ClutterDexGrasp: A Sim-to-Real System for General Dexterous Grasping in Cluttered Scenes

Zeyuan Chen[1,2*], **Qiyang Yan**[1,2*], **Yuanpei Chen**[1,3*]
Tianhao Wu[1,2†], **Jiyao Zhang**[1,2†], **Zihan Ding**[4], **Jinzhou Li**[1,2], **Yaodong Yang**[1,3], **Hao Dong**[1,2‡]

[1]CFCS, School of Computer Science, Peking University
[2]PKU-AgiBot Lab, [3]PKU-PsiBot Lab, [4]Princeton University

[*]Equal Contribution, [†]Project leader, [‡]Corresponding author

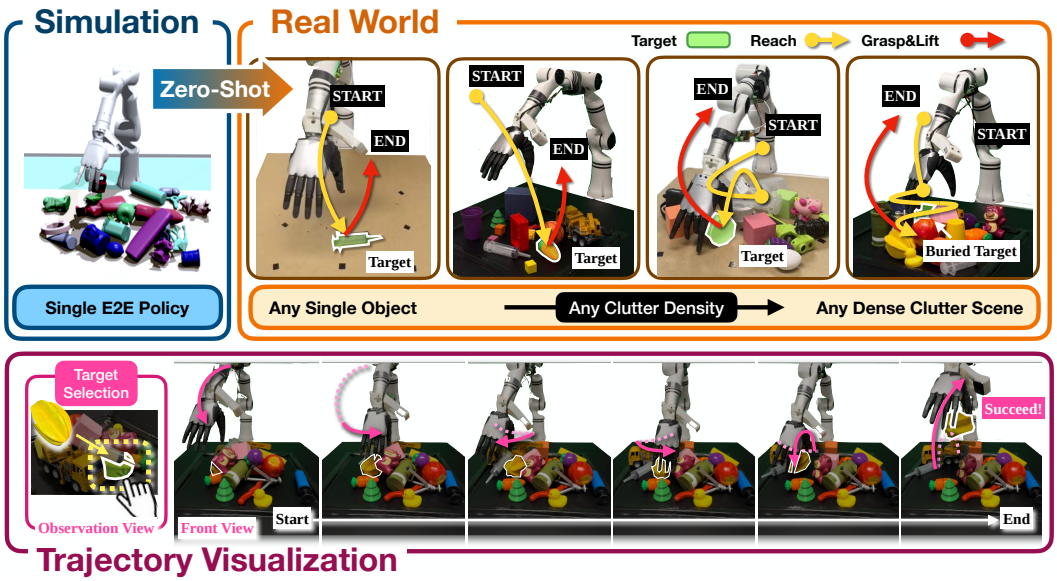

Figure 1: ClutterDexGrasp achieves zero-shot sim-to-real transfer for closed-loop target-oriented dexterous grasping in cluttered scenes, enabling robust generalization across diverse objects and cluttered scenes, even with severe object occlusion.

Abstract: Dexterous grasping in cluttered scenes presents significant challenges due to diverse object geometries, occlusions, and potential collisions. Existing methods primarily focus on single-object grasping or grasp-pose prediction without interaction, which are insufficient for complex, cluttered scenes. Recent vision-language-action models offer a potential solution but require extensive real-world demonstrations, making them costly and difficult to scale. To address these limitations, we revisit the sim-to-real transfer pipeline and develop key techniques that enable zero-shot deployment in reality while maintaining robust generalization. We propose ClutterDexGrasp, a two-stage teacher-student framework for closed-loop target-oriented dexterous grasping in cluttered scenes. The framework features a teacher policy trained in simulation using clutter density curriculum learning, incorporating both a geometry- and spatially-embedded scene representation and a novel comprehensive safety curriculum, enabling general, dynamic, and safe grasping behaviors. Through imitation learning, we distill the teacher's knowledge into a student 3D diffusion policy (DP3) that operates on partial point cloud observations. To the best of our knowledge, this represents the first zero-shot sim-to-real closed-loop system for target-oriented dexterous grasping in cluttered scenes, demonstrating robust performance across

9th Conference on Robot Learning (CoRL 2025), Seoul, Korea.

diverse objects and layouts. More details and videos are available at https://clutterdexgrasp.github.io/.

Keywords: Dexterous Grasping, Cluttered Scene, Sim-to-Real

1 Introduction

Humans can use their hands to grasp diverse objects in cluttered scenes [1], which serves as the foundation for performing subsequent manipulation tasks. Achieving such dexterity is critical for applying a dexterous hand in complex real-world environments [2, 3]. While current methods have made significant progress in generating grasp poses for single objects [4, 5, 6] or closed-loop grasping on single-object tabletop setups [7, 8, 9, 10, 11, 12], cluttered scenes present substantially different challenges. These scenes contain many objects with diverse geometries that are close together or even overlapping, leading to complex dynamics and occlusions during the grasping process [13, 14].

To enable dexterous grasping in cluttered scenes, some works first generate a non-collision grasp pose dataset specifically for cluttered scenes, then train a model to generate grasp poses and execute via motion planning [15, 14, 13]. However, such methods are open-loop and cannot handle unexpected changes. To achieve a closed-loop grasping policy, current methods rely on real-world imitation learning (IL)[16, 17, 18] or sim-to-real reinforcement learning (RL) [19, 20, 21]. However, real-world IL requires a large amount of human-collected data to generalize across diverse cluttered scenes[16, 22, 23], which is costly and lacks scalability. Current RL-based methods, which can achieve high success rates in the real world, are mainly designed for relatively simple scenes, such as single-object tabletop grasping [11, 12]. These methods, however, aren't directly applicable to cluttered scenes because they rely on simplified object states, like 6D poses, to ensure stable and convergent teacher policy learning, yet fail to capture the essential local geometric details crucial for dexterous grasping in cluttered scenes [24, 25]. To compactly represent the interaction between the object and the dexterous hand, [26] proposes a representation that computes the distance between the links of the dexterous hand and the graspable and ungraspable areas, which further improves teacher policy training [27].

In this paper, we aim to develop a generalized closed-loop policy for dexterous target-oriented grasping in cluttered scenes, trained in a simulation environment without expert demonstrations [28, 19], and to achieve zero-shot sim-to-real transfer [29, 30, 31]. The main challenges are as follows: (1) training instability in RL due to high-dimensional observation and action spaces caused by the diversity of object geometries and layouts, as well as the high DoF of the dexterous hand [19, 21, 20], (2) the sim-to-real gap in observation space, policy safety, and physical dynamics [32, 33, 34]. To address these challenges, we propose ClutterDexGrasp, a two-stage teacher-student framework. The teacher is trained in simulation using clutter density curriculum learning to reduce the difficulty of directly learning complex grasping strategies. Additionally, we leverage the distance representation [26] and extend it to clutterd scenes by computing distance between links of the dexterous hand and the target and non-target objects, such a geometry and spatially embedded scene representation captures the local geometry of the target object and its surrounding environment, enabling the teacher policy to effectively learn human-like grasping strategies in cluttered scenes [1]. To ensure safety during contact-rich interactions, we introduce an interaction-aware safety curriculum that minimizes collisions and avoids harmful behaviors, such as excessive force. The student policy is trained using offline imitation learning with a 3D diffusion policy (DP3) [35] on partial point cloud observations, enabling zero-shot sim-to-real transfer.

As shown in Fig. 1, ClutterDexGrasp achieves general, dynamic, and safe clutter target-oriented dexterous grasping in simulation while also demonstrating zero-shot sim-to-real transfer capability for the first time. Our results demonstrate that RL, combined with effective understanding, task representation, safety mechanisms, and efficient distillation [36], can enable robust general dexterous grasping in complex, cluttered environments and facilitate zero-shot sim-to-real transfer.

2 Related Works

Dexterous Grasping Dexterous grasping has progressed from single-object setups to complex cluttered environments [4, 5, 6, 3, 2]. Open-loop methods for cluttered scenes focus on grasp pose generation and planning [15, 14, 13, 4, 5], but lack adaptability. For dynamic control, closed-loop approaches using RL [7, 8, 9, 10, 11, 12, 37, 19, 21, 20, 38] achieve success in single-object scenes but face challenges in cluttered environments due to complex interactions and state representations [24, 25, 13, 14]. [26] and [27] introduce distance between links of the dexterous hand and graspable and ungraspable area of the target object as representation for RL training, with [27] demonstrating great performance on real robots using a teacher-student framework. However, their training is conducted exclusively on single objects, which limits their applicability to complex scenarios. In contrast, our approach directly incorporates cluttered scene representations during RL training, enabling enhanced performance in complex cluttered configurations, particularly under extreme stacking conditions. Alternately, IL-based methods [16, 17, 18, 39] require extensive demonstrations that are costly to collect for diverse cluttered scenarios [22, 23]. Recent advances using vision-language-action frameworks [16, 40] and motion capture [41] aim to reduce demonstration burden but still require significant human effort. Our approach addresses these challenges through RL with specialized representations and curriculum learning that can generalize to cluttered environments without real-world demonstrations.

Sim-to-Real Transfer with Reinforcement Learning Transferring RL policies from simulation to real-world remains challenging for dexterous manipulation [20, 34, 33, 32, 42]. RL enables learning complex behaviors that are hard to engineer manually [28, 43, 44], but introduces sim-to-real challenges due to distributional shifts in dynamics and observations [19, 21]. Techniques like domain randomization [29, 30] improve robustness by training across randomized simulation parameters, while system identification [31, 45] calibrates simulations to better match real-world dynamics. Safety-aware RL is critical for contact-rich behaviors [46, 47, 48, 42], especially in cluttered scenes with unpredictable object interactions. [33] and [32] address sim-to-real RL for dexterous manipulation, focusing on single-object tasks. Curriculum learning [38] structures training by gradually increasing task complexity while maintaining robustness and safety. Our method combines these strategies with an interaction-aware safety curriculum that progressively tightens constraints during training for safe real-world deployment.

Diffusion Policy and Teacher-Student Distillation Recent advances in generative modeling have introduced diffusion models for policy learning [49, 50, 51, 52, 53, 54, 55, 42], with significant potential for modeling complex action distributions. [35] proposed a 3D diffusion policy that operates directly on point cloud observations, making it well-suited for robotic manipulation tasks with partial observations. Policy distillation [36] transfers knowledge from a teacher policy with privileged information to a student policy with limited observation, which is particularly valuable for sim-to-real transfer. The teacher-student framework [3, 11], combined with diffusion policies, enables zero-shot sim-to-real transfer by distilling privileged simulation knowledge into policies that operate with real-world sensory input. Our approach leverages this paradigm, using a privileged teacher policy to generate demonstrations across varying clutter densities, which are then distilled into a point-cloud-based student policy capable of zero-shot transfer to real-world environments.

3 Problem Formulation

We formulate target-oriented dexterous grasping as an end-to-end learning problem, using RL to train a teacher policy and IL to train a student policy. The learned policies control both a robotic arm and a dexterous hand to grasp target objects in cluttered environments. For RL, we define a partially observable Markov decision process $(\mathscr{S}, \mathscr{O}, \mathscr{A}, R, \mathscr{T}, \mathscr{E}, \gamma)$ with state space $\mathscr{S}$, observation space $\mathscr{O}$, action space $\mathscr{A}$, reward function $R(s,a)$, transition function $\mathscr{T}(s'|s,a)$, and observation function $\mathscr{E}(o|s)$. A stochastic teacher policy $\pi^E(a|o)$ maps observations to actions, optimiz-

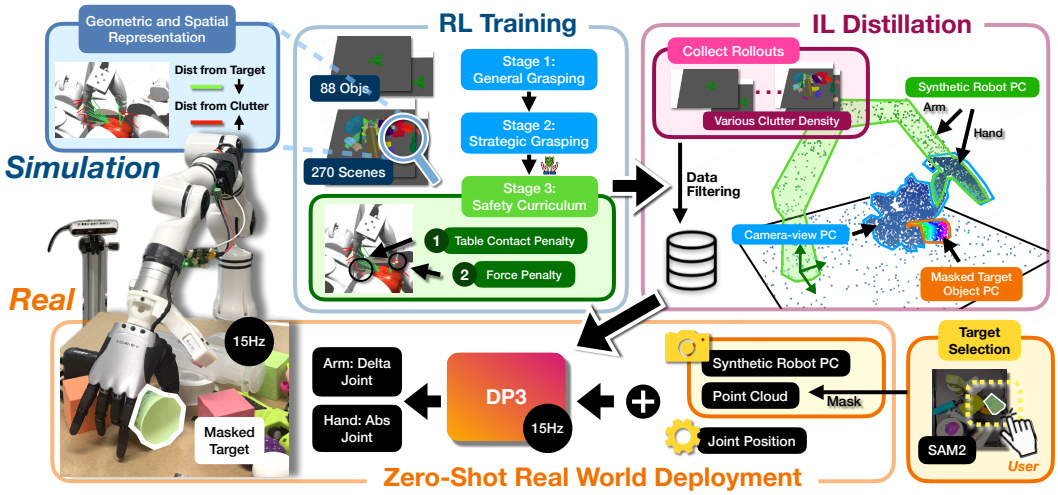

Figure 2: Training Framework

ing the discounted return $\mathbb{E}_\pi[\sum_{t=0}^{\infty}\gamma^t r(o_t,a_t)]$. For IL, we learn a student policy π^S by minimizing $\mathbb{E}_{(o,a)\sim\mathscr{D}_E}[-\log\pi(a|o)]$ on expert demonstrations $\mathscr{D}_E$ from π^E.

State and Action Spaces We consider a tabletop scenario with a 7-DoF robotic arm $\mathbf{J}^a\in\mathbb{R}^7$ and a 12-DoF dexterous hand $\mathbf{J}^h\in\mathbb{R}^{12}$. The hand joints consist of 6-DoF actuated finger joints $\mathbf{J}^f\in\mathbb{R}^6$, and 6-DoF underactuated finger joints $\mathbf{J}^u\in\mathbb{R}^6$. The action space $\mathscr{A}\subseteq\mathbb{R}^{13}$ encompasses 7D relative changes for the arm joint $\mathbf{a}^a$ and 6D absolute joint positions for the actuated hand joints $\mathbf{a}^h$ as [56].

Task Simulation For each grasping trial, we initially sample a random number of objects from an object prior distribution. The sampled objects are used to construct the cluttered scenes. Then the target object is randomly selected for grasping.

Observation The student observation space $\mathscr{O}=\mathbf{J}^a\times\mathbf{J}^f\times\mathscr{O}^{pc}$ consists of the robot joint positions $\mathbf{J}^a,\mathbf{J}^f$, and a point cloud observation $\mathscr{O}^{pc}\in\mathbb{R}^{4\times5120}$, which is generated from the robot's camera (Detailed in Appendix C.1). The teacher observation space $\mathscr{O}=\mathbf{J}^a\times\mathbf{J}^h\times\mathscr{O}^E$ with additional privileged observation $\mathscr{O}^E$, which includes the geometry and spatial representation (Sec.4.1.1), along with privileged state information (Detailed in Appendix B.2).

Objective For teacher policy, the RL objective is to find optimal policy $\pi(\mathbf{a}|o),o\sim\mathscr{O}$ that maximizes the expected discounted reward:

$$\pi^{E^*}=\arg\max_\pi\mathbb{E}_{\mathbf{a}_t\sim\pi(\cdot|o_t)}\left[\sum_{t=0}^{T}\gamma^t r(o_t,\mathbf{a}_t)\right] \tag{1}$$

$$\pi^{S^*}=\arg\min_\pi\mathbb{E}_{(o,\mathbf{a})\sim\mathscr{D}^E}[-\log\pi(\mathbf{a}|o)] \tag{2}$$

For student π^S it optimizes the IL objective with demonstration dataset $\mathscr{D}^E=\{(o,\mathbf{a})\},\mathbf{a}\sim\pi^E(\cdot|o)$ sampled by teacher policy. The above equations pose challenging objectives, since the policy needs to adapt to different objects and layouts.

4 Method

We adopt a teacher-student learning paradigm to develop a policy that is general, dynamic, and safe for dexterous grasping of target objects in cluttered scenes, as illustrated in Fig.2. The framework consists of two main components: (1) a teacher policy trained using privileged state information through our proposed geometry-spatial representation and curriculum learning strategies (*Clutter-Density* and *Safety*), detailed in Sec.4.1; and (2) a student policy distilled from successful teacher

demonstrations across varying clutter densities. It applies a 3D diffusion policy with point-cloud observations and sim-to-real transfer techniques, described in Sec.4.2. The entire training paradigm is conducted in simulation, with no real-world demonstrations used throughout the process.

4.1 Teacher Policy Learning in Simulation

To enable general, dynamic, and safe grasping behavior, we train a teacher policy in simulation using a three-stage curriculum learning strategy. The first two curriculum progressively increases task complexity to guide policy development. The final stage fine-tunes the policy with safety constraints to promote robust and safe execution during real-world deployment. We implement three stages using a unified RL formulation with PPO [57] and a novel geometry and spatial representation with privileged state information (Sec.4.1.1). Following the objective in Eq. (1), policies $\pi^E(a|\mathscr{O}) : \mathscr{O} \rightarrow \Pr(\mathscr{A})$ map privileged observations to actions. We detail each component below.

4.1.1 Geometry and Spatial Representation

Cluttered Scene Representation For each finger link, we compute the 3D distance vector to the nearest sampled points from both the target (d_{pos}) and non-target objects (d_{neg}), as illustrated in Fig. 9. This representation efficiently encodes both object geometry and spatial relationships between the hand and the clutter. It enables collision-aware grasping while maintaining a compact observation space that simplifies learning of complex geometric features and avoiding computationally expensive point-cloud rendering. Detailed implementation is provided in Appendix B.1.

Dense Reward Function The reward function with embedded representation at each timestep is:

$$r = (c_1 \cdot r_{\text{grasp}} + c_2 \cdot r_{pos}) \cdot (1 - r_{neg}) \tag{3}$$

where $c_1, c_2 > 0$ are weighting coefficients. Here, r_{pos} provides a reward term encouraging proximity to the target, while r_{neg} imposes a penalty for risky closeness to non-target objects, and r_{grasp} represents the base grasping reward (detailed in Appendix B.3).

4.1.2 Two-Stage Clutter-Density Curriculum

As the first two stages of curriculum learning, we introduce the *Clutter-Density Curriculum*. Given the complexity of learning general grasping in cluttered scenes, where intricate interactions and robust grasping are required, learning entirely from scratch is found to be infeasible as shown in Fig. 8. To tackle this, we decompose the learning process into two stages: (1) learning robust, generalizable grasping policy for single-object scenarios, and (2) fine-tuning the policy in cluttered environments to learn contact-rich, strategic grasping. This curriculum allows the agent to first efficiently master basic grasping skills for single objects before learning more complex behaviors required for cluttered scenes, such as moving obstacle objects away before grasping the target object.

4.1.3 Interaction Safety Curriculum

As the final stage, we introduce an interaction *Safety Curriculum* to ensure safe and robust grasping in cluttered scenes. This curriculum is crucial for sim-to-real transfer, tempering overly aggressive behaviors while preserving human-like strategies. It is designed to minimize collisions and excessive force during contact-rich interactions common in cluttered environments. Specifically, the policy is fine-tuned by progressively tightening a force threshold $\bar{f}$ whenever the success rate w surpasses a threshold $\bar{w}$. Exceeding this force threshold incurs a sparse penalty term r_{force} in the reward and leads to early termination. The reward function Eq. (7) is updated with safety penalties as:

$$r_{\text{safe}} = r - c_3 \cdot r_{\text{force}}, \tag{4}$$

where $c_3 > 0$ are weighting coefficients. To avoid colliding with table, particularly when grasping small objects, we also disabled collisions between the hand and the table in simulation, and terminate the episode if the fingertip penetrates too deeply into the table surface. The implementation details of safety curriculum are deferred to Appendix B.4.

4.2 Student Point-Cloud Policy Distillation

To enable real-world target-oriented dexterous grasping in cluttered scenes, we distill the teacher policy into a student policy that operates on partial point cloud observations from a single camera. The student policy is trained offline using demonstrations collected across multiple clutter densities (Sec.4.2.1). To bridge the sim-to-real gap, we employ point cloud alignment for perception (Sec.4.2.2) and system identification for dynamics (Sec.4.2.3).

4.2.1 Multi-Level Clutter Density Demonstration

The student policy should exhibit adaptive behaviors based on scene clutter density. In sparse scenes, direct grasping is preferred, while in dense scenes, the target-oriented policy must first clear obstacles before grasping. To capture this full spectrum of behaviors, we collect expert demonstrations from the teacher policy π^E across varying clutter densities, creating a balanced dataset $\mathscr{D}^E$ for distillation. To model these diverse and complex behaviors, we leverage Diffusion Policy [49], which has proven effective for high-dimensional dexterous manipulation [58, 59]. Specifically, we use DP3 [35] as our backbone for processing point cloud observations. The student policy π^S is optimized by minimizing the negative log likelihood on the expert demonstrations as Eq. (2). The details are deferred to Appendix C.2.

4.2.2 Point Cloud Alignment

To address the sim-to-real perception gap and handle occlusions in cluttered scenes, we augment the observed point cloud with synthetic, dense robot point clouds as shown in Fig.10. Similar to Dexpoint[7], our augmentation includes both arm and hand point clouds, which helps establish critical spatial relationships for accurate grasping and collision avoidance. The details are deferred to Appendix C.3.

4.2.3 System Identification for Sim-to-Real Transfer

To minimize the dynamics gap between simulation and reality, we perform system identification (SI) [31, 60] to calibrate the physical parameters of both the robotic arm and hand. The SI process iteratively compares trajectories from simulation and the real world under identical commands, optimizing parameters until behaviors closely match [34], detailed in Appendix C.4.

5 Experiments

We conduct comprehensive experiments in both sim and real to validate the following questions:

- Can our framework produce robust and safe policy with high success rates in simulation?
- Can our policy be trained in simulation, zero-shot transfer to the real world?
- Does the policy generalize across unseen objects and unseen layouts?

5.1 Simulation Experiments

Experiment Setup The object datasets consist of 88 training objects from GraspNet1Billion [61] and 2029 testing objects from Omni6DPose [62]. The training set is used to generate 270 training scenes. To evaluate across clutter levels, we create three scene types: sparse ($N_{\text{object}} \in [4, 8]$), dense ($N_{\text{object}} \in [9, 15]$), and ultra-dense ($N_{\text{object}} \in [16, 25]$). For evaluation, we generate 500 unseen layouts using training objects with a sparse:dense:ultra-dense ratio of 3:4:3. For unseen objects, we generate 550 unseen layouts following a 6:3:2 ratio. In each trial, a random visible object is selected as the grasp target. All experiments use three random seeds, and we report the mean and standard deviation of the **Success Rate**: a trial is successful if the target object is lifted 0.1 meters. Simulation environment details are in Appendix D.

Dataset	Policy	Seen Layouts	Unseen Layouts		
			Sparse	Dense	Ultra-dense
GraspNet1Billion (Seen)	Teacher	91.9 ± 0.3	92.5 ± 0.8	87.5 ± 0.3	80.9 ± 0.2
	Student	87.0 ± 0.3	89.7 ± 0.5	83.4 ± 0.3	73.5 ± 0.5
Omni6DPose (Unseen)	Teacher	/	92.6 ± 0.4	86.6 ± 0.4	81.6 ± 0.3
	Student	/	90.8 ± 0.7	82.1 ± 1.6	74.2 ± 1.8

Table 1: **Simulation Success Rate of Random Object Grasping**. The unseen clutter scenes are classified into three-density-level: Sparse, Dense, Ultra-dense.

5.1.1 Main Results

Generalization Across Diverse Clutter Density and Novel Scenes As shown in Tab. 1, our teacher policy achieves a 91.9% success rate on seen objects and seen layouts. For seen objects with unseen layouts, the teacher policy generalizes well across different clutter densities and achieves comparable results even on unseen objects and unseen layouts, demonstrating the strong generalization capability of our method. The student policy, distilled to partial point cloud observation, shows less than a 5% average success rate drop across all combinations of seen and unseen objects and layouts, highlighting the effectiveness of our distillation framework. Importantly, both policies were trained only on scenes with sparse and dense clutter levels, yet they generalize well to much harder ultra-dense clutter scenes that were not encountered during training.

Human-like Clutter Scene Grasping Strategy Quantitative and qualitative results show that the policy learns a human-like grasping strategy for cluttered scenes in both simulation and the real world. Specifically: (1) *Clutter Clearance via Gentle Interaction*: As shown in Fig.1, when the target object is deeply buried and direct access is obstructed, the robot gently nudges overlying objects instead of pushing forcefully. As shown in Tab.3, our policy applies low contact force during interaction, further indicating gentle behavior. (2) *Clutter-aware Grasp*: As shown in Fig.3, once the obstacles are partially removed, the hand begins to approach from the side to grasp the target object. Note that these behaviors emerge automatically, without heuristic mode identification or switching, enabling effective, adaptive, and collision-minimized grasping.

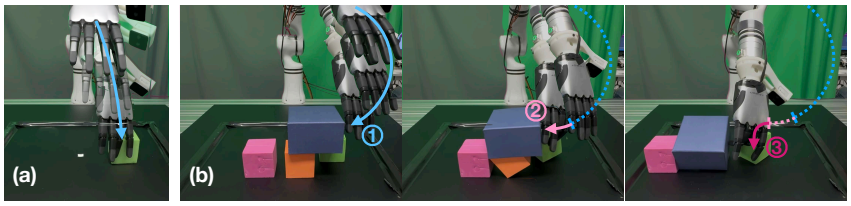

Figure 3: Visualization of Human-like Grasping Strategy: (a) Efficient grasping in simple scenes. (b) Clutter-aware grasping in cluttered scenes.

5.1.2 Effectiveness of Density Curriculum and Safety Curriculum

To evaluate the necessity of the *Density Curriculum*, we conduct an experiment by directly training the policy on cluttered scenes. Under the same training duration, training directly in clutter results in complete failure (0% success) due to complex interactions, whereas initializing from the general grasp policy achieves 87.0% success rate—demonstrating the effectiveness of progressive curriculum learning. The learning curve can be found in Fig. 8.

	Success Rate	Force (unit)
Ours	87.0 ± 0.3	43.2 ± 0.7
w/o Safety	88.9 ± 0.3	80.6 ± 1.9

Table 2: **Ablation**: test on unseen layouts with unseen objects.

To evaluate the *Safety Curriculum*, we measure the average maximum contact forces (in Isaac Gym [63] default force units) across all evaluation trajectories. As shown in Fig. 6, although the teacher policy without safety training achieves slightly higher success rate (1.9%), it exhibits excessive force due to risky actions such as poking, jabbing, or squeezing the object, making it unsafe

for deployment (Fig. 7). With the safety curriculum, the policy performs more gentle interactions, significantly reduces the force, and shows lower variance (Tab. 2). Crucially, the curriculum filters out overly dynamic, high-risk behaviors—retaining only those that are both human-like and suitable for sim-to-real transfer.

5.2 Real-world Experiments

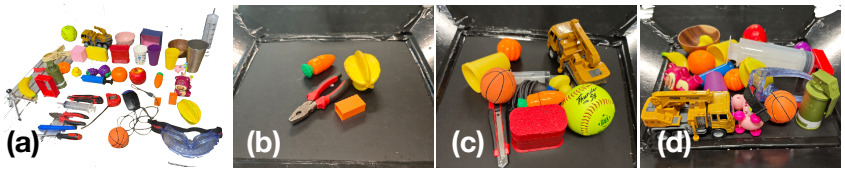

Figure 4: Real-world Objects and Example of Cluttered Scenes: (a): real-world object datasets, (b): sparse cluttered scene, (c) dense cluttered scene, (d) ultra-dense cluttered scene.

Experiment Setup We chose 41 objects with diverse shapes, sizes, and materials, as shown in Fig.4. Following the simulation experiments setup, we also conduct grasping under three level clutter density: 9 sparse scenes, 5 dense scenes, and 3 extreme scenes, as shown in Fig.4. Ensuring that the entire object dataset is covered across all levels of clutter. For each scene, the policy keeps grasping until three consecutive failures happen. For each grasping attempt, the target object is randomly selected from visible masks, increasing difficulty by requiring interaction with occluding clutter. The policy runs at 15 Hz on the real robot. We report **Success Rate** as $N_{\text{successfully grasped objects}}/N_{\text{total attempts}}$ and the **Area under the Curve (AUC)** to evaluate the efficiency of our system.

5.2.1 Main Results

Our sim-to-real strategy demonstrates strong real-world performance across varying levels of clutter, achieving an overall 83.9% success rate on unseen layouts with unseen objects over 167 grasping attempts, without any cluttered scene being early-stopped due to three consecutive failures. This performance is comparable to simulation results, confirming the effectiveness of our transfer pipeline. The policy generalizes robustly to a wide range of object geometries and sizes—including irregular and challenging shapes such as a Realsense tripod—without any real-world training. As shown in Fig.3, the student policy reproduces human-like, contact-aware strategies similar to those seen in simulation. Furthermore, Fig.5 illustrates that most of successful grasps are completed within 30 seconds, even in dense clutter, underscoring the system's efficiency. The cumulative success curve reaches 80% within 40 seconds, demonstrating that the policy reliably achieves high success rates given sufficient time.

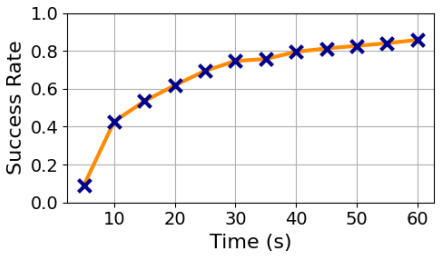

Figure 5: Real-world Experiment

Method	SR@20s	SR@40s	SR@60s	AUC
Ours	61.8	79.5	83.9	0.617

Table 3: Quantitative evaluation results in the real world.

6 Conclusion

In this paper, we introduce ClutterDexGrasp, a framework designed for dexterous target-oriented grasping in cluttered environments. Utilizing a two-stage teacher-student framework, we train a general, dynamic, and safe teacher policy in simulation with geometry- and spatially-embedded scene representation, via clutter density and safety curriculum, and transfer it to the point cloud-based student policy using a 3D diffusion policy with several sim-to-real strategies. Experimental results demonstrate that the method performs well in simulation and can be zero-shot transferred to real-world environments, effectively generalizing to various objects and layouts for the first time.

7 Limitation

Although our method achieves a high success rate, it still faces limitations with tiny objects due to imprecise grasps caused by the sim-to-real gap and frequent occlusions by the hand and surrounding objects. This highlights the need for enhanced perception strategies—such as multi-view fusion or active vision—to improve visibility and grasp planning in cluttered real-world scenes.

Acknowledgments

We thank Tianyu Wang, Hongwei Fan and Hang Dai for their support in conducting real-world experiments, Hongjie Fang and Tyler Ga Wei Lum for their insightful discussion. We are also grateful to Yanzhou Jin and Haotian Jin for their assistance with the hardware in this project. This research was supported by The National Youth Talent Support Program (8200800081) and National Natural Science Foundation of China (62376006).

References

[1] I. M. Bullock and A. M. Dollar. Classifying human manipulation behavior. In *2011 IEEE international conference on rehabilitation robotics*, pages 1–6. IEEE, 2011.

[2] A. I. Weinberg, A. Shirizly, O. Azulay, and A. Sintov. Survey of learning approaches for robotic in-hand manipulation. *arXiv preprint arXiv:2401.07915*, 2024.

[3] T. Chen, J. Xu, and P. Agrawal. A system for general in-hand object re-orientation. *CoRR*, abs/2111.03043, 2021. URL https://arxiv.org/abs/2111.03043.

[4] G.-H. Xu, Y.-L. Wei, D. Zheng, X.-M. Wu, and W.-S. Zheng. Dexterous grasp transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17933–17942, 2024.

[5] J. Lu, H. Kang, H. Li, B. Liu, Y. Yang, Q. Huang, and G. Hua. Ugg: Unified generative grasping. In *European Conference on Computer Vision*, pages 414–433. Springer, 2024.

[6] Y. Zhong, Q. Jiang, J. Yu, and Y. Ma. Dexgrasp anything: Towards universal robotic dexterous grasping with physics awareness. *arXiv preprint arXiv:2503.08257*, 2025.

[7] Y. Qin, B. Huang, Z.-H. Yin, H. Su, and X. Wang. Dexpoint: Generalizable point cloud reinforcement learning for sim-to-real dexterous manipulation. *Conference on Robot Learning (CoRL)*, 2022.

[8] Y.-H. Wu, J. Wang, and X. Wang. Learning generalizable dexterous manipulation from human grasp affordance. In *Conference on Robot Learning*, pages 618–629. PMLR, 2023.

[9] 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*, pages 3891–3902, 2023.

[10] Z. Huang, H. Yuan, Y. Fu, and Z. Lu. Efficient residual learning with mixture-of-experts for universal dexterous grasping. *arXiv preprint arXiv:2410.02475*, 2024.

[11] T. G. W. Lum, M. Matak, V. Makoviychuk, 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. In *8th Annual Conference on Robot Learning*.

[12] 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.

[13] H.-S. Fang, H. Yan, Z. Tang, H. Fang, C. Wang, and C. Lu. Anydexgrasp: Learning general dexterous grasping for any hands with human-level learning efficiency. In *7th Robot Learning Workshop: Towards Robots with Human-Level Abilities*.

[14] 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.

[15] W. Wei, D. Li, P. Wang, Y. Li, W. Li, Y. Luo, and J. Zhong. Dvgg: Deep variational grasp generation for dextrous manipulation. *IEEE Robotics and Automation Letters*, 7(2):1659–1666, 2022.

[16] 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.

[17] M. Zare, P. M. Kebria, A. Khosravi, and S. Nahavandi. A survey of imitation learning: Algorithms, recent developments, and challenges, 2023. URL https://arxiv.org/abs/2309.02473.

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

[19] A. Rajeswaran, V. Kumar, A. Gupta, J. Schulman, E. Todorov, and S. Levine. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. *CoRR*, abs/1709.10087, 2017. URL http://arxiv.org/abs/1709.10087.

[20] O. M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, et al. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020.

[21] H. Zhu, A. Gupta, A. Rajeswaran, S. Levine, and V. Kumar. Dexterous manipulation with deep reinforcement learning: Efficient, general, and low-cost. *CoRR*, abs/1810.06045, 2018. URL http://arxiv.org/abs/1810.06045.

[22] S. Ross, G. Gordon, and D. 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.

[23] S. Ross, G. Gordon, and D. 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.

[24] A. Nagabandi, K. Konolige, S. Levine, and V. Kumar. Deep dynamics models for learning dexterous manipulation. *CoRR*, abs/1909.11652, 2019. URL http://arxiv.org/abs/1909.11652.

[25] V. Kumar, E. Todorov, and S. Levine. Optimal control with learned local models: Application to dexterous manipulation. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 378–383, 2016. doi:10.1109/ICRA.2016.7487156.

[26] H. Zhang, S. Christen, Z. Fan, O. Hilliges, and J. Song. Graspxl: Generating grasping motions for diverse objects at scale. In *European Conference on Computer Vision*, pages 386–403. Springer, 2024.

[27] H. Zhang, Z. Wu, L. Huang, S. Christen, and J. Song. Robustdexgrasp: Robust dexterous grasping of general objects. *arXiv preprint arXiv:2504.05287*, 2025.

[28] M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, O. Pieter Abbeel, and W. Zaremba. Hindsight experience replay. *Advances in neural information processing systems*, 30, 2017.

[29] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. *CoRR*, abs/1703.06907, 2017. URL http://arxiv.org/abs/1703.06907.

[30] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 23–30. IEEE, 2017.

[31] L. Ljung. Perspectives on system identification. *Annual Reviews in Control*, 34(1):1–12, 2010. ISSN 1367-5788. doi:https://doi.org/10.1016/j.arcontrol.2009.12.001. URL https://www.sciencedirect.com/science/article/pii/S1367578810000027.

[32] A. Maddukuri, Z. Jiang, L. Y. Chen, S. Nasiriany, Y. Xie, Y. Fang, W. Huang, Z. Wang, Z. Xu, N. Chernyadev, et al. Sim-and-real co-training: A simple recipe for vision-based robotic manipulation. *arXiv preprint arXiv:2503.24361*, 2025.

[33] 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.

[34] E. Valassakis, Z. Ding, and E. Johns. Crossing the gap: A deep dive into zero-shot sim-to-real transfer for dynamics. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5372–5379. IEEE, 2020.

[35] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu. 3d diffusion policy. *arXiv e-prints*, pages arXiv–2403, 2024.

[36] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network, 2015. URL https://arxiv.org/abs/1503.02531.

[37] Z. Ding, Y. Chen, A. Z. Ren, S. S. Gu, Q. Wang, H. Dong, and C. Jin. Learning a universal human prior for dexterous manipulation from human preference. *arXiv preprint arXiv:2304.04602*, 2023.

[38] OpenAI, I. Akkaya, M. Andrychowicz, M. Chociej, M. Litwin, B. McGrew, A. Petron, A. Paino, M. Plappert, G. Powell, R. Ribas, J. Schneider, N. Tezak, J. Tworek, P. Welinder, L. Weng, Q. Yuan, W. Zaremba, and L. Zhang. Solving rubik's cube with a robot hand. *CoRR*, abs/1910.07113, 2019. URL http://arxiv.org/abs/1910.07113.

[39] Y. Qin, H. Su, and X. Wang. From one hand to multiple hands: Imitation learning for dexterous manipulation from single-camera teleoperation. *IEEE Robotics and Automation Letters*, 7(4): 10873–10881, 2022.

[40] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, et al. *pi_0*: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

[41] R. Ding, Y. Qin, J. Zhu, C. Jia, S. Yang, R. Yang, X. Qi, and X. Wang. Bunny-visionpro: Real-time bimanual dexterous teleoperation for imitation learning, 2024. URL https://arxiv.org/abs/2407.03162.

[42] Q. Yan, Z. Ding, X. Zhou, and A. J. Spiers. Variable-friction in-hand manipulation for arbitrary objects via diffusion-based imitation learning. *arXiv preprint arXiv:2503.02738*, 2025.

[43] P. Abbeel and A. Y. Ng. Exploration and apprenticeship learning in reinforcement learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 1–8, 2005.

[44] Z. Ding, Y.-Y. Tsai, W. W. Lee, and B. Huang. Sim-to-real transfer for robotic manipulation with tactile sensory. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6778–6785. IEEE, 2021.

[45] L. Huang, H. Zhang, Z. Wu, S. Christen, and J. Song. Fungrasp: Functional grasping for diverse dexterous hands. *arXiv preprint arXiv:2411.16755*, 2024.

[46] L. Sievers, J. Pitz, and B. Bäuml. Learning purely tactile in-hand manipulation with a torque-controlled hand. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 2745–2751, 2022. doi:10.1109/ICRA46639.2022.9812093.

[47] G. Narayanan, J. A. Raj, A. Gandhi, A. A. Gupte, A. J. Spiers, and B. Calli. Within-hand manipulation planning and control for variable friction hands. In *Experimental Robotics: The 17th International Symposium*, pages 600–610. Springer, 2021.

[48] A. Sahin, A. J. Spiers, and B. Calli. Region-based planning for 3d within-hand-manipulation via variable friction robot fingers and extrinsic contacts. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6549–6555. IEEE, 2021.

[49] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. *arXiv preprint arXiv:2303.04137*, 2023.

[50] Z. Wang, J. J. Hunt, and M. Zhou. Diffusion policies as an expressive policy class for offline reinforcement learning. *arXiv preprint arXiv:2208.06193*, 2022.

[51] T. Pearce, T. Rashid, A. Kanervisto, D. Bignell, M. Sun, R. Georgescu, S. V. Macua, S. Z. Tan, I. Momennejad, K. Hofmann, et al. Imitating human behaviour with diffusion models. *arXiv preprint arXiv:2301.10677*, 2023.

[52] M. Reuss, M. Li, X. Jia, and R. Lioutikov. Goal-conditioned imitation learning using score-based diffusion policies. *arXiv preprint arXiv:2304.02532*, 2023.

[53] A. Ajay, Y. Du, A. Gupta, J. Tenenbaum, T. Jaakkola, and P. Agrawal. Is conditional generative modeling all you need for decision-making? *arXiv preprint arXiv:2211.15657*, 2022.

[54] M. Janner, Y. Du, J. B. Tenenbaum, and S. Levine. Planning with diffusion for flexible behavior synthesis. *arXiv preprint arXiv:2205.09991*, 2022.

[55] Z. Ding and C. Jin. Consistency models as a rich and efficient policy class for reinforcement learning. *arXiv preprint arXiv:2309.16984*, 2023.

[56] B. Huang, Y. Chen, T. Wang, Y. Qin, Y. Yang, N. Atanasov, and X. Wang. Dynamic handover: Throw and catch with bimanual hands, 2023. URL https://arxiv.org/abs/2309.05655.

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

[58] T. Wu, M. Wu, J. Zhang, Y. Gan, and H. Dong. Learning score-based grasping primitive for human-assisting dexterous grasping. *Advances in Neural Information Processing Systems*, 36: 22132–22150, 2023.

[59] T. Wu, Y. Gan, M. Wu, J. Cheng, Y. Yang, Y. Zhu, and H. Dong. Unidexfpm: Universal dexterous functional pre-grasp manipulation via diffusion policy. *arXiv e-prints*, pages arXiv–2403, 2024.

[60] T. Chen, M. Tippur, S. Wu, V. Kumar, E. Adelson, and P. Agrawal. Visual dexterity: In-hand reorientation of novel and complex object shapes. *Science Robotics*, 8(84):eadc9244, 2023.

[61] 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*, pages 11444–11453, 2020.

[62] J. Zhang, W. Huang, B. Peng, M. Wu, F. Hu, Z. Chen, B. Zhao, and H. Dong. Omni6dpose: A benchmark and model for universal 6d object pose estimation and tracking. In *European Conference on Computer Vision*, pages 199–216. Springer, 2024.

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

[64] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer. Sam 2: Segment anything in images and videos, 2024. URL https://arxiv.org/abs/2408.00714.

A Ablation Study

A.1 Geometric and Spatial (GS) Representation

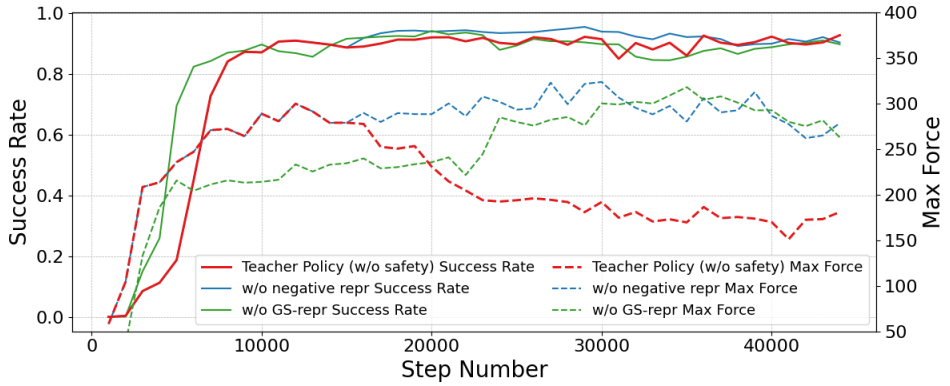

Figure 6: Learning curves of the cluttered-scene policies with or without the Geometric and Spatial Representation we introduced. All experiments shown in this figure were conducted without a safety curriculum.

To better understand the contribution of GS-representation, we introduce two ablation experiments with the same framework: (1) one excludes the proposed geometry-and-spatial representation from both observation and reward function (*teacher policy w/o GS-repr*) (2) and another without the distance to non-target object (*teacher policy w/o negative repr*).

In the *teacher policy w/o GS-repr* setting, we replaced the GS-representation with a simplified distance vector: a distance vector from the object center to the hand palm. The observation still includes the distance to both target and non-target objects but lacks object geometric information. In the *teacher policy w/o negative repr* setting, we use the same geometry-and-spatial representation as in our full method, but mask out features d_{neg} corresponding to surrounding (non-target) objects during cluttered grasping.

Fig.6 shows that while the success rates of these methods are comparable, our method achieves significantly lower maximum contact force as the training progresses, indicating that it learns to avoid collisions with surrounding objects when grasping the target and established safer strategy.

The qualitative difference is more apparent in rollout visualizations (Fig.7). Both ablated variants, *teacher policy w/o GS-repr* and *teacher policy w/o negative repr*, frequently attempt direct top-down grasps, ignoring clutter and applying excessive force on clutter, especially when the target is partially occluded, leading to unsafe behaviors. In contrast, the GS-based policy exhibits more strategic behavior: gently repositioning occluding objects from the side and approaching the target from angles that minimize collision. Such human-like strategies are critical for sim-to-real transfer. Importantly, this also leads to a smooth transition into the safety curriculum stage, with negligible performance drop (Table2).

A.2 Clutter-Density Curriculum Learning

Learning dexterous grasping in cluttered scenes purely through random trial-and-error is extremely challenging. As shown in Figure 8, a policy trained only on clutter scenes with identical settings and full two-stage duration (*w/o curriculum*) fails to make progress, with success rates remaining at zero. To address this, we introduce a two-stage Clutter-Density Curriculum: the policy is first trained on general single-object grasping, then fine-tuned in cluttered scenes to develop strategic, human-like behaviors. This staged approach enables effective learning, as illustrated by the performance curve in Figure 8.

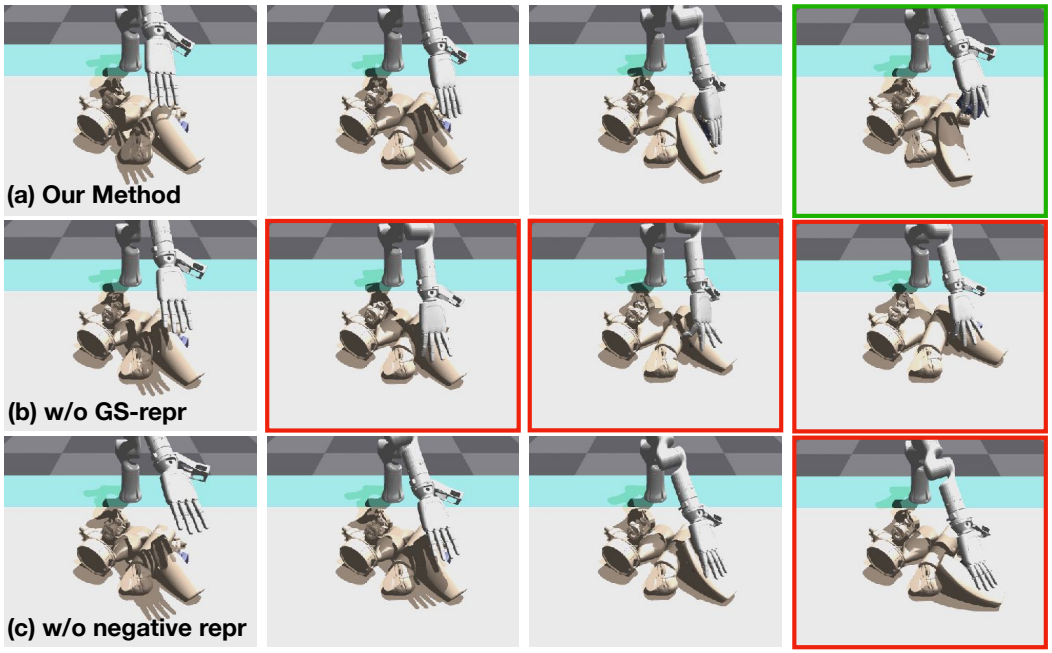

Figure 7: Cluttered Scene Policy Strategy Comparison: (a) Policy trained with GS-Representation, (b) Policy trained without GS-Representation, (c) Policy trained without negative representation. Green bounding boxes indicate successful grasps, while red bounding boxes highlight unsafe or risky actions.

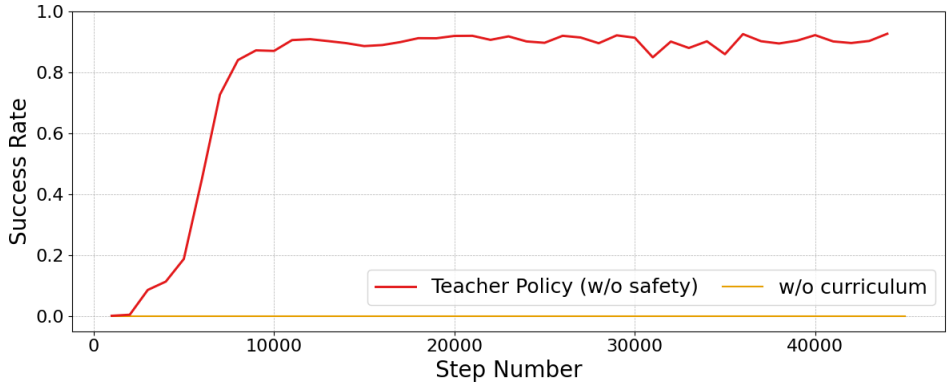

Figure 8: Learning curves of the cluttered-scene policies (1) *Teacher Policy (w/o safety)*: initialized with stage 1 general single-object grasping policy, (2) *w/o curriculum*: trained from scratch directly in cluttered scenes for the full two-stage duration.

B Implentation Details of Teacher Policy Learning

B.1 Geometric and Spatial (GS) Representation

Direct grasp attempts in dense cluttered scenes often result in unsafe collisions with surrounding non-target objects, we therefore introduced a Geometric and Spatial Representation to encourage collision-minimized behavior. This representation serves as a *privileged observation* for the *teacher policy*, providing essential spatial and geometric information for clutter-aware grasping, while bypassing the need for time-consuming point-cloud rendering. At each time step, we have N_{env} environments. For each environment, we sample 200 points from a randomly selected target object, and represent all points as a $N_{\text{env}} \times 200 \times 3$ matrix. We sample 50 points from non-target objects N_{neg},

and represent all points as a $N_{\text{env}} \times 50 \times 3$ matrix. We get 11 base positions from selected finger links, and represent as a $N_{\text{env}} \times 11 \times 3$ matrix. Using torch.norm, we compute distances between link positions and both target object points and non-target object points, identify nearest points, and construct 3D vectors from link bases: $d_{pos}, d_{neg} \in \mathbb{R}^{N_{\text{env}} \times 11 \times 3}$, which serve as observations. During training, $N_{\text{env}} = 16384$ with computation time cost 6ms. This computation is not required by the student policy during real-world deployment.

For single-object grasping scenarios, the representation-related components in both observation and reward are zero-padded for non-target objects.

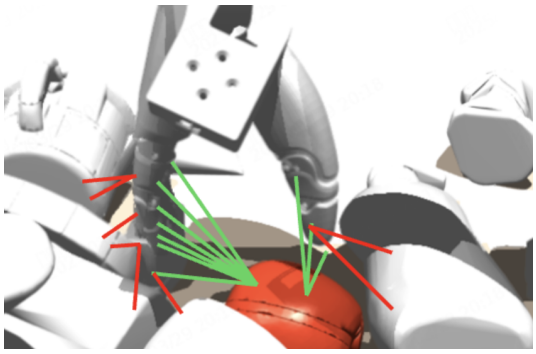

Figure 9: Visualization of the Geometric and Spatial Representation. For each finger joint, distances to the nearest N_{pos} surface points sampled from the target object mesh (d_{pos}, green) and N_{neg} points from surrounding non-target object meshes (d_{neg}, red) are computed and visualized.

B.2 Teacher Observation Space

The observation space for teacher $o_t \in \mathcal{O} = \mathbf{J}^a \times \mathbf{J}^h \times \mathcal{O}^E$ is detailed in Tab. 4.

Table 4: Teacher Observation Space

Index	Description
0 – 19	DoF positions (unscaled) $\mathbf{J}^a \in \mathbb{R}^7$, $\mathbf{J}^h \in \mathbb{R}^{12}$
19 – 22	End-effector position (XYZ)
22 – 25	End-effector orientation (Euler angles: roll, pitch, yaw)
25 – 28	End-effector linear velocity
28 – 31	End-effector angular velocity
31 – 34	Vector from object to middle point
34 – 37	Middle point position
37 – 44	Object pose (position + quaternion)
44 – 47	Object linear velocity
47 – 50	Object angular velocity
50 – 57	Object goal pose (position + quaternion)
57 – 90	Distance features (flattened)
90 – 123	Negative distance features (flattened)
123 – 128	Safety: finger-tip-to-table height (5 values)

B.3 Teacher Reward Function

We first introduce the definitions of r_{pos} and r_{neg} as:

$$r_{pos} = \exp\left(-\alpha_{pos} \cdot \|d_{pos}\|_2\right) \tag{5}$$

$$r_{neg} = \exp\left(-\alpha_{neg} \cdot \bar{d}_{neg}^{\min}\right) \tag{6}$$

Here, r_{pos} provides a positive reward that encourages the dexterous hand to approach the target object. Conversely, r_{neg} introduces a global penalty term based on $\bar{d}_{neg}^{\min}$, the minimum absolute

distance to any non-target (negative) object surface observed throughout the entire grasping episode. This term serves as a global penalty term to discourage risky proximity to non-target objects at any point during execution. At each timestep, the complete reward function is defined as:

$$r = (c_1 \cdot r_{grasp} + c_2 \cdot r_{pos}) \cdot (1 - r_{neg}) \tag{7}$$

and r_{grasp} is defined as:

$$\begin{aligned} r_{grasp} = c_4 \cdot \big(& c_5 \cdot (0.2 - \|p_{current} - p_{goal}\|_2) && \text{goal distance reward} \\ & + c_6 \cdot \exp(-\alpha_{mid} \cdot \|d_{mid}\|_2) \big) && \text{middle point reward} \end{aligned} \tag{8}$$

where $c_1, c_2, c_4, c_5, c_6, \alpha_{pos}, \alpha_{neg}, \alpha_{mid} > 0$ are weighting coefficients. d_{mid} is the 3D vector from middle-point between index finger and thumb to center of object. $p_{current}$ and p_{goal} denote the current position of the target object and the desired position the target object should be lifted to after a successful grasp.

Stage 1 For general single-object grasping, the representation-related components in both observation and reward are zero-padded, with reward function being:

$$r_{stage1} = c_1 \cdot r_{grasp} + c_2 \cdot r_{pos} \tag{9}$$

Stage 2 For strategic cluttered-scene grasping, with reward function is:

$$r_{stage2} = r \tag{10}$$

where r is defined in Eq. 7

Stage 3 For safety finetuning, the reward function for the curriculum is:

$$r_{stage3} = r_{safe} = r - c_3 \cdot r_{\text{force}} \tag{11}$$

where r_{force} is defined as:

$$r_{\text{force}} = \begin{cases} 1 & \text{if } \max_i(f_{z,i}) > f, \\ 0 & \text{otherwise,} \end{cases} \tag{12}$$

with $f_{z,i}$ representing the z-direction contact force at the i-th fingertip and f being the predefined force threshold.

B.4 Safety Curriculum

The safety threshold starts at an initial value of $f_0 = 200$ and gradually decreases to a final value of $\bar{f} = 50$. After reaching a certain success rate threshold, the safety threshold is reduced by 5 units at each step. To prevent overly rapid updates to the threshold, we define a hyperparameter $\Delta T_{\min}$, which specifies the minimum number of iterations required between consecutive threshold updates. During the actual training, we disable collisions between the hand and the table, and introduce two early termination conditions: the first is when the contact force of any fingertip exceeds the threshold f, and the second is when any fingertip penetrates the table surface.

B.5 Teacher Coarse-to-Fine Data-Efficient Learning

The high degrees of freedom (DoFs) in dexterous hands pose significant challenges for efficient learning. To address this, we model the grasping process as a coarse-to-fine trajectory: a coarse approach phase followed by a fine interaction phase. During the approach, hand DoFs are frozen by masking hand actions with their initial state, leveraging privileged object-hand distance information d_{hand}. Once the hand is sufficiently close to the object $d_{hand} < \bar{d}$, the full action space is enabled. $\bar{d} = 0.08$ is used.

Algorithm 1 Safety Curriculum

Initialize empty FIFO queue Q of size K, $\Delta T = 0$, safety threshold $f = f_0$
for $i \leftarrow 1$ to M **do**
  $\tau = \text{rollout\_policy}(\pi_\theta)$ ▷ get rollout trajectory
  $\pi_\theta = \text{optimize\_policy}(\pi_\theta, \tau)$ ▷ update policy
  $\Delta T = \Delta T + 1$
  if $i \bmod L = 0$ **then**
    $w = \text{evaluate\_policy}(\pi_\theta)$ ▷ get success rate w
    append w to the queue Q
    if $\text{avg}(Q) > \bar{w}$ and $\Delta T > \Delta T_{\min}$ **then**
      $f = \min(f + \Delta f, f_{\max})$ ▷ tighten safety constraint
      $\Delta T = 0$
    end if
  end if
end for

Hyperparameters	Value
Num mini-batches	4096
Num opt-epochs	5
Num episode-length	8
Hidden size	[1024, 512, 256]
Clip range	0.2
Max grad norm	1
Learning rate	3e-4
Discount (γ)	0.99
GAE lambda (λ)	0.95
Init noise std	–
Desired kl	0.02
Ent-coef	0.0

Table 5: Hyperparameters of PPO.

	Mem.	Time
Teacher	22G	8 days
Student	22G	1 day

Table 6: Computation resources.

Hyperparameters	Value
Downsample dims	[128, 256, 384]
Encoder output dim	64
Crop shape	[80, 80]
Horizon	4
Num observation steps	2
Num action steps	1
Kernel size	5
Num groups	8
Diffusion steps (training)	100
Diffusion steps (inference)	10
Learning rate	1e-4
Optimizer	AdamW
Weight decay	1e-6
Betas	(0.95, 0.999)
EMA power	0.75
EMA max value	0.9999
LR scheduler	cosine
LR warmup steps	500

Table 7: Key Hyperparameters of Simple DP3 Policy.

B.6 Training Details

All the training and experiment in the paper were run on a single GeForce RTX 4090 GPU with i9-13900K CPU. Memory usage and training time are summarized in Table 6, and PPO hyperparameters for the teacher policy are listed in Table 5.

C Implementation Details of Student Policy Distillation

We offline distilled the teacher policy π^E trained with privilage state information $\mathcal{O}^E$ into a student policy π^S that takes sensory observations $\mathcal{O}^S \in \mathbb{R}^{4109}$ that can be obtained in the real world. The student observation space contains the robot joint position $\mathcal{O}_{robot} \in \mathbb{R}^{13}$ and the partial point cloud $\mathcal{O}_{pc} \in \mathbb{R}^{4096}$ from a fixed side-view camera cropped and transformed to the robot frame.

C.1 Student Observation

The point cloud observation $\mathcal{O}^{pc} = \mathcal{O}^{p} \times \mathcal{O}^{g} \times \mathcal{O}^{s}$ is composed of three components: (1) $\mathcal{O}^{p} \in \mathbb{R}^{4\times 3584}$, the partial point cloud observation from a single third-person camera, with a 1D mask to indicate the grasping target; (2) $\mathcal{O}^{g} \in \mathbb{R}^{4\times 512}$, a synthetic ground point cloud replacing the table surface to mitigate sensor noise; and (3) $\mathcal{O}^{s} \in \mathbb{R}^{4\times 1024}$, the synthetic robot point cloud observation (Sec.4.2.2) with a 1D mask to differentiate between real and synthetic point clouds. The visualization of the point-cloud can be found in Fig. 2.

C.2 Multi-Level Clutter Density Dataset Generation

We collect a dataset $\mathcal{D}^{E}$ consisting of 20,000 successful trajectories generated by the teacher policy π^{E} across 500 clutter scenes with varying levels of clutter density. Demonstrations are evenly distributed across the different clutter levels. The student policy is trained using Imitation Learning (IL) with the DP3 algorithm[35], where a random batch size of 120 is sampled from the dataset. Observations are normalized based on the data distribution's statistics. Detailed training hyperparameters can be found in Tab. 7

C.3 Point Cloud Observation Processing

The point cloud is created from a single-view depth image. The point-cloud pre-processing includes four steps: (i) cropping the point cloud to the workspace region using a manually defined bounding box, (ii) downsampling the point cloud to 3584 points, (iii) transforming the point cloud from the camera frame to the robot base frame, and (iv) replacing the table surface with a synthetic point cloud (512 points) to mitigate point cloud holes caused by the flat table. We apply consistent point cloud preprocessing across simulation and real-world data to ensure alignment. However, we found that adding Gaussian noise and random transformation to point cloud during DP3 training did not improve policy performance in real-world settings.

Figure 10: Point-cloud comparison between simulation(Left) and real world(Right).

C.4 System Identification

To minimize the sim-to-real gap, we calibrate the physical and dynamic parameters of robot via system identification (SI) by aligning simulated trajectories with their real-world counterparts, following the approach in [60, 42].

C.5 Training Details

Memory usage and training time are summarized in Table 6, and the hyperparameters for student policy training are listed in Tab. 7.

D Simulation Environment Generation Details

To generate cluttered tabletop scenes for training and evaluation, we follow a physics-based simulation pipeline. Specifically, we sequentially drop a set of randomly selected objects from a fixed height above the workspace. After all objects are dropped, we allow the simulation to run until the scene reaches a physically stable state. Only those scenes where all objects remain on the tabletop surface are retained for further use. Our simulated robot setup consists of a RealMan RM75-6F 7-DoF robotic arm, equipped with the AgiBot 6-DoF dexterous hand.

E Implementation Details in Real-World

E.1 Real-World Hardware Setup

For real-world deployment, the target object is selected and segmented using SAM2 [64]. The resulting binary mask is projected onto the point cloud using one-hot encoding to isolate the target object. The entire system operates at a frequency of 15 Hz on a single GeForce RTX 4090 GPU with i9-13900K CPU.

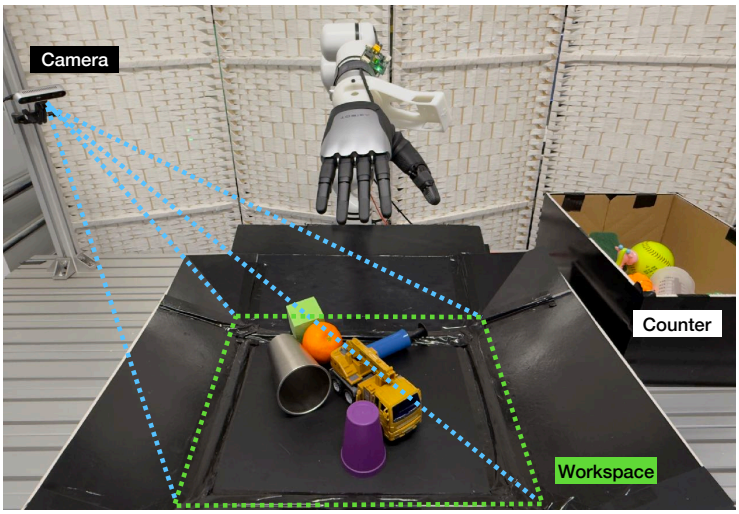

Figure 11: Real-World Setup

To improve deployment stability, we introduce several practical adjustments. Since the system occasionally failed to detect successful grasps, leading to redundant actions, we use finger torque as a success signal: Once it exceeds a threshold, the grasp is considered successful and a predefined lift-up action is executed. To guarantee safety, we add a threshold on end-effector force; if exceeded, the end-effector is gently raised to avoid approaching dangerous force boundaries. Additionally, we slow down the end-effector slightly compared to simulation to ensure smoother and safer real-world execution. Note that, in all real-world experiments, no safety-related human intervention was observed.

F Failure Case Analysis

Our policy struggles with grasping extreme shapes and sizes due to the limitations of hand morphology. Specifically, the policy fails with large objects or excessively flat ones. Additionally, due to the limited working space of our robotic arm, we did not strictly constrain the scene generation to the arm's operational range. As a result, some failures occur when objects fall outside the working area or are pushed out of the range by the arm during grasping. Check our website for failure videos.

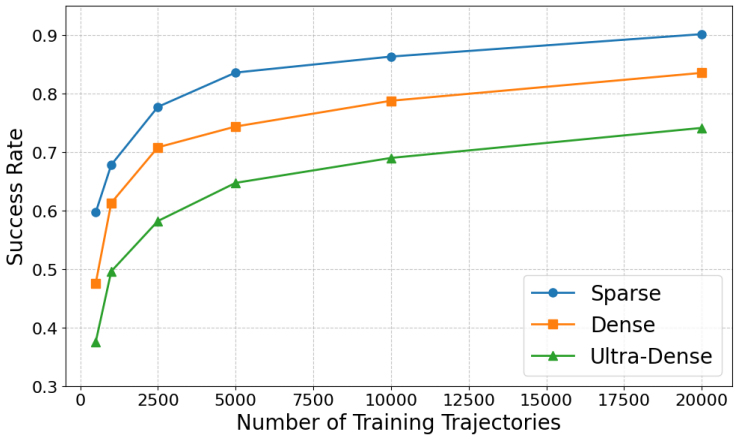

Figure 12: Data Scaling for Generalization

G Data Scaling for Scene-Level Generalization

In this section, we investigate how the performance of student policy scales with the number of collected trajectories. Specifically, we use a consistent teacher policy to collect trajectories in identical scenes, but employ varying scales of trajectory data for student policy distillation. Consistent with the main manuscript, we conduct evaluations across three scenario types: Sparse, Dense, and Ultra-Dense. As illustrated in Figure 12, results from these representative scenes consistently show significant performance improvements as the data scale increases. This enhancement in performance is attributed to the richer diversity and more comprehensive coverage of the state-action space afforded by larger datasets, which facilitates more effective learning and generalization.