are collected for testing.

1) *Experimental Setup:* The real-world experimental setup is illustrated in Fig. 6. Two UR5e robots are placed on a table at a distance of 0.9 m apart. Each robot’s end-effector is equipped with a 12-DoF XHand, attached via a flange. Two Azure Kinect DK cameras are positioned on the tabletop to capture scene point clouds as input for the policy. The control frequency of the robotic system is set to 10 Hz.

2) *Sim-to-Real Implementation Details:* To facilitate sim-to-real transfer, we implement several procedures. We establish a consistent coordinate system for both domains and calibrate the intrinsic and extrinsic parameters of the real-world cameras to obtain point clouds aligned with those in the simulation. Since depth cameras are subject to noise, Statistical Outlier Removal (SOR) is applied to filter out outlier points. Following [39], imaged point clouds for robots are used to further mitigate the sim-to-real gap caused by low-quality, noisy, and incomplete point clouds obtained in the real world. Additionally, during demonstration generation,

joint impedance randomization is employed to reduce the dynamics gap between the real world and simulation.

3) *Benchmarks:* We evaluate the grasping performance of the policies using 25 objects, which are categorized into small, medium, and large based on their sizes. Each policy is tested 15 times on each object, with a different object pose for each trial. Consistent with the simulation benchmark, objects in the small and medium categories are randomly placed within a 0.8 m \times 0.2 m area, while large objects are randomly placed within a 0.15 m \times 0.16 m area. DP3 and DexGraspNet are chosen as baselines. For DexGraspNet, which requires object poses and complete object meshes, object poses are estimated using FoundationPose [43], and meshes are acquired through AR code scanning.

TABLE III: **Results on real-world benchmarks.** The success rates (%) are reported. The proposed grasp policy trained on UltraDexGrasp-20M demonstrates robust sim-to-real transfer and consistently outperforms the baselines.

Object Size	DP3	DexGraspNet	Ours
Small	37.3	51.3	72.0
Medium	56.0	73.3	82.2
Large	46.7	-	89.3
Average	46.7	62.3	81.2

4) *Main Results:* To answer Question 5, we directly deploy the proposed universal dexterous grasp policy, trained on UltraDexGrasp-20M, in real-world scenarios. The policy demonstrates robust zero-shot sim-to-real transfer and successfully handles novel objects of various shapes, sizes, and weights. Among the tested objects, the smallest has a volume of only 18 cm³, while the largest is 26,400 cm³; the lightest weighs only 3.6 g, and the heaviest is 1,095 g. Our policy effectively adapts grasping strategies to these diverse objects, exhibiting strategies such as three-finger tripod, whole-hand grasp, and bimanual grasp. As presented in Tab. III, our policy achieves robust grasps with an average success rate of 81.2%, significantly outperforming the baselines.

VII. CONCLUSION

In this paper, we introduce **UltraDexGrasp**, a framework for universal dexterous grasping with bimanual robots. We propose a data generation pipeline that features a complementary integration of optimization-based grasp synthesis and planning-based demonstration generation to produce universal dexterous grasp data for bimanual robots. With this pipeline, we curate **UltraDexGrasp-20M**, a large-scale multi-strategy grasp dataset, which contains 20 million frames across 1,000 objects. Tested on 600 diverse objects in simulation, our proposed policy achieves an average success rate of 84.0%. Deployed directly in the real world, the policy demonstrates robust zero-shot sim-to-real transfer.

REFERENCES

- [1] H.-S. Fang, C. Wang, H. Fang, M. Gou, J. Liu, H. Yan, W. Liu, Y. Xie, and C. Lu, “Anygrasp: Robust and efficient grasp perception in spatial and temporal domains,” *IEEE Transactions on Robotics*, vol. 39, no. 5, pp. 3929–3945, 2023.

- [2] H.-S. Fang, H. Yan, Z. Tang, H. Fang, C. Wang, and C. Lu, “Anydex-grasp: General dexterous grasping for different hands with human-level learning efficiency,” *arXiv preprint arXiv:2502.16420*, 2025.
- [3] 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,” *arXiv preprint arXiv:2210.02697*, 2022.
- [4] Y. Zhong, X. Huang, R. Li, C. Zhang, Y. Liang, Y. Yang, and Y. Chen, “Dexgraspvla: A vision-language-action framework towards general dexterous grasping,” *arXiv preprint arXiv:2502.20900*, 2025.
- [5] S. Deng, M. Yan, S. Wei, H. Ma, Y. Yang, J. Chen, Z. Zhang, T. Yang, X. Zhang, H. Cui, *et al.*, “Graspvla: A grasping foundation model pre-trained on billion-scale synthetic action data,” *arXiv preprint arXiv:2505.03233*, 2025.
- [6] H.-S. Fang, C. Wang, M. Gou, and C. Lu, “Graspnet-1billion: A large-scale benchmark for general object grasping,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 444–11 453.
- [7] Z. Wei, Z. Xu, J. Guo, Y. Hou, C. Gao, Z. Cai, J. Luo, and L. Shao, “D (r, o) grasp: A unified representation of robot and object interaction for cross-embodiment dexterous grasping,” *arXiv preprint arXiv:2410.01702*, 2024.
- [8] W. Wan, H. Geng, Y. Liu, Z. Shan, Y. Yang, L. Yi, and H. Wang, “Unidexgrasp++: Improving dexterous grasping policy learning via geometry-aware curriculum and iterative generalist-specialist learning,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 3891–3902.
- [9] T. G. W. Lum, M. Matak, V. Makovychuk, A. Handa, A. Allshire, T. Hermans, N. D. Ratliff, and K. Van Wyk, “Dextrah-g: Pixels-to-action dexterous arm-hand grasping with geometric fabrics,” *arXiv preprint arXiv:2407.02274*, 2024.
- [10] R. Singh, A. Allshire, A. Handa, N. Ratliff, and K. Van Wyk, “Dextrah-rgb: Visuomotor policies to grasp anything with dexterous hands,” *arXiv preprint arXiv:2412.01791*, 2024.
- [11] Z. Liang, Y. Mu, Y. Wang, T. Chen, W. Shao, W. Zhan, M. Tomizuka, P. Luo, and M. Ding, “Interaction-aware diffusion planning for adaptive dexterous manipulation,” in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 1745–1755.
- [12] 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, no. 1, pp. 470–477, 2021.
- [13] A. H. Li, P. Culbertson, J. W. Burdick, and A. D. Ames, “Frogger: Fast robust grasp generation via the min-weight metric,” in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 6809–6816.
- [14] 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.
- [15] Y. Xu, W. Wan, J. Zhang, H. Liu, Z. Shan, H. Shen, R. Wang, H. Geng, Y. Weng, J. Chen, *et al.*, “Unidexgrasp: Universal robotic dexterous grasping via learning diverse proposal generation and goal-conditioned policy,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4737–4746.
- [16] J. Ye, K. Wang, C. Yuan, R. Yang, Y. Li, J. Zhu, Y. Qin, X. Zou, and X. Wang, “Dex1b: Learning with 1b demonstrations for dexterous manipulation,” *arXiv preprint arXiv:2506.17198*, 2025.
- [17] H. Jiang, S. Liu, J. Wang, and X. Wang, “Hand-object contact consistency reasoning for human grasps generation,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 11 107–11 116.
- [18] A. T. Miller and P. K. Allen, “Graspit! a versatile simulator for robotic grasping,” *IEEE Robotics & Automation Magazine*, vol. 11, no. 4, pp. 110–122, 2004.
- [19] 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.
- [20] R. Zurbrugg, A. Cramariuc, and M. Hutter, “Graspqp: Differentiable optimization of force closure for diverse and robust dexterous grasping,” *arXiv preprint arXiv:2508.15002*, 2025.
- [21] P. Li, T. Liu, Y. Li, Y. Geng, Y. Zhu, Y. Yang, and S. Huang, “Gendexgrasp: Generalizable dexterous grasping,” *arXiv preprint arXiv:2210.00722*, 2022.
- [22] J. Chen, Y. Ke, and H. Wang, “Bodex: Scalable and efficient robotic dexterous grasp synthesis using bilevel optimization,” *arXiv preprint arXiv:2412.16490*, 2024.
- [23] W. Wei, D. Li, P. Wang, Y. Li, W. Li, Y. Luo, and J. Zhong, “Dvvg: Deep variational grasp generation for dextrous manipulation,” *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1659–1666, 2022.
- [24] C. Goldfeder, M. Ciocarlie, H. Dang, and P. K. Allen, “The columbia grasp database,” in *2009 IEEE international conference on robotics and automation*. IEEE, 2009, pp. 1710–1716.
- [25] S. Brahmabhatt, C. Tang, C. D. Twigg, C. C. Kemp, and J. Hays, “Contactpose: A dataset of grasps with object contact and hand pose,” in *European Conference on Computer Vision*, 2020, pp. 361–378.
- [26] Y.-W. Chao, W. Yang, Y. Xiang, P. Molchanov, A. Handa, J. Tremblay, Y. S. Narang, K. Van Wyk, *et al.*, “Dexycb: A benchmark for capturing hand grasping of objects,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 9044–9053.
- [27] Y. Liu, Y. Yang, Y. Wang, X. Wu, J. Wang, Y. Yao, S. Schwertfeger, S. Yang, *et al.*, “Realdex: Towards human-like grasping for robotic dexterous hand,” *arXiv preprint arXiv:2402.13853*, 2024.
- [28] Y. Shao and C. Xiao, “Bimanual grasp synthesis for dexterous robot hands,” *IEEE Robotics and Automation Letters*, 2024.
- [29] T. Lin, K. Sachdev, L. Fan, J. Malik, and Y. Zhu, “Sim-to-real reinforcement learning for vision-based dexterous manipulation on humanoids,” *arXiv preprint arXiv:2502.20396*, 2025.
- [30] A. Mousavian, C. Eppner, and D. Fox, “6-dof graspnet: Variational grasp generation for object manipulation,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 2901–2910.
- [31] Y. Zhong, Q. Jiang, J. Yu, and Y. Ma, “Dexgrasp anything: Towards universal robotic dexterous grasping with physics awareness,” in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 22 584–22 594.
- [32] L. Huang, H. Zhang, Z. Wu, S. Christen, and J. Song, “Fungrasp: functional grasping for diverse dexterous hands,” *IEEE Robotics and Automation Letters*, 2025.
- [33] J. He, D. Li, X. Yu, Z. Qi, W. Zhang, J. Chen, Z. Zhang, Z. Zhang, L. Yi, and H. Wang, “Dexvlg: Dexterous vision-language-grasp model at scale,” *arXiv preprint arXiv:2507.02747*, 2025.
- [34] J. Zhang, H. Liu, D. Li, X. Yu, H. Geng, Y. Ding, J. Chen, and H. Wang, “Dexgraspnet 2.0: Learning generative dexterous grasping in large-scale synthetic cluttered scenes,” in *8th Annual Conference on Robot Learning*, 2024.
- [35] T. G. W. Lum, A. H. Li, P. Culbertson, K. Srinivasan, A. D. Ames, M. Schwager, and J. Bohg, “Get a grip: Multi-finger grasp evaluation at scale enables robust sim-to-real transfer,” *arXiv preprint arXiv:2410.23701*, 2024.
- [36] H. Zhang, Z. Wu, L. Huang, S. Christen, and J. Song, “Robustdex-grasp: Robust dexterous grasping of general objects,” *arXiv preprint arXiv:2504.05287*, 2025.
- [37] Z. Chen, Q. Yan, Y. Chen, T. Wu, J. Zhang, Z. Ding, J. Li, Y. Yang, and H. Dong, “Clutterdexgrasp: A sim-to-real system for general dexterous grasping in cluttered scenes,” *arXiv preprint arXiv:2506.14317*, 2025.
- [38] B. Sundaralingam, S. K. S. Hari, A. Fishman, C. Garrett, K. V. Wyk, V. Blukis, A. Millane, H. Oleynikova, A. Handa, F. Ramos, N. Ratliff, and D. Fox, “curobo: Parallelized collision-free minimum-jerk robot motion generation,” 2023.
- [39] Y. Qin, B. Huang, Z.-H. Yin, H. Su, and X. Wang, “Dexpoint: Generalizable point cloud reinforcement learning for sim-to-real dexterous manipulation,” in *Conference on Robot Learning*. PMLR, 2023, pp. 594–605.
- [40] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, “Pointnet: Deep learning on point sets for 3d classification and segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.
- [41] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, “Pointnet++: Deep hierarchical feature learning on point sets in a metric space,” *Advances in neural information processing systems*, vol. 30, 2017.
- [42] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu, “3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations,” in *Proceedings of Robotics: Science and Systems (RSS)*, 2024.
- [43] B. Wen, W. Yang, J. Kautz, and S. Birchfield, “Foundationpose: Unified 6d pose estimation and tracking of novel objects,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 17 868–17 879.