Grasp-Oriented Non-Prehensile Manipulation via Learning a Graspability Field

Licheng Zhong and Gim Hee Lee

Department of Computer Science, National University of Singapore, Singapore
{zlicheng, gimhee.lee}@comp.nus.edu.sg
https://zlicheng.com/gomp_page/

Abstract. Non-prehensile manipulation is often used as a preparatory step for robotic grasping, yet existing approaches typically require a predefined target object pose. In practice, however, objects admit multiple graspable configurations and the desired pose is not known in advance. We reformulate non-prehensile manipulation for grasping as optimizing an object centric graspability objective rather than reaching a specific pose. We construct a graspable set from synthesized grasps and define a graspability field that measures how suitable an object configuration is for successful grasp execution. The scalar measure provides a dense learning signal for reinforcement learning and determines when to terminate manipulation. This yields a closed-loop manipulation-to-grasp pipeline driven by a single policy. Experiments in simulation and on a real robot show that the policy reliably reconfigures objects into graspable states and transitions to grasping without external planners or manually specified stopping conditions. The predicted graspability distance correlates with real world grasp success, which indicates that the learned representation captures grasp feasibility of object configurations.

Keywords: Non-Prehensile Manipulation · Grasp-Oriented Manipulation · Graspability Field

1 Introduction

Robotic grasping in unstructured environments often fails not because feasible grasps do not exist, but because the object is initially in an unfavorable configuration. Before a reliable grasp can be executed, the robot frequently needs to interact with the object through pushing, sliding, or reorienting it. These non-prehensile interactions serve as a preparation stage that transforms the object into a configuration suitable for grasping, especially when the object is unstable, partially occluded, or poorly aligned with the gripper.

Recent learning-based approaches have shown promising results by using non-prehensile manipulation as a preparation stage for grasping. Methods such as CORN [7] and DyWA [28] formulate manipulation as a goal-conditioned control

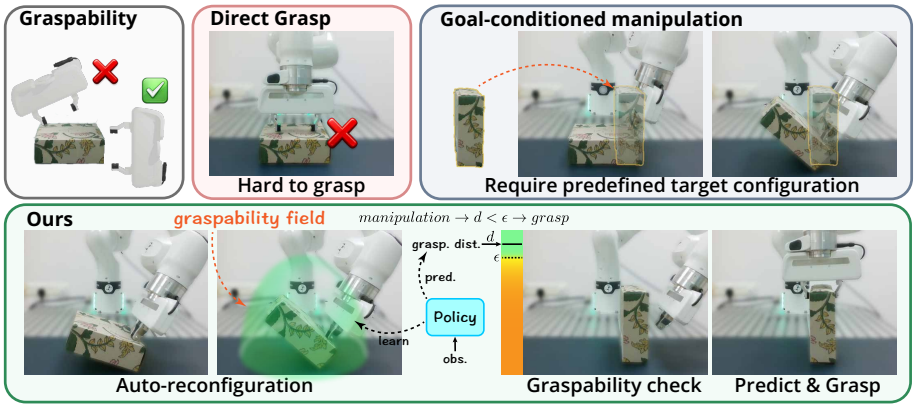


Fig. 1: Direct grasps from a fixed approach can fail due to unfavorable object configurations, even when feasible grasps exist. Instead of requiring a predefined target pose, we learn a graspability field over object states to guide manipulation and trigger the manipulation-to-grasp transition. Once the object enters a graspable region, the robot autonomously transitions from manipulation to grasp execution.

problem, where the policy is guided toward a desired object pose. This formulation enables stable learning and has demonstrated effective object repositioning behaviors in both simulation and real-world settings.

However, the target-pose formulation implicitly assumes that grasp preparation can be reduced to reaching a single predefined configuration. In practical grasping scenarios, many objects admit multiple graspable configurations, and selecting one goal configuration does not necessarily reflect how a robot should interact with the object from its current state. A policy may successfully match the target pose while failing to meaningfully improve grasp success.

More importantly, even when a target pose is specified, reaching it does not necessarily yield the most graspable configuration: a pose-reaching objective optimizes geometric similarity to a predefined configuration, while grasp preparation requires moving toward a nearby region of high graspability. Since graspability typically exhibits multiple local maxima in the object configuration space, a predefined target may lie outside the nearest graspable basin, forcing the policy to move the object away from an easily graspable configuration before eventually approaching the target. As a result, manipulation behavior ends up guided by distance to the target rather than improvement of grasp success.

This reveals a mismatch between pose-reaching objectives and grasp preparation: preparing an object for grasping is not a pose regulation task, but a state improvement process, where the policy should be guided by how graspable the object currently is rather than its proximity to a predefined pose. As shown in Fig. 1, we therefore argue that non-prehensile manipulation for grasping should be driven by graspability rather than explicit target poses: manipulation is a process of progressively improving the graspability of the object, with the

goal not of reaching a particular configuration, but of transforming the object into a state that admits reliable and stable grasps.

Based on this formulation, we propose **GOMP**, a **G**rasp-**O**riented non-prehensile **M**anipulation **P**olicy that maximizes a learned graspability function defined over object configurations. Instead of predicting or tracking a target pose, we learn a graspability field that estimates how suitable the current configuration is for executing a grasp. The policy is trained using reinforcement learning to progressively increase this graspability over time, which naturally enables both manipulation behaviors and stopping decisions for grasp execution.

Experiments in simulation and on a real Franka robot demonstrate that GOMP consistently improves grasp success over target-conditioned baselines. In summary, our contributions are threefold:

- We introduce a new formulation of non-prehensile manipulation as graspability optimization, which replaces pose-reaching objectives with a state-based improvement objective.
- We propose a graspability field representation that provides dense learning signals and enables both manipulation and stopping without requiring explicit target poses.
- We validate the proposed approach through extensive simulation studies on non-prehensile manipulation, and real-world closed-loop experiments that demonstrate consistent improvements in grasp success when using our grasp-oriented manipulation policy as a preparation stage.

2 Related Work

Non-Prehensile Manipulation. Non-prehensile manipulation studies how a robot deliberately changes the configuration of an object without grasping it. Typical actions include pushing, sliding, and reorientation [21, 23, 27, 30, 41, 48]. Both classical planning approaches and contact-based control methods aim to transform an object from an initial state to a desired configuration [5, 32, 33]. Recent learning-based methods acquire such behaviors directly from interaction data. Self-supervised and reinforcement learning approaches learn pushing and rearrangement policies [46], including push–grasp coordination strategies that improve robustness [10, 16, 43]. Despite improved adaptability, the manipulation objective in these methods remains externally specified, typically through a target pose or goal conditioned policy. More recent work investigates coordinated whole arm reorientation [7, 19, 28, 50, 51]. CORN [7] learns policies that manipulate objects toward a predefined target pose, and DyWA [28] further improves perception and distillation. To achieve strong performance, these methods require the desired object configuration to be known in advance. However, the desired configuration is usually unknown in practical grasping scenarios. Objects often admit multiple feasible grasp poses, and perception alone cannot determine which configuration should be reached before interaction. Therefore, for grasping, the objective is not to reach a specific pose but to bring the object into any configuration from which a stable grasp can be executed.

Learning Based Robotic Grasping. Learning-based robotic grasping aims to predict grasp actions directly from sensory observations [8,17]. Early methods learn grasp success predictors or analytic grasp synthesis models from large-scale data, often assuming precise object geometry or full visibility [40, 47]. Vision-based approaches then train networks to estimate stable grasp poses or grasp affordances from images or point clouds [11–14, 37, 39]. Recent generative models produce diverse grasp candidates conditioned on object observations, which improve robustness to shape variation and perception noise [15, 20, 38, 42]. Although effective for grasp execution, these approaches generally assume that the object is already in a graspable configuration when grasp poses are predicted. Some work considers interaction with the environment to make specific grasp poses reachable, such as extrinsic dexterity where objects are pushed against surfaces to enable occluded grasps [49]. These methods still aim to execute a given grasp by modifying reachability. In contrast, our objective is to prepare the object itself by bringing it into any configuration that admits stable grasps without specifying a target pose.

Affordance Guided Manipulation. Several works attempt to connect manipulation and grasping through grasp-related predictions [18, 35]. Vision-based grasping systems often estimate grasp affordances, reachability, or success likelihoods from sensory observations [1, 9, 25, 29, 37, 39, 45]. Manipulation actions are then chosen to expose or enable predicted grasps [24, 26, 44, 46]. These approaches still require a grasp to be selected or evaluated during planning. The objective is to realize a particular grasp candidate or maximize predicted grasp success, rather than to optimize the object configuration itself. In contrast, our method does not predict or evaluate grasps during manipulation. We instead define a graspability measure over object states and learn actions that transform the object toward configurations that admit stable grasps. Thus, manipulation is guided by improving the object state rather than enabling a specific grasp.

3 Problem Formulation

Goal. We study non-prehensile manipulation as a preparatory process for robotic grasping, where the goal is to transform an object into a configuration that admits reliable grasps.

Formulation. Let s_o^t denote the (unobserved) rigid 6D pose (R_o, p_o) of the object at time t, and the policy only receives an observation o_t derived from sensor measurements such as point clouds. We consider an observation-based manipulation policy $\pi(a_t \mid o_t)$ that outputs non-prehensile actions a_t. Instead of specifying a desired target pose, we consider that an object may admit multiple graspable configurations. We therefore define a *graspable set* $\mathcal{S}_g$, which contains object states that allow stable grasps. Manipulation is formulated as moving the object toward this set instead of toward a single predefined pose.

To quantify progress toward graspability, we define a scalar graspability measure $G(s_o)$ that evaluates how close a state is to the graspable set $\mathcal{S}_g$. Given an

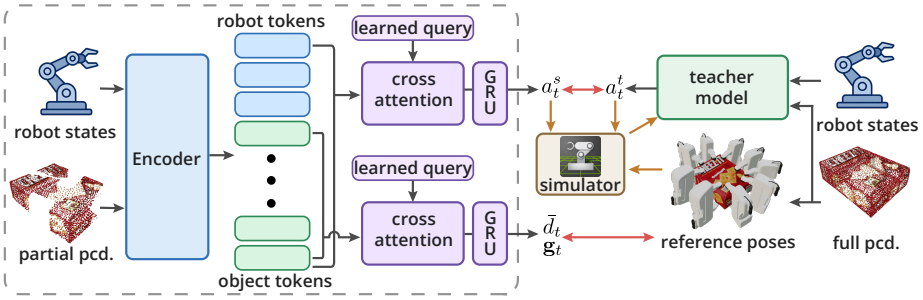


Fig. 2: Method achitecture. A teacher–student framework that encodes observations, aggregates temporal information with a recurrent module, and predicts manipulation actions and a graspability distance. The teacher is used only during training, and the robot executes a grasp once the predicted graspability indicates a feasible configuration.

initial state s_o^0, the manipulation policy aims to produce a trajectory $\{s_o^t\}_{t=0}^T$ that maximizes the graspability of the final state:

$$\pi^* = \arg\max_{\pi} \mathbb{E}_{\pi} \left[G(s_o^T) \right], \tag{1}$$

where the expectation is taken over the state distribution induced by the policy and the environment dynamics. This formulation eliminates the need for an explicit target pose: any state within the graspable set is considered successful. In the following section, we show how the graspable set can be constructed and how $G(s_o)$ is derived from distances to this set.

4 Our Method

Overview. Fig. 2 shows the overview of our framework. Reference graspable configurations from full object geometry (Sec. 4.1) are used only during training to build graspability field for supervision (Sec. 4.2) and reward (Sec. 4.3), and are not available at test time. From partial observations, the policy predicts manipulation actions, a graspability distance, and a grasp pose (Sec. 4.4). The robot manipulates in closed loop and executes a grasp once the predicted distance indicates a sufficiently graspable configuration (Sec. 4.5), removing the need for a predefined target pose.

Architecture. As shown in Fig. 2, we employ a perception–control architecture that estimates graspability and actions jointly. Observations are encoded using the object-centric encoder from CORN [7]. Since our tasks and objects differ from the original setting, we fine-tune only the final attention layer while freezing earlier layers to preserve geometric features while adapting to new interaction dynamics. The encoder produces object and robot tokens, which are queried through cross-attention conditioned on the robot state to obtain a compact representation of the current object configuration.

Due to partial observability of the object configuration from a single frame, the student policy incorporates a recurrent module (GRU [6]) to aggregate temporal information. The recurrent state allows the policy to infer object motion from recent observations and actions, which improves robustness to occlusion and contact-induced dynamics. The resulting feature is passed to the prediction heads that output the manipulation action, the predicted graspability distance, and the terminal grasp pose. A teacher policy with a similar perception backbone but without recurrent memory is used only to generate stable manipulation behaviors for distillation.

4.1 Graspable Reference Pose Generation

The graspable set $\mathcal{S}_g$ is not directly available in practice and cannot be specified analytically for arbitrary objects. To approximate this set, we employ an off-the-shelf grasp synthesis model [38] that predicts feasible grasps from object geometry.

Given an observed object point cloud, the grasp model produces multiple candidate grasps. Each grasp corresponds to a relative pose between the gripper and the object. We convert each grasp into a reference object pose by fixing a vertical top-down gripper pose and inverting the grasp transform $T_{\text{obj}}^{\text{ref}} = R_x(\pi)\left(T_{\text{grasp}}^{\text{obj}}\right)^{-1}$, where $R_x(\pi)$ is a 180° rotation about the x-axis for vertical top-down alignment. The object is then vertically translated to rest on the table. This results in a set of graspable object states:

$$\mathcal{S}_g = \{\hat{s}_o^i\}_{i=1}^{G}, \tag{2}$$

where each $\hat{s}_o^i$ represents a configuration in which a stable grasp can be executed.

Importantly, these configurations are not used as target poses for the manipulation policy. They are never provided to the policy as observations, nor used for imitation or supervision. Instead, they serve solely to define a set of desirable object states that characterize graspability.

In the next section, we derive a scalar graspability measure by computing distances between the current object state and this graspable set.

4.2 Graspability Field Definition

A graspable configuration is not unique: many distinct object poses may allow stable grasps. Therefore, measuring progress toward grasping using a single target pose is inappropriate since it artificially constrains the manipulation objective. Instead, we measure how close the current object configuration is to a set of graspable states.

Given the graspable set $\mathcal{S}_g = \{\hat{s}_o^i\}_{i=1}^{G}$ defined in Sec. 3, we define a distance from the current state s_o to this set. A natural measure of graspability is the minimum distance to the set:

$$d_{\text{anchor}}(s_o) = \min_{i \in \{1,\dots,G\}} \mathcal{D}(s_o, \hat{s}_o^i), \tag{3}$$

where $\mathcal{D}(\cdot,\cdot)$ measures the geometric discrepancy between two object configurations. In practice, $\mathcal{D}$ compares the spatial arrangement of the object geometry (*e.g.*, keypoints or surface points) under two configurations, and does not depend on the robot end-effector pose. A smaller value indicates that the object is closer to a configuration that admits a stable grasp.

In our implementation, we approximate $\mathcal{D}$ using geometry alignment. Let $\{p_k\}_{k=1}^{K}$ denote a fixed set of keypoints sampled from the object bounding box. For a configuration s_o with rigid transform $T(s_o)$, the distance between two configurations is computed as the mean Euclidean discrepancy of transformed keypoints:

$$\mathcal{D}(s_o, \hat{s}_o^i) = \frac{1}{K}\sum_{k=1}^{K} \|T(s_o)p_k - T(\hat{s}_o^i)p_k\|_2. \tag{4}$$

This formulation compares object geometry rather than pose parameters and does not require an explicit pose representation at inference time. In particular, the policy only observes point clouds, while the geometric distance is used internally to construct the learning signal.

However, directly optimizing the minimum distance leads to discontinuities since the closest reference configuration may switch abruptly during manipulation. This instability makes learning difficult. To obtain a smooth and differentiable objective, we replace the hard minimum with a soft aggregation:

$$d_{\text{soft}}(s_o) = -\tau \log \sum_{i=1}^{G} \exp\left(-\frac{\mathcal{D}(s_o, \hat{s}_o^i)}{\tau}\right), \tag{5}$$

which smoothly approximates the minimum distance. As $\tau \to 0$, d_{soft} approaches d_{anchor}, while larger τ allows the measure to account for multiple nearby graspable configurations simultaneously.

Although the soft distance provides a smooth objective, it may still encourage frequent switching between different graspable modes across time steps. To promote temporal consistency, we introduce an anchor term based on the previously selected configuration index i^*:

$$\tilde{d}(s_o^t) = d_{\text{soft}}(s_o^t) + \lambda \mathcal{D}(s_o^t, \hat{s}_o^{i^*}). \tag{6}$$

The anchor steers the object toward the selected graspable configuration as it approaches the target. This mechanism stabilizes the trajectory and prevents oscillations between competing grasp modes.

We interpret $\tilde{d}(s_o)$ as a graspability field defined over the space of object configurations. States with smaller values correspond to higher graspability. Importantly, this field evaluates a property of the object state rather than alignment with any specific grasp pose. Specifically, the policy does not aim for a particular end-effector pose. Instead, it seeks to move the object into a configuration that admits many feasible grasps. Unlike target-based objectives that optimize distance to a single pose, this field defines a continuous landscape guiding the object toward regions that allow reliable grasps. Consequently, multiple manipulation strategies that produce graspable object states are equally valid.

4.3 Learning with Graspability-Driven Rewards

The graspability field defined in Sec. 4.2 provides a continuous measure of how suitable an object configuration is for grasping. To learn a manipulation policy, we convert this measure into a training signal. A straightforward choice is to define the instantaneous reward as a monotonic function of the graspability distance:

$$r_t = f\left(\tilde{d}(s_o^t)\right), \tag{7}$$

where smaller distance corresponds to higher reward. This directly aligns policy learning with the objective of reducing the distance to the graspable set.

However, optimizing a state-based reward can lead to slow learning due to delayed credit assignment. To encourage incremental progress toward graspable configurations, we employ potential-based reward shaping. Following previous work [7], we use a potential function as a monotonic decreasing function of the graspability distance:

$$\Phi(s_o) \triangleq \phi(\tilde{d}(s_o)), \qquad \phi(d) = k_1\, \beta^{k_2 d}, \tag{8}$$

where k_1 and k_2 control the scale and sharpness of the potential and $\beta \in (0, 1)$. The shaped reward is then defined as:

$$r_t = \gamma \Phi(s_o^{t+1}) - \Phi(s_o^t), \tag{9}$$

which rewards reductions in the graspability distance between consecutive states.

Intuitively, the agent receives a positive reward whenever the object moves closer to the graspable set (*i.e.*, when $\tilde{d}$ decreases). This provides a dense supervision signal and eliminates the need to execute grasp attempts during training. This formulation encourages the policy to progressively move the object toward graspable configurations while preserving the optimal policy under the original objective. Importantly, the reward is not a heuristic signal but a direct consequence of the graspability field: learning corresponds to minimizing the distance to the graspable set over time.

4.4 Grasp-Oriented Manipulation Policy (GOMP)

We instantiate the proposed formulation using a grasp-oriented manipulation policy, termed GOMP. The policy maps observations to non-prehensile manipulation actions and is trained to optimize the graspability field defined in Sec. 4.2. Instead of assuming access to the full object state, the policy operates directly on sensory observations. At each time step t, the policy receives an observation o_t derived from perception, including point cloud measurements and robot proprioceptive information, and produces an action

$$a_t \sim \pi_\theta(\cdot \mid o_t), \tag{10}$$

where a_t corresponds to a non-prehensile manipulation command. In addition to actions, the policy predicts a scalar graspability distance $\hat{d}_t$ and a grasp pose

used only for the final grasp execution. Neither reference configurations nor any target pose is provided as input.

We emphasize that the policy architecture itself is not specific to our method. Any policy capable of mapping observations to manipulation actions can be used within our framework since the key contribution lies in the graspability-based objective instead of the network design. During training, the predicted distance $\hat{d}_t$ is supervised to regress the graspability distance $d(s_o^t)$ defined by the graspability field. This enables the policy to estimate graspability directly from observations and to use the same signal for both manipulation and the manipulation-to-grasp transition at execution time.

4.5 Graspability-Guided Manipulation-to-Grasp Transition

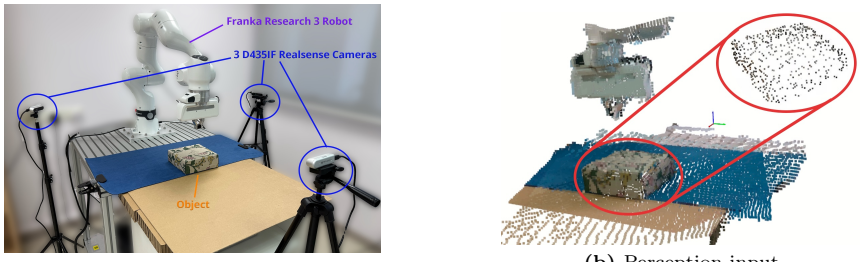
Unlike target-based manipulation, our formulation does not specify a desired object pose. Consequently, the system must determine when the object is sufficiently prepared for grasping and when manipulation should terminate. We leverage the graspability field to provide this decision signal. Although the graspability field is defined over object states during training, the system relies on the policy-predicted graspability distance $\hat{d}_t$ at execution time. Manipulation is considered complete once the predicted graspability indicates that the object has reached a sufficiently graspable configuration.

Since the magnitude of the graspability distance depends on object scale, a fixed threshold would vary across objects of different sizes. To obtain a scale-invariant stopping criterion, we normalize the predicted distance by the object size. Let L denote the diagonal length of the object bounding box. We define the normalized predicted graspability distance as $\bar{d}_t = \hat{d}_t / L$. We transition from manipulation to grasp execution when the graspability distance remains below a threshold for a short temporal window:

$$\bar{d}_{t:t+K} < \epsilon, \tag{11}$$

where $\bar{d}_{t:t+K}$ denotes the maximum of the predicted normalized graspability distances over a temporal window of K steps.

Importantly, the transition criterion does not rely on a predefined target pose, and thus the policy can decide when to stop manipulating. Any configuration in the graspable set counts as success. Since GOMP optimizes over the entire graspable set rather than a single target, the policy can move toward alternative graspable regions that are more stable and reachable under contact dynamics; geometrically graspable but dynamically unstable states do not satisfy the stability constraint and are therefore not counted as successful. At the transition time, the policy outputs a grasp pose $\mathbf{g}_t$ to parameterize the final grasp action. The pose does not supervise manipulation and does not define the learning objective. It only specifies how the gripper should execute the grasp once the object reaches a graspable configuration. Using the same graspability measure for both learning (Sec. 4.3) and stopping teaches the policy how to manipulate the object and when further manipulation is unnecessary. This produces a closed-loop pipeline in which manipulation naturally prepares the object for grasp execution.



(a) Real world setup **(b)** Perception input

Fig. 3: Real-world experimental setup **(a)** and perception input **(b)**.

Fig. 4: Objects used in real world experiments.

5 Experiment

5.1 Experimental Setup & Implementation Details

Simulation Setup. All training is done in IsaacSim [34] and IsaacLab [31], using a Franka Emika Panda with a parallel jaw gripper. For both training and evaluation, we manually select task-suitable objects from the DexGraspNet [40], GraspFactory [36], and the YCB dataset [2–4]. The teacher policy receives full object point cloud, while the student policy operates on partial observations. Following CORN [7], we render partial point clouds using nvdiffrast [22].

Real World Setup. As shown in Fig. 3, we run real-world experiments on a Franka Research 3 with a parallel jaw gripper. Three Intel RealSense D435IF RGB-D cameras capture multi-view depth images that are fused into a single partial point cloud. The cloud is filtered by a workspace bounding box, and the robot points are removed using forward kinematics [7]. As illustrated in Fig. 4, the evaluation set includes objects with diverse geometries, friction properties, and compliance levels, and no ground-truth poses, markers, or external tracking systems are used. All experiments use the standard parallel jaw gripper without friction enhancing materials. Successful grasps therefore depend on the policy that brings objects into graspable configurations, not on extra gripper friction.

Implementation Details. We train our policy with a teacher-student framework: reinforcement learning followed by online policy distillation. The teacher policy is trained for 15k iterations (approximately 10 hours on a single NVIDIA RTX A6000 GPU). A velocity constraint on the object is introduced after 5k iterations. Before policy distillation, we further train the teacher for an additional 10k iterations with an action-scale curriculum to reduce the effective action range. Domain randomization is applied throughout training, including variations in object mass, scale, friction, and initial poses.

Table 1: Simulation evaluation of non-prehensile manipulation as a grasp preparation task. Target Pose indicates whether the policy is explicitly conditioned on a predefined object configuration. Stable Check indicates whether dynamic stability constraints (velocity thresholds) are required in the success criterion.

Method	Target Pose	Stable Check	Seen Objects		Unseen Objects	
			Success Rate ↑	Avg. Time ↓	Success Rate ↑	Avg. Time ↓
Retrained Baseline [7, 28]	✓	✗	36.1%	5.3s	28.8%	6.1s
Nearest-reference Oracle	✓	✗	52.0%	8.1s	50.2%	8.8s
Multi-reference Oracle	✓	✗	42.2%	11.3s	39.4%	9.9s
Pretrained Baseline [7, 28]	✓	✗	–	–	6.1%	12.9s
Pretrained Baseline [7, 28]	✓	✓	–	–	1.9%	12.1s
GOMP (ours) Teacher	✗	✓	84.0%	4.6s	77.9%	4.8s
GOMP (ours) Student	✗	✓	75.5%	6.1s	68.1%	6.2s

5.2 Non-Prehensile Manipulation Evaluation

We first evaluate non-prehensile manipulation in simulation, where precise object states are available. This experiment does not assess grasp success; it tests whether the policy can reliably reconfigure objects toward graspable configurations without predefined target poses.

Baselines. We compare with the goal conditioned pose reaching teacher model used in both CORN [7] and DyWA [28], whose original evaluation targets physically stable resting poses verified in a different simulator (IsaacGym). Direct comparison is non-trivial because these methods target pose reaching, not grasp preparation: our reference poses come from grasp synthesis and are graspable but not always stable resting poses. Thus, we evaluate: 1) **Pretrained baseline**: the released CORN/DyWA teacher checkpoints on our objects, with targets sampled from our (possibly dynamically unstable) synthesized references, making pose reaching difficult even when feasible grasps exist. 2) **Retrained baseline**: the same teacher policy reimplemented and retrained in our codebase with identical observations and action space, isolating the effect of the objective from implementation details; the unstable-target issue remains. 3) **Multi-reference oracle**: the retrained baseline evaluated against all candidate references per episode, reporting the best result, which upper-bounds the gain from simply having access to multiple targets. 4) **Nearest-reference oracle**: the retrained baseline using, at every step, the reference closest to the current object state as its target – the fairest upper bound, since it always selects the most reachable target. 5) **Our method**: our target-free policy that optimizes the graspability objective defined in Sec. 4. We use 10 reference configurations in all evaluations.

Evaluation metric. We evaluate manipulation as grasp preparation, so success requires both geometric alignment and dynamic stability: position error < 0.05m, orientation error $< 15°$, linear velocity < 0.01m/s, and angular velocity < 0.1rad/s. The goal-conditioned baseline targets its assigned reference pose, while GOMP – not conditioned on a target pose – succeeds if the final

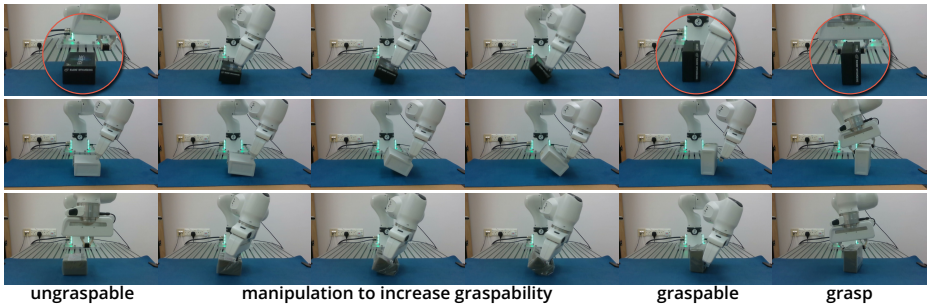


Fig. 5: Qualitative results of the full manipulation-to-grasp pipeline on real world objects. Each row shows a temporal sequence where the robot manipulates the object to increase graspability and autonomously transitions to grasp execution once a graspable configuration is reached.

configuration reaches *any* graspable reference within the same tolerances; both therefore share the same geometric criterion, with GOMP simply having access to the full graspable set rather than a single target. For completeness, we also report the geometric criterion used by prior work (without velocity thresholds).

Results. Tab. 1 shows that goal conditioned pose reaching degrades when the targets are unstable (pretrained baseline); this does not indicate a failure of the original method, but reveals a mismatch between pose-reaching objectives and grasp preparation. Retraining the same policy in our setup improves performance but still remains substantially worse, often oscillating due to residual pose error. In contrast, our policy achieves high success on seen/unseen objects without any target pose and terminates once the configuration becomes sufficiently graspable. This experiment only evaluates the manipulation stage: goal conditioned methods require an external grasp planner once the target pose is reached, while our policy also predicts when to transition to grasp execution. We evaluate the closed-loop pipeline on a real robot in Sec. 5.3.

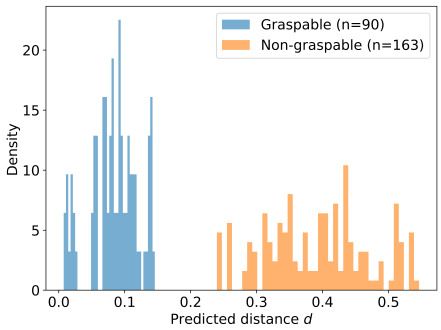
5.3 Manipulation-to-Grasp Evaluation

We evaluate the full manipulation-to-grasp pipeline on a real robot (Sec. 5.1). For each object, we manually set challenging initial configurations where the object is in a non-graspable state, so that a direct grasp from a fixed approach pose without prior manipulation almost always fails (8% success). The same policy performs non-prehensile manipulation until the predicted graspability distance falls below the transition threshold, and then executes a grasp using its predicted grasp pose (no auxiliary planners or external detectors). Quantitative and qualitative results are shown in Tab. 2 and Fig. 5.

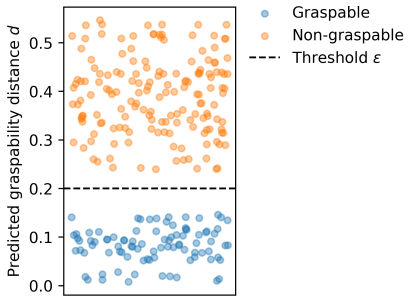
The subsequent grasps are almost always successful once non-prehensile manipulation succeeded. Failures are mainly due to unsuccessful reconfiguration, which indicates that the graspability field provides a reliable signal for deciding when manipulation has sufficiently prepared the object for grasping. The

Table 2: Real world manipulation-to-grasp performance on diverse objects. Each object is tested five times from challenging initial configurations.

	PaperBox	SmallBookend	BigBookend	RealsenseBox	DustBin	Sponge	Bottel	Can	TissuePack	Snack	Avg.
Manip.	4/5	4/5	4/5	5/5	3/5	4/5	4/5	3/5	1/5	0/5	64%(32/50)
Grasp	4/4	4/4	3/4	5/5	3/3	4/4	4/4	3/3	1/1	0/0	97%(31/32)
Total	4/5	4/5	3/5	5/5	3/5	4/5	4/5	3/5	1/5	0/5	62%(31/50)



(a) Distribution of predicted graspability distance.



(b) Graspability-based separation and transition threshold.

Fig. 6: Validation of graspability-based transition in real-world experiments. Graspable configurations correspond to lower predicted distances; a single threshold separates graspable from non-graspable states.

only consistently unsuccessful object is *Snack*, which is highly deformable and slippery. For rigid objects such as *PaperBox* and *DustBin*, failures arise from two modes: (1) the object slides during manipulation, causing insufficient or inaccurate reconfiguration within the time limit; and (2) the robot reaches the predefined safety force limit due to strong contact with the table or object, terminating manipulation before a graspable configuration is reached. The predicted graspability distance still correlates with grasp success when manually placed in a favorable configuration.

Validation of Graspability Based Transition. We validate the predicted graspability distance in Eq. (11) across diverse real world object configurations. Labeling each configuration as graspable or non-graspable reveals a clear separation in predicted distances (Fig. 6), supporting a single transition threshold ϵ. Instead of reflecting the executed motion, the result indicates that the predicted distance captures a property of the object configuration. This justifies its use as both a learning objective and the manipulation-to-grasp transition signal.

5.4 Ablation Studies

We ablate the number of reference poses and the graspability distance design (hard-min vs. soft aggregation, and removing the temporal anchoring term in Eq. (6)). Tab. 3 shows that increasing the number of references generally

Table 3: Ablation studies in simulation.

	Seen Objects			Unseen Objects		
	Success Rate ↑	Avg. Time ↓	Final $\bar{d}_T$ ↓	success rate ↑	Avg. Time ↓	Final $\bar{d}_T$ ↓
1 refer. poses	82.2%	4.6s	0.095	72.8%	4.8s	0.115
3 refer. poses	80.8%	4.7s	0.102	75.2%	5.1s	0.107
5 refer. poses	76.2%	5.6s	0.112	71.4%	5.8s	0.122
8 refer. poses	81.2%	4.8s	0.102	75.6%	5.2s	0.111
10 refer. poses	**84.0%**	**4.6s**	0.098	**77.9%**	**4.8s**	0.118
15 refer. poses	79.1%	4.8s	0.106	72.1%	5.3s	0.121
20 refer. poses	76.0%	6.4s	0.115	68.1%	6.0s	0.132
use hardmin Eq. (3)	81.1%	5.0s	0.102	74.3%	5.2s	0.118
w/o anchor Eq. (6)	78.1%	5.0s	0.108	74.1%	5.1s	0.116

improves performance on unseen objects, and thus indicates better coverage of graspable configurations. Using 10 references provides the best trade-off between success rate and execution time, while too many references slightly slow execution and may reduce success due to conflicting descent directions.

Interestingly, even a single reference pose already yields reasonable performance. This does not reduce the objective to target-pose reaching: supervision is defined by distance to a grasp-feasible configuration rather than reproducing a specific pose. Due to contact and support constraints, configurations far from the reference pose can still be graspable while some configurations with small SE(3) error are not. The policy therefore learns to move the object into a graspable basin instead of matching a particular configuration; multiple references mainly improve coverage of this basin, while a single reference still provides a locally valid descent direction. In contrast, a pose-tracking reward continues to penalize pose error even after the object becomes graspable. This often induces corrective motions or oscillations.

6 Conclusion

We proposed a grasp-oriented formulation of non-prehensile manipulation for robotic grasping. Instead of reaching a predefined target pose, the robot is trained to optimize an object-centric graspability objective. We construct a graspable set from synthesized grasps and define a graspability field that measures how suitable an object configuration is for grasp execution. This scalar measure provides a unified signal for learning manipulation behaviors and decides when manipulation should terminate, which enables a closed-loop manipulation-to-grasp pipeline with a single policy. Experiments in simulation and the real world show that the policy reliably reconfigures objects into graspable states and transitions to grasping without external planners or manually specified stopping conditions. The same graspability measure is used for learning, transition control, and evaluation, and consistently predicts real-world grasp success. This suggests the learned distance represents grasp feasibility of object configurations instead of a task-specific reward heuristic.

Acknowledgements

This research / project is supported by the National Research Foundation (NRF) Singapore, under its NRF-Investigatorship Programme (Award ID. NRF-NRFI09-0008), and the Tier 2 grant MOET2EP20124-0015 from the Singapore Ministry of Education.

References

1. Breyer, M., Chung, J.J., Ott, L., Siegwart, R., Nieto, J.: Volumetric grasping network: Real-time 6 dof grasp detection in clutter. In: CoRL. pp. 1602–1611. PMLR (2021)
2. Calli, B., Singh, A., Bruce, J., Walsman, A., Konolige, K., Srinivasa, S., Abbeel, P., Dollar, A.M.: Yale-cmu-berkeley dataset for robotic manipulation research. The International Journal of Robotics Research **36**(3), 261–268 (2017)
3. Calli, B., Singh, A., Walsman, A., Srinivasa, S., Abbeel, P., Dollar, A.M.: The ycb object and model set: Towards common benchmarks for manipulation research. In: ICRA. pp. 510–517. IEEE (2015)
4. Calli, B., Walsman, A., Singh, A., Srinivasa, S., Abbeel, P., Dollar, A.M.: Benchmarking in manipulation research: Using the yale-cmu-berkeley object and model set. IEEE Robotics & Automation Magazine **22**(3), 36–52 (2015)
5. Cheng, X., Huang, E., Hou, Y., Mason, M.T.: Contact mode guided motion planning for quasidynamic dexterous manipulation in 3d. In: ICRA. pp. 2730–2736. IEEE (2022)
6. Cho, K., Van Merriënboer, B., Gulçehre, Ç., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder–decoder for statistical machine translation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp. 1724–1734 (2014)
7. Cho, Y., Han, J., Cho, Y., Kim, B.: Corn: Contact-based object representation for nonprehensile manipulation of general unseen objects. In: ICLR (2024)
8. Deng, S., Yan, M., Wei, S., Ma, H., Yang, Y., Chen, J., Zhang, Z., Yang, T., Zhang, X., Cui, H., et al.: Graspvla: a grasping foundation model pre-trained on billion-scale synthetic action data. In: CoRL. PMLR (2025)
9. Ding, K., Chen, B., Wu, R., Li, Y., Zhang, Z., Gao, H.a., Li, S., Zhou, G., Zhu, Y., Dong, H., et al.: Preafford: Universal affordance-based pre-grasping for diverse objects and environments. In: IROS. pp. 7278–7285. IEEE (2024)
10. Dogar, M.R., Srinivasa, S.S.: Push-grasping with dexterous hands: Mechanics and a method. In: IROS. pp. 2123–2130. IEEE (2010)
11. Fang, H.S., Wang, C., Fang, H., Gou, M., Liu, J., Yan, H., Liu, W., Xie, Y., Lu, C.: Anygrasp: Robust and efficient grasp perception in spatial and temporal domains. IEEE TRO (2023)
12. Fang, H.S., Wang, C., Gou, M., Lu, C.: Graspnet: a large-scale clustered and densely annotated dataset for object grasping. arXiv preprint arXiv:1912.13470 (2019)
13. Fang, H.S., Wang, C., Gou, M., Lu, C.: Graspnet-1billion: A large-scale benchmark for general object grasping. In: CVPR. pp. 11444–11453 (2020)
14. Guo, D., Sun, F., Kong, T., Liu, H.: Deep vision networks for real-time robotic grasp detection. International Journal of Advanced Robotic Systems **14**(1), 1729881416682706 (2016)

15. Hager, J., Bauer, R., Toussaint, M., Mainprice, J.: Graspme-grasp manifold estimator. In: 2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN). pp. 626–632. IEEE (2021)
16. Hu, B., Tian, H., Wang, D., Huang, H., Zhu, X., Walters, R., Platt, R.: Push-grasp policy learning using equivariant models and grasp score optimization. IEEE RAL **10**(11), 11180–11187 (2025)
17. Huang, Y., Davies, T., Yan, J., Chen, X., Tian, Y., Hu, L.: Robograsp: A universal grasping policy for robust robotic control. arXiv preprint arXiv:2502.03072 (2025)
18. Hundt, A., Killeen, B., Greene, N., Wu, H., Kwon, H., Paxton, C., Hager, G.D.: "good robot!": Efficient reinforcement learning for multi-step visual tasks with sim to real transfer. IEEE RAL **5**(4), 6724–6731 (2020)
19. Jiang, B., Wu, Y., Zhou, W., Paxton, C., Held, D.: Hacman++: Spatially-grounded motion primitives for manipulation. In: RSS (2024)
20. Jiang, Z., Zhu, Y., Svetlik, M., Fang, K., Zhu, Y.: Synergies between affordance and geometry: 6-dof grasp detection via implicit representations. arXiv preprint arXiv:2104.01542 (2021)
21. Jung, H., Lee, D., Park, H., Kim, J., Kim, B.: Spin: distilling skill-rrt for long-horizon prehensile and non-prehensile manipulation. arXiv preprint arXiv:2502.18015 (2025)
22. Laine, S., Hellsten, J., Karras, T., Seol, Y., Lehtinen, J., Aila, T.: Modular primitives for high-performance differentiable rendering. ACM TOG **39**(6), 1–14 (2020)
23. Lee, H.Y., Zhou, P., Duan, A., Ma, W., Yang, C., Navarro-Alarcon, D.: Non-prehensile tool-object manipulation by integrating llm-based planning and manoeuvrability-driven controls. arXiv preprint arXiv:2412.06931 (2024)
24. Li, D., Zhao, C., Yang, S., Ma, L., Li, Y., Zhang, W.: Learning instruction-guided manipulation affordance via large models for embodied robotic tasks. In: 2024 International Conference on Advanced Robotics and Mechatronics (ICARM). pp. 662–667. IEEE (2024)
25. Li, G., Tsagkas, N., Song, J., Mon-Williams, R., Vijayakumar, S., Shao, K., Sevilla-Lara, L.: Learning precise affordances from egocentric videos for robotic manipulation. In: ICCV (2025)
26. Lin, T.J., Yeh, J.F., Su, H.T., Lin, C.Y., Chen, Y.T., Hsu, W.H.: Affordance-guided coarse-to-fine exploration for base placement in open-vocabulary mobile manipulation. arXiv preprint arXiv:2511.06240 (2025)
27. Lynch, K.M., Mason, M.T.: Stable pushing: Mechanics, controllability, and planning. The international journal of robotics research **15**(6), 533–556 (1996)
28. Lyu, J., Li, Z., Shi, X., Xu, C., Wang, Y., Wang, H.: Dywa: Dynamics-adaptive world action model for generalizable non-prehensile manipulation. In: ICCV (2025)
29. Mahler, J., Liang, J., Niyaz, S., Laskey, M., Doan, R., Liu, X., Ojea, J.A., Goldberg, K.: Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. arXiv preprint arXiv:1703.09312 (2017)
30. Mason, M.T.: Progress in nonprehensile manipulation. The International Journal of Robotics Research **18**(11), 1129–1141 (1999)
31. Mittal, M., Roth, P., Tigue, J., Richard, A., Zhang, O., et al.: Isaac Lab - A GPU-Accelerated Simulation Framework for Multi-Modal Robot Learning. arXiv preprint arXiv:2511.04831 (2025)
32. Mordatch, I., Popović, Z., Todorov, E.: Contact-invariant optimization for hand manipulation. In: Proceedings of the ACM SIGGRAPH/Eurographics symposium on computer animation. pp. 137–144 (2012)
33. Mordatch, I., Todorov, E., Popović, Z.: Discovery of complex behaviors through contact-invariant optimization. ACM TOG **31**(4), 1–8 (2012)

34. NVIDIA: Isaac Sim, `https://github.com/isaac-sim/IsaacSim`, (accessed June 29, 2026)
35. Pohl, C., Hitzler, K., Grimm, R., Zea, A., Hanebeck, U.D., Asfour, T.: Affordance-based grasping and manipulation in real world applications. In: IROS (2020)
36. Srinivas, S.K., Shukla, Y., Arnold, A., Chitta, S.: Graspfactory: A large object-centric grasping dataset. arXiv preprint arXiv:2509.20550 (2025)
37. Sundermeyer, M., Mousavian, A., Triebel, R., Fox, D.: Contact-graspnet: Efficient 6-dof grasp generation in cluttered scenes. In: ICRA. pp. 13438–13444. IEEE (2021)
38. Urain, J., Funk, N., Peters, J., Chalvatzaki, G.: Se(3)-diffusionfields: Learning smooth cost functions for joint grasp and motion optimization through diffusion. In: ICRA. IEEE (2023)
39. Wang, C., Fang, H.S., Gou, M., Fang, H., Gao, J., Lu, C.: Graspness discovery in clutters for fast and accurate grasp detection. In: ICCV. pp. 15964–15973 (October 2021)
40. Wang, R., Zhang, J., Chen, J., Xu, Y., Li, P., Liu, T., Wang, H.: Dexgraspnet: A large-scale robotic dexterous grasp dataset for general objects based on simulation. In: ICRA. pp. 11359–11366. IEEE (2023)
41. Wang, Y., Li, Y., Yang, Y., Chen, Y.: Dexterous non-prehensile manipulation for ungraspable object via extrinsic dexterity. arXiv preprint arXiv:2503.23120 (2025)
42. Weng, T., Held, D., Meier, F., Mukadam, M.: Neural grasp distance fields for robot manipulation. In: ICRA. pp. 1814–1821. IEEE (2023)
43. Xu, K., Yu, H., Lai, Q., Wang, Y., Xiong, R.: Efficient learning of goal-oriented push-grasping synergy in clutter. IEEE RAL **6**(4), 6337–6344 (2021)
44. Xu, R., Shen, Y., Li, X., Wu, R., Dong, H.: Naturalvlm: Leveraging fine-grained natural language for affordance-guided visual manipulation. IEEE RAL **9**(12), 10842–10849 (2024)
45. Yuan, W., Duan, J., Blukis, V., Pumacay, W., Krishna, R., Murali, A., Mousavian, A., Fox, D.: Robopoint: A vision-language model for spatial affordance prediction in robotics. In: CoRL. PMLR (2025)
46. Zeng, A., Song, S., Welker, S., Lee, J., Rodriguez, A., Funkhouser, T.: Learning synergies between pushing and grasping with self-supervised deep reinforcement learning. In: IROS. pp. 4238–4245. IEEE (2018)
47. Zhang, J., Liu, H., Li, D., Yu, X., Geng, H., Ding, Y., Chen, J., Wang, H.: Dexgraspnet 2.0: Learning generative dexterous grasping in large-scale synthetic cluttered scenes. In: CoRL. PMLR (2024)
48. Zhong, Z., Golestaneh, S., Chamzas, C.: Activepusher: Active learning and planning with residual physics for nonprehensile manipulation. arXiv preprint arXiv:2506.04646 (2025)
49. Zhou, W., Held, D.: Learning to grasp the ungraspable with emergent extrinsic dexterity. In: CoRL. pp. 150–160. PMLR (2023)
50. Zhou, W., Jiang, B., Yang, F., Paxton, C., Held, D.: Hacman: Learning hybrid actor-critic maps for 6d non-prehensile manipulation. In: CoRL. pp. 241–265. PMLR (2023)
51. Zhu, J., Tie, C., Cao, X., Wang, Y., Guo, J., Chen, Z., Chen, H., Chen, J., Xiao, Y., Wu, R., et al.: Adaptpnp: Integrating prehensile and non-prehensile skills for adaptive robotic manipulation. arXiv preprint arXiv:2511.11052 (2025)