

Designing Pin-pression Gripper and Learning its Dexterous Grasping with Online In-hand Adjustment

HEWEN XIAO*, Dalian University of Technology, China

XIUPING LIU*, Dalian University of Technology, China

HANG ZHAO*, Wuhan University, China

JIAN LIU†, Shenyang University of Technology, China

KAI XU, National University of Defense Technology, China

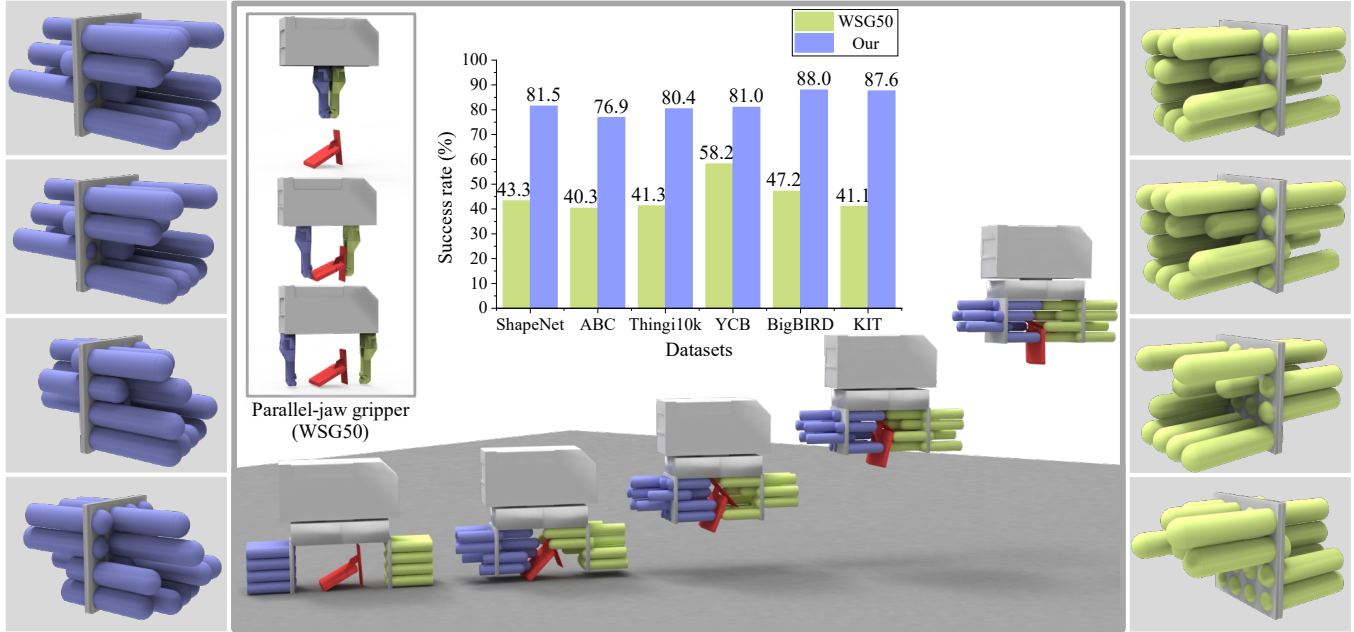


Fig. 1. The proposed pin-pression gripper offers object adaption and in-hand re-orientation through dynamically adjusting the extension and retraction of pins. The target object is successfully grasped and securely lifted with online pin adjustments on both fingers (shown on two sides) throughout the grasping process. As a comparison, we show the grasping of a parallel-jaw gripper (WSG50). The comparison of success rates tested on various datasets is also reported.

We introduce a novel design of parallel-jaw grippers drawing inspiration from pin-pression toys. The proposed pin-pression gripper features a distinctive mechanism in which each finger integrates a 2D array of pins capable of independent extension and retraction. This unique design allows the

gripper to instantaneously customize its finger's shape to conform to the object being grasped by dynamically adjusting the extension/retraction of the pins. In addition, the gripper excels in in-hand re-orientation of objects for enhanced grasping stability again via dynamically adjusting the pins. To learn the dynamic grasping skills of pin-pression grippers, we devise a dedicated reinforcement learning algorithm with careful designs of state representation and reward shaping. To achieve a more efficient grasp-while-lift grasping mode, we propose a curriculum learning scheme. Extensive evaluations demonstrate that our design, together with the learned skills, leads to highly flexible and robust grasping with much stronger generality to unseen objects than alternatives. We also highlight encouraging physical results of sim-to-real transfer on a physically manufactured pin-pression gripper, demonstrating the practical significance of our novel gripper design and grasping skill. Demonstration videos for this paper are available at <https://github.com/siggraph-pin-pression-gripper/pin-pression-gripper-video>.

CCS Concepts: • **Computing methodologies** → **Shape analysis**.

Additional Key Words and Phrases: Robotic gripper design, dexterous grasping, reinforcement learning

*Joint first authors.

†Corresponding author.

Authors' addresses: Hewen Xiao, Dalian University of Technology, China, hewenxiao@mail.dlut.edu.cn; Xiuping Liu, Dalian University of Technology, China, xpliu@dlut.edu.cn; Hang Zhao, Wuhan University, China, alexfrom0815@gmail.com; Jian Liu, Shenyang University of Technology, China, jianliu2006@gmail.com; Kai Xu, National University of Defense Technology, China, kevin.kai.xu@gmail.com.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2025 Association for Computing Machinery.

0730-0301/2025/8-ART \$15.00

<https://doi.org/10.1145/3730880>



Fig. 2. A picture of a pin-pression toy (image from the Internet). The toy features a 2D array of pins capable of independent extension and retraction. When pressing over the pins, an imprint of the hand is formed, which is naturally adaptive to the hand shape.

ACM Reference Format:

Hewen Xiao, Xiuping Liu, Hang Zhao, Jian Liu, and Kai Xu. 2025. Designing Pin-pression Gripper and Learning its Dexterous Grasping with Online In-hand Adjustment. *ACM Trans. Graph.* 44, 4 (August 2025), 17 pages. <https://doi.org/10.1145/3730880>

1 INTRODUCTION

The realm of dexterous robotic manipulation has undergone notable advancements lately. Currently, research endeavors predominantly focus on acquiring dexterous grasping skills tailored to existing gripper designs [Mahler et al. 2017]. However, a conspicuous scarcity exists in the literature that addresses the dual challenge of simultaneous gripper designing and grasping skill learning for maximized performance. The existing works on gripper design primarily revolve around refining parallel-jaw grippers [Honarpardaz et al. 2017]. This refinement centers on the optimization of the two fingers' shape, aiming to improve grasping performance tailored to specific object instances or categories (e.g., [Ha et al. 2020]). Thus, the capability of these grippers can hardly be universally generalized to novel, unseen objects.

We introduce a novel design of parallel-jaw grippers featuring dynamically adjustable finger shapes, aiming to enhance grasping adaptability to objects of arbitrary shapes. Our design draws inspiration from the captivating pin-pression toy (see Figure 2). Our proposed gripper, coined *pin-pression gripper*, features a distinctive mechanism in which each finger integrates a 2D array of pins capable of independent extension and retraction (see Figure 1). This design allows the gripper to instantaneously customize its fingers' shape to conform to the object being grasped by dynamically adjusting the extension and retraction of the pins. Moreover, our gripper excels in in-hand re-orientation of objects, similar to [Andrychowicz et al. 2020], achieving enhanced stability in grasping, again through online adjusting the pins. This design endows our gripper with both flexibility and robustness, elevating its proficiency in adeptly grasping a diverse range of objects. A similar design dates back to [Scott 1985] which, however, does not allow active adjustment of pins for target adaption. Such active adjustment requires a controller.

Learning the control skill of a pin-pression gripper for dexterous grasping and in-hand manipulation is a non-trivial task since it has $6 + N$ DoFs with N being the total number of pins on both fingers.

In this work, we focus on a top-down grasp policy in which the gripper approaches the target object from the top, forms a closure against the object with pin adjustments, and lifts the object while performing some in-hand re-orientation to improve the grasping stability if needed. To this end, we devise a dedicated reinforcement learning (RL) procedure to learn a policy model that encompasses both grasping and in-hand manipulation and is able to generalize to unseen objects with arbitrary shapes.

We learn our grasping policy network using the off-policy RL algorithm Soft Actor-Critic (SAC) [Haarnoja et al. 2018], with carefully designed state representation and rewards. In particular, the state representation of our policy model consists of the surface geometry (3D point cloud) of the target object, the interaction between the pins and the object surface (pin-surface distance and the amount and direction of protrusion of each pin), as well as the pose of the gripper. Our reward is designed to encourage successful and high-quality grasps while punishing long execution time and redundant movement of pins. The actions include extending and retracting pins, lifting the gripper, and stopping. The grasping policy is learned in the PyBullet simulation environment via trial and error.

There are two different modes of grasping for a pin-pression gripper: *Grasp-then-Lift* (GtL) and *Grasp-while-Lift* (GwL). GtL is basically a static grasping policy where the gripper forms a force closure of the object and then lifts it without further adjustment. In contrast, GwL performs pin adjustment dynamically throughout the process of grasping and lifting. While the former is easier to learn, the latter is more efficient since the gripper does not need to wait for the finish of ground adjustments. In some cases, GwL leads to more stable grasps since the gripper may find a better closure via unveiling the bottom of the target object when it is partially lifted. Albeit more efficient and effective, GwL is considerably more challenging to learn than GtL due to the more complicated grasping process. Our goal is to learn a grasping policy that could produce both GtL and GwL motions, whichever is more efficient. To do so, we opt for a curriculum learning strategy.

In particular, we adopt a two-stage training process with a properly aggregated replay buffer. In the first stage, we learn the GtL policy where the gripper is allowed only for ground adjustments, and no further adjustment is conducted once the object is lifted. After the learning, we collect all the state-action trajectories (experiences) into a self-exploration replay buffer. Bootstrapped with the replay buffer, the second stage continues to learn a mixed policy admitting both GtL and GwL, while enriching the replay buffer with both experiments. To further encourage the policy to include more air adjustments, we inject the replay buffer with experiences of air-only grasping collected from a policy pre-trained with only air adjustments. Such a mixed replay buffer directs the policy model to gain proficiency in both GtL and GwL skills.

We have conducted extensive experiments to evaluate the flexibility, stability, and generality of our gripper design along with the learned grasping policy. A notable result is that our gripper attains robust and efficient grasping on unseen objects with significant variation of geometry and topology, and under arbitrary poses. We have also performed quantitative evaluations on different physics simulators such as PyBullet [Coumans and Bai 2016] and Isaac Gym [Makoviychuk et al. 2021]. We have also compared our

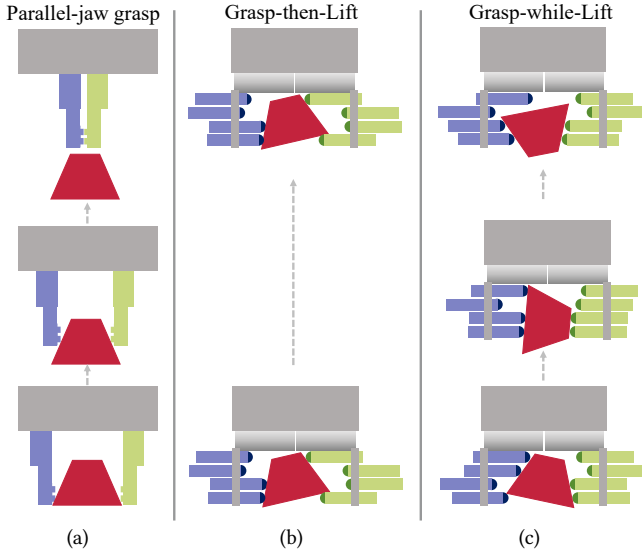


Fig. 3. The parallel-jaw gripper (a) finds difficulty in grasping the trapezoid. Our pin-pression gripper (b and c) achieves shape-adaptive grasping with dynamic adjustment of pins. In the grasp-then-lift (GtL) mode (b), the gripper can form a better closure of the trapezoid by partially lifting it. The grasp-while-lift (GwL) mode (c) allows for dynamic adjustment of pins throughout the process of grasping and lifting and therefore has the opportunity to more tightly lock the trapezoid via in-hand re-orientation.

gripper with several alternatives with state-of-the-art grasping algorithms and demonstrated consistent superiority in terms of success rate on both seen and unseen objects. Furthermore, we have showcased encouraging physical results of sim-to-real transfer through physically manufacturing a pin-pression gripper with electrically controlled pins and employing a two-stage teacher-student training paradigm. Real robot platform experiments demonstrated that our novel gripper design and grasping skill learning perform well in the real world. In summary, our contributions include the following.

- A novel gripper design capable of dynamically adjusting finger shape for object shape adaption and in-hand re-orientation.
- A dynamic grasping policy model as well as a curriculum learning scheme achieving robust and efficient grasping with both ground and air adjustment.
- Extensive evaluations and physical experiments conducted on a real robot platform demonstrating the efficacy and practical significance of our gripper design and grasping skill learning.

2 RELATED WORK

Finding effective ways of manipulating diverse objects has been a long-standing pursuit both in the virtual (e.g., animation) and the real world (e.g., robotics). Our work addresses a particularly challenging form of object manipulation—simultaneous grasping and reorientation—by integrating a novel gripper design with an efficient learning-based control policy for such dexterous manipulation. The unique gripper design and the associated effective policy learning together enable effective dynamic interactions between the gripper and arbitrarily-shaped objects. This highlights the synergy between designing and learning which may offer insights relevant to both robot structure design [Kodnongbua et al. 2022; Zhao et al.

2020] and gripper-object interaction [Christen et al. 2024; She et al. 2022], as well as their joint optimization [Geiling et al. 2018].

2.1 Robotic Gripper Design

The existing works on gripper design primarily revolve around refining parallel-jaw grippers [Honarpardaz et al. 2017], centering on the optimization of the two fingers' shape. These parallel-jaw grippers are typically designed based on the expert experience [Robotiq 2022; SCHUNK 2023]. To enhance task performance, the evolving industrial landscape demands customized gripper shapes tailored to specific target objects. Early research focuses on modular or re-configurable design methods, selecting appropriate basic components from a finger library to assemble modular fingers or refining fingertip shapes through direct formulaic methodologies [Balan and Bone 2003; Brost and Peters 1996; Brown and Brost 1999; Schroeffner et al. 2019; Zhao et al. 2025]. Some efforts turn to directly introducing external force to enable the automatic adjustment of finger shapes [Robotics 2019; Scott 1985]. The OmniGripper [Scott 1985] utilizes vertical pin arrays to wrap around and conform to the object passively, requiring a dense drive mechanism of 254 pins, and subsequent works [Fu et al. 2017; Fu and Zhang 2019; Mo and Zhang 2017, 2019] have led to improvements. These gripper fingers typically operate in a fixed mode, lacking adaptability to different shapes.

Soft grippers also demonstrate remarkable potential in grasping tasks, leveraging the inherent flexibility and mechanical compliance of soft materials, usually through three primary mechanisms: actuation, controlled stiffness, and controlled adhesion [Shintake et al. 2018]. The grippers like SDM Hand [Dollar and Howe 2010] address the inherent uncertainty of unstructured environments by incorporating compliance and adaptability into the hand's mechanical structure. The deformable grippers like VERSABALL [Robotics 2019] can leverage positive and negative pressure to grasp objects. Such soft grippers do not need any feedback or computation to adjust to different shapes. However, these grippers usually impose stringent requirements on the material, such as the elastic membrane and the encapsulated granules [Amend et al. 2012]. Our gripper does not rely on specific materials but instead achieves active shape compliance through rigid body motion.

Recently, the revolution in machine learning technology has provided new insights for task-specific gripper design, reducing reliance on expertise. Given the object to be grasped, Fit2form [Ha et al. 2020] proposes a 3D generative framework that utilizes data-driven algorithms to automatically generate finger shapes for parallel-jaw grippers, enabling stable and robust grasps. Despite demonstrating satisfactory grasping performance for specific object categories, Fit2form still encounters challenges in *rapidly adapting to novel objects* and requires a large amount of training data. ArrayBot [Xue et al. 2024] utilizes 16×16 sliding pillars to vertically perceive and support the target objects for re-positioning. Unlike Arraybot's reliance on a large number of actuators, our gripper system relies solely on a 4×4 array of pins to actively adjust its finger shape, thus acquiring dynamic grasping and in-hand manipulation capabilities through online pin extension and retraction. This also enables the handling of challenging objects like thin-shaped objects and trapezoids.

2.2 Robotic Grasping Policy

Analytical grasping. Robotic grasping policies can be generally categorized as analytical or learning-based techniques, with the latter further divided into supervised and reinforcement learning methods. Analytical approaches typically analyze the known geometry of a target object and obtain the optimal grasps with given quality metrics such as form closure or force closure [Bicchi 1995; Zhu and Wang 2003]. Some works search for grasping poses with the given configuration by minimizing torque or frictional forces [Miller and Allen 2004] while other efforts compute the contact points or contact areas that ensure the optimal grasp quality and subsequently direct fingers to reach the optimized positions [Jia 2004; Pan et al. 2021; Zhu and Wang 2003]. Contrary to the sample-based methods above, the introduction of differentiable losses [Kiatos and Malassiotis 2019; Maldonado et al. 2010] drives the continuous optimization techniques that apply gradient descent to directly find the optimal grasp configurations. A recent work by Liu et al. [2020] introduces a differentiable grasp quality metric applicable to the high-DOF gripper, enhancing the efficiency of continuously exploring optimal grasps. However, these analytical methods still face challenges in obtaining an online and real-time grasp response.

Supervising learning for grasping. Supervised learning methods involve using deep neural networks to obtain the 2D or 3D grasp configuration for the gripper. These methods rely on the labeled data, which can be acquired through various techniques such as human demonstration [Chu et al. 2018; Lenz et al. 2015], physics simulation [Depierre et al. 2018], physical tests [Pinto and Gupta 2016], and analytical computation [Fang et al. 2020]. Leveraging these datasets, discriminative methods usually train a neural network to predict the grasp score of sampled grasping candidates and then select the best one [Levine et al. 2016; Mahler et al. 2017, 2016; Song et al. 2020]. Generative approaches directly output a grasp configuration based on 2D or 3D information [Morrison et al. 2018; Mousavian et al. 2019; Redmon and Angelova 2015; Sharma and Valles 2020]. Recently, to overcome the limitations of grippers' adaptability, several data-driven approaches, albeit demanding high computational costs or expensive data generation, have concentrated on learning to adapt various types of grippers and autonomously perform grasp execution [Mun et al. 2023; Shao et al. 2020; Xu et al. 2021]. Our research enhances the adaptability of grippers by employing an innovative shape design and enables learning a dynamic grasping policy through trial and error, without the need for extensive demonstration data.

Reinforcement learning for grasping. Deep Reinforcement Learning (DRL) [Arulkumaran et al. 2017], offering a counterpoint to the robot grasping paradigm, has gained extensive attention in recent years. Various vision-based approaches have been developed [Du et al. 2021; Quillen et al. 2018] to perceive and interact with the environment. Kalashnikov et al. [2018] and Bodnar et al. [2019] adopt the self-supervised reinforcement learning framework to perform closed-loop grasping. DRL also facilitates grippers' development of dynamic policies with varying mechanisms (like pick [Shao et al. 2019] or push [Dengler et al. 2022]). Zeng et al. [2018] present a synergy parallel-jaw grasping system, which designs two fully

convolutional networks that map from visual observations to pushing and grasping actions. Wang et al. [2024b] employ a dual RL model to learn a policy composed of grasping and pushing. Deng et al. [2019] utilize the pushing strategy to explore the environment to obtain an affordance map and adopt Deep Q-Network (DQN) to evaluate the map of suction point candidates. Recently, Imtiaz et al. [2023] design the prehensile and non-prehensile actions and employ a memory-efficient DQN to output the map of each action. Our work designs allow both grasp-then-lift and grasp-while-lift motions, enabling the dynamic and flexible grasping policy learned by the RL approach.

Curriculum learning. For complex or dexterous grasping tasks, recent studies have delved into leveraging auxiliary tasks or human demonstrations to enhance the performance of DRL [Rajeswaran et al. 2017; Wang et al. 2022; Zhu et al. 2018]. To solve the reach-and-grasp problem, She et al. [2022] adopt the Soft Actor-Critic algorithm [Haarnoja et al. 2018] to learn a grasp policy from a double replay buffer mechanism with self-exploration data and imperfect demonstrations. Rajeswaran et al. [2017] explore how to combine DRL and the demonstration for learning dexterous manipulation. Integrating curriculum learning with reinforcement learning can also enrich grasping policy [Ecoffet et al. 2019; Narvekar et al. 2020]. Xu et al. [2023] propose a generalizable dexterous grasping method with the PPO algorithm [Schulman et al. 2017] and the DAgger imitation learning algorithm [Ross et al. 2011a], leveraging a 3-stage object curriculum learning. Further, Wan et al. [2023] propose a geometry-aware curriculum and iterative generalist-specialist learning method to improve performance. Similarly, Zhang et al. [2023] achieve natural and robust in-hand manipulation of simple objects through deep reinforcement learning. They further facilitate the adaptation of these skills to more complex shapes through curriculum learning. Our research focuses on tackling the challenge of gripper design and grasping skill learning concurrently. We apply curriculum learning techniques to learn the trade-off between grasp-then-lift and grasp-while-lift motion and solve the dynamic grasping problem solely based on the gripper's self-exploration experience, eliminating the need for external guidance like human demonstrations.

3 OVERVIEW

Given the target object, the goal of this work is to output a sequence of in-hand manipulations that facilitates gripper grasping with a high success rate and efficiency. We introduce a novel parallel-jaw gripper design featuring dynamic finger shape adjustments (illustrated in Figure 5). Our gripper differentiates itself by a distinctive mechanism in which each finger integrates a 2D array of pins capable of independent extension and retraction. This design allows the gripper to instantaneously customize its shape to conform to the target object by online adjustment, i.e., extension and retraction of the pins. We devise an effective reinforcement learning (RL) procedure to learn policy models via trial and error. A curriculum training fashion as well as the careful designs of state representation and reward shaping are adopted to enable the grasping policy to produce both *grasp-then-lift* (GtL) and *grasp-while-lift* (GwL) motions, possessing flexibility, stability, and generality.

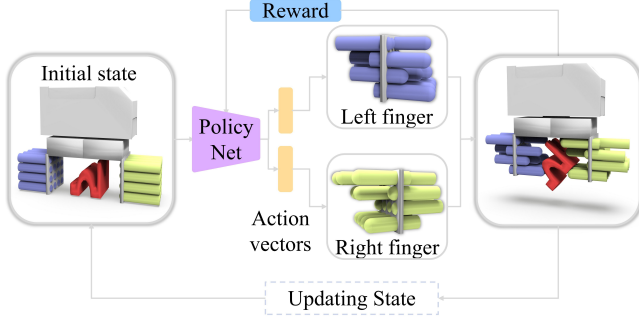


Fig. 4. Overview of our learning-based approach. Our method obtains the current state information about the object, gripper, and their interaction to predict the appropriate action that moves the pins of the gripper finger in GtL or GwL grasping modes. After executing the online predicted action, the updated state is then passed through the same pipeline to predict the next action, so that a successful grasp is gradually formed.

The grasping task can be modeled as a Markov Decision Process (MDP) [Bellman 1957] $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$, where \mathcal{S} represents the state space, \mathcal{A} is the action space, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \mapsto \mathcal{S}$ is the state transition function, and $\mathcal{R} : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ is the reward function. Figure 4 illustrates the pipeline of our learning-based approach. The learning starts by obtaining the current state representation s_t , encompassing the surface geometry of the target object, the pose of the gripper, and the interaction between the pins and the object surface. A set of features is extracted from s_t to predict the action a_t , adjusting the extension and retraction of the pins so that the finger shape actively adapts to the surface of the target object and ensures a physically reliable grasp. After the execution of the predicted action a_t , the updated state undergoes the same pipeline to determine the subsequent action. We employ a policy network learned through RL, specifically utilizing the Soft Actor-Critic method [Haarnoja et al. 2018], with carefully designed grasp rewards and efficiency rewards enhancing the learning process. We also employ a two-stage curriculum learning approach, which injects the replay buffer with experiences (state-action trajectories), to achieve gripper grasping capable of both GtL and GwL motions.

4 METHOD

In this section, we first introduce our novel pin-pressure gripper design in Section 4.1. Subsequently, we delve into the details of the proposed learning method, covering the state and action representations in Section 4.2, the reward formulation and network architecture design in Section 4.3, and the two-stage curriculum learning technique in Section 4.4.

4.1 Design of the Pin-pressure Gripper

The robot end-effector is an essential interface between the robot and the target object, deeply impacting the overall performance of industrial robotic grasping tasks. The prevailing research [Mahler et al. 2017; Zeng et al. 2018] has predominantly focused on learning grasping policies for universal parallel-jaw grippers, where the designed shape insufficiently adapts to various target shapes. Although a recent data-driven approach [Ha et al. 2020] generates the customized 3D gripper finger tailored to different target objects,

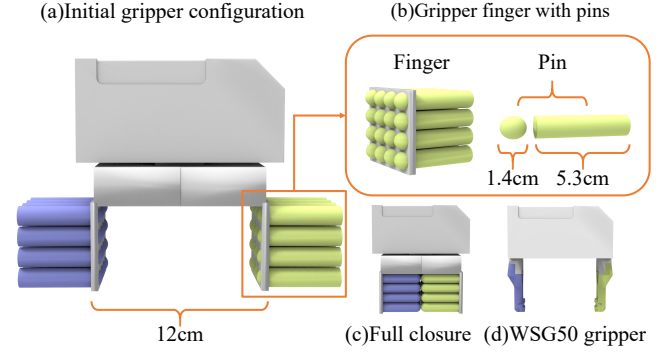


Fig. 5. The specific constitution of the parallel pin-pressure gripper. The gripper finger is a 2D array of pins capable of independent extension and retraction. Each pin consists of a cylinder and a sphere. The gripper achieves full closure when every finger reaches its maximum forward movement.

it cannot adapt well to the variations in the object's poses during the grasping process, let alone adapt to shapes it has never seen before. Consequently, the challenge of designing a gripper that adaptively attains robust and efficient online grasping on unseen objects persists.

To tackle this challenge, we strive to design an innovative gripper, coined *pin-pressure gripper*, which is geometry-aware and capable of automatically adjusting its shape. Similar to Fit2form [Ha et al. 2020], we employ a general parallel-jaw gripper as the foundation since its modular design facilitates effortless customization of the shape of the gripper fingers. We observe a pin-pressure toy, as depicted in Figure 2, that inherently possesses both geometry awareness and the ability to adaptively adjust its shape. Such a toy can create impressions by pressing objects against its plastic pins, resulting in instant 3D images. Inspired by this, we integrate each gripper finger with a 2D array of pins, as illustrated in Figure 5a for the initial gripper configuration. This grants our gripper capable of independent extension and retraction to instantaneously customize its fingers' shape to conform to the target object.

More specifically, each gripper finger is made up of 4×4 cylinder pins with a height of 6cm. The pin tip for interacting with objects is a sphere with a radius of 0.7cm. For specific configurations regarding fingers and pins, please refer to Figure 5b. In this pin-pressure gripper, the two sets of opposing pins can be extended and retracted by using a linear motion system [Kurfess et al. 2005], thereby performing grasps. In practice, each pin has a maximum movement distance of 5.5cm, and the gripper achieves full closure when every pin reaches its maximum forward movement, as depicted in Figure 5c.

For the designed gripper, each grasp pose is characterized by a $(6+32)$ -DOF configuration of the gripper. The first 6-DOF is the rigid transformation of the gripper, while the remaining 32-DOF is the stretching value, i.e., the extension length of the pins. Despite the gripper, serving as the end effector of the robotic arm, is able to move freely within a 6-DOF space, we primarily focus on dynamic grasping with a fixed top-down grasping manner. That is, the gripper approaches the target object from the top, forms a closure against the object through online adjustments, and lifts the object while

performing in-hand re-orientation to improve the grasping stability if needed. Unlike grippers with specialized fingers, such as the WSG50 gripper depicted in Figure 5d, our pin-pression gripper inherently incorporates the variances in object geometry and pose through adaptively adjusting finger shapes.

4.2 State and Action Representation

State representation. An effective state representation is crucial for guiding the gripper to achieve a physically reliable grasp concerning the geometry and pose of the target object. It's well-known that appropriately fusing multi-modal information as input can significantly outperform single-mode data [Fazeli et al. 2019; Lee et al. 2020; Miki et al. 2022; Zhang et al. 2024; Zhou et al. 2024]. This demonstrates that providing networks with sufficient information can enhance the performance of learned policies. Consequently, we carefully design comprehensive state representations for enhancing online grasping. We recognize that the interaction between the pins and the target object surface provides informative descriptions regarding the spatial variances. Therefore, our state vector s_t at timestep t (t is omitted in the subsequent discussion), involves object information o , gripper information g , and gripper-object interaction information f :

$$s = [o, g, f]. \quad (1)$$

We elaborate on each component of the state s . The object information o is structured as a triplet $o = [o_p, o_r, o_f]$, where o_p denotes the object position, o_r represents the object orientation represented using a quaternion, and o_f is the feature vector derived from the object point cloud, which provides a rich representation of the object's 3D geometry characteristics [Chen et al. 2023; Su et al. 2023; Wang et al. 2024a]. We employ DGCNN [Wang et al. 2018] to extract o_f , which is a 512-element vector. It can also be effortlessly adapted to use other alternative extractors owing to the modular structure of our network. In summary, the entire object information o is a 519-element vector.

The gripper information g for describing the gripper is formatted as a column vector, which is presented as:

$$g = [g_p, g_p^o, g_r, g_r^o, g_{\text{stage}}, g_{\text{step}}]. \quad (2)$$

Here g_p and g_r are the position and orientation of the gripper in the world frame, while g_p^o and g_r^o are in the object frame of reference. The 3-element one-hot vector g_{stage} indicates the current stage, distinguishing between ground adjustment, air adjustment, and adjustment completion. The C -element one-hot vector g_{step} indicates the current step number. Thus, the gripper information g is a $(14+3+C)$ -element vector. In practice, we set C as 11.

Capturing the dynamic changes in the relationship between the gripper and the object is crucial for learning effective gripper grasping policies. Recently, She et al. [2022] leverage Interaction Bisector Surface (IBS) to represent gripper-object interaction in the context of reach-to-grasp planning for dexterous hands. However, the computational overhead for IBS extraction is inherently substantial. Therefore, we opt for a faster and more efficient approach to characterize the gripper-object interaction, which is better suited for our pin-pression gripper.

For each pin p of our gripper, we capture the following details to form the gripper-object interaction information f :

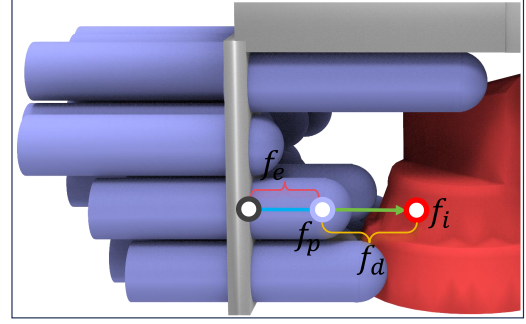


Fig. 6. Informative dynamic interaction representations of each pin p , including a set of relationship information between the target object and the gripper. Detailed explanations of the notations can be found in Section 4.2.

- Pin position $f_p \in \mathbb{R}^3$ in the world frame, represented by the tip center of p .
- Pin position $f_p^o \in \mathbb{R}^3$ in the object frame.
- Point of intersection $f_i \in \mathbb{R}^3$ in the world frame, obtained by extending a horizontal line from the tip to the object surface.
- Point of interaction $f_i^o \in \mathbb{R}^3$ in the object frame.
- Distance $f_d \in \mathbb{R}$ between the tip and the object surface.
- Pin stretching value $f_e \in \mathbb{R}$.
- Indicator $f_b \in \{-1, 1\}$ informing which finger p belongs to.
- One-hot indicator $f_c \in \{0, 1\}^{32}$ informing the pin index.

We provide visualizations of the gripper-object interaction information components in Figure 6. The total length of f is 1504 dimensions. The final state $s = [o, g, f]$ represents a geometry-aware depiction of the current task scene, allowing for rapid computation and accurate captures of the dynamic changes in the relationship between the gripper and the target object.

Action representation. A hybrid representation of the gripper action, integrating both GtL and GwL motions, is imperative for learning grasping skills with adaptive adjustments. To this end, the action a is composed of three parts:

$$a = [l, \chi_1, \chi_2]. \quad (3)$$

The first part l denotes the stretching value of each pin, constituting a 32-element vector confined within $[0.0\text{cm}, 5.5\text{cm}]$. This means that during the grasping process, our gripper can be flexibly extended and retracted for online in-hand adjustment and better grasping. The subsequent part is the switch action χ_1 , serving as a binary switch value to indicate lifting the gripper, and the potential transition from the GtL motion to the GwL motion. Here, $\chi_1 = 0$ corresponds to the former, and $\chi_1 = 1$ to the latter. The final part is the stop action χ_2 serving as a termination signal for concluding the grasping. The termination criteria is elucidated in Section 4.3. Therefore, an action is represented as a 34-element vector. More implementation details can be found in Section A of supplemental material.

4.3 Reward Signal and Policy Network Design

Reward function. An appropriate reward function facilitates learning adaptive grasping policies. Given our primary purpose of ensuring the secure grasp of the target object without falling, we first formulate a *grasp reward* r_{grasp} to evaluate the grasp quality

once the entire grasping procedure is completed. We further introduce another *efficiency reward* r_{effici} to encourage minimizing unnecessary pin movements.

Similar to She et al. [2022], we quantify the grasp quality by considering both the signal $G \in \{-1, 1\}$ indicating whether the final grasp was successful, and the generalized $Q1$ analytic measure for grasp stability proposed by Liu et al. [2020]. Consequently, the grasp reward function is formulated as follows:

$$r_{\text{grasp}} = \omega_1 G + \omega_2 Q1 + r_{\text{time}}, \quad (4)$$

where $\omega_1 = 2000$ and $\omega_2 = 1000$ in all our experiments. An additional execution time term $r_{\text{time}} = \omega_3 T_{\text{grasp}} + \omega_4 T_{\text{lift}}$ is included to guide the policy in completing tasks promptly. The weight ω_3 for grasping time T_{grasp} before lifting the object is set to -250, while the weight ω_4 for the subsequent lifting time T_{lift} is set to -50. Note that the entire grasp reward will only be calculated once the grasping task is finalized, i.e., r_{grasp} is a terminal reward.

To encourage efficient grasping, we would like to limit redundant pin movement as much as possible. Hence, we restrain the total length of all pin movements r_{len} and the number of executed steps r_{step} . We formulate the efficiency reward r_{effici} , serving as an intermediate reward, as follows:

$$r_{\text{effici}} = \omega_5 r_{\text{len}} + \omega_6 r_{\text{step}}. \quad (5)$$

In all our experiments, we consistently set $\omega_5 = -1$ and $\omega_6 = -10$.

Network architecture and policy training. For the state representation, we extract features from the object point cloud and concatenate them with features obtained from gripper-object interaction and gripper information. The concatenated feature is then fed into the Multi-Layer Perceptrons (MLPs) to predict action a , which consists of the gripper configuration l , the switch value χ_1 , and the terminal value χ_2 . The grasping process terminates when either a sampled value of χ_2 is greater than a threshold or a maximum number of steps is reached. We adopt DGCNN [Wang et al. 2018] for object point cloud encoding, PointNet [Charles et al. 2017] for gripper-object interaction encoding, and MLPs for gripper information encoding. More details about the network architecture can be found in Section A of the supplemental material.

To train the policy network, we adopt the off-policy Soft Actor-Critic (SAC) method [Haarnoja et al. 2018]. Given the state s as input, the actor network outputs the policy $\pi(\cdot|s, \theta)$ as a Gaussian distribution for sampling actions. The critic network takes both the state s and action a as inputs and outputs the state-action value for assisting the policy training. She et al. [2022] have proved that outputting a vector by the critic network can make the training more effective than outputting a single value. Therefore, we separately approximate the accumulated grasp reward r_{grasp} and efficiency reward r_{effici} with a vector (Q_g, Q_e) output by the critic. The loss function for training the critic network is based on the TD update approach:

$$L_Q(\beta) = [(Q_g(s, a; \beta) - y_g(r_{\text{grasp}}, s', d, Q_g(s', \tilde{a}', \beta')))^2 + (Q_e(s, a; \beta) - y_g(r_{\text{effici}}, s', d, Q_e(s', \tilde{a}', \beta')))^2]. \quad (6)$$

Here the target value y_g is formulated as:

$$y_g(r, s', d, Q) = r + \gamma(1 - d)[Q - \alpha \log \pi(\tilde{a}|s'; \theta)], \quad (7)$$

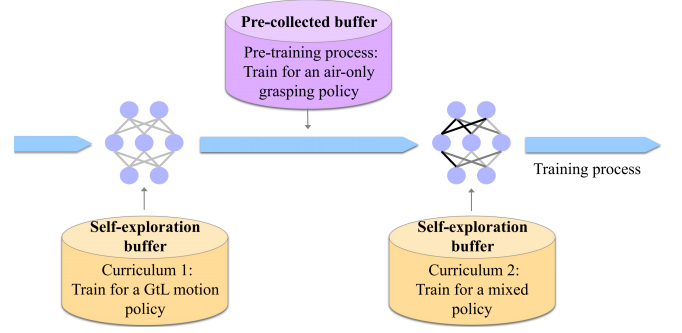


Fig. 7. Our learning framework consists of a two-stage curriculum (colored in yellow) and a pre-training process (in purple) for collecting GwL experiences. The curriculum learning progressively guides the policy that extends the ability of gripper manipulation from GtL motion to GwL motion. To further encourage the policy to include more air adjustments, we inject the replay buffer with experiences of air-only grasping, collected from a policy pre-trained with only air adjustments in the learning of the second stage.

where \tilde{a}' is the action sampled from the policy based on the next state s' , γ denotes the discount factor, and the temperature parameter α represents the relative importance of the entropy term against the reward. The training loss for the actor network is defined as:

$$L_Q(\theta) = Q_g(s, \tilde{a}(s; \theta)) + Q_e(s, \tilde{a}(s; \theta)) - \alpha \log \pi(\tilde{a}|s; \theta), \quad (8)$$

where \tilde{a} represents the action sampled from the policy with state s and the parameter θ will be updated.

4.4 Curriculum Learning

In order to allow our pin-pressure gripper to adaptively grasp objects with both GtL and GwL modes, we adopt a two-stage curriculum learning method with a properly aggregated replay buffer. The curriculum learning gradually bootstraps the policy to gain grasping proficiency, progressing the grasp from the GtL motion to the GwL motion. We illustrate our learning process in Figure 7. In the first stage of the curriculum, we learn the GtL policy where the gripper is allowed only for ground adjustments, and no further adjustment is allowed once the object is lifted. All experiences in this stage are recorded into a self-exploration replay buffer, which also assists the second stage curriculum. The second stage continues to learn a mixed policy admitting both GtL and GwL modes while enriching the replay buffer with both experiments. In addition to the GtL data inherited from the first stage, we also inject the replay buffer with experiences of air-only grasping collected from a policy pre-trained with only air adjustments (colored purple in Figure 7) to further encourage the policy to include more air adjustments. Such a mixed replay buffer directs the policy model to gain proficiency in both GtL and GwL skills.

More specifically, we utilize two replay buffers for each curriculum learning stage: one for the self-exploration data with a maximal size set at $m = 50000$, and the other for the pre-collected GwL data with a maximal size set at $n = 5000$. Each replay buffer is maintained as a first-in, first-out queue. We first initialize an empty buffer to start the first stage curriculum and continuously append experiences to the self-exploration buffer. In the second stage of learning, we leverage both the self-exploration buffer and the pre-collected

buffer to enhance the policy’s capability for cross-motion grasping. Throughout the training phase, the self-exploration buffer continuously updates with new experiences, while the pre-collected buffer remains fixed without further updates. The probability of sampling data from each buffer is directly proportional to their capacity. Our two-stage training approach enables the network to capitalize on abundant experiences and strike a balance between success rate and task execution cost.

5 EXPERIMENT

We first introduce our experimental settings in Section 5.1. Our method achieves notable performance in online grasping, rooted in the careful designs of action planning, state representations, and reward shaping. We conduct ablation studies in Section 5.2 to investigate the contributions of these components. Subsequently, we provide comparisons with alternative methods in Section 5.3 to elucidate the efficacy of our approach, particularly when compared with the passive grasping method. Finally, in Section 5.4, we delve into a more in-depth analysis of our pin-pression gripper and examine the performance of both sim-to-sim transfer and sim-to-real transfer under a different physics engine and on a real robot platform, respectively.

5.1 Experimental Settings

Data preparation. To train a reliable policy for dynamically grasping 3D shapes of variations in geometry and topology, it’s necessary to generate a target object dataset containing diverse shapes with appropriate sizes while compatible with the gripper. Specifically, we collect our target objects from the 3DNet dataset [Wohlkinger et al. 2012], the BigBIRD dataset [Singh et al. 2014], the ShapeNet dataset [Chang et al. 2015], the YCB dataset [Çalli et al. 2015], the GD dataset [Kappler et al. 2015], the KIT dataset [Mandery et al. 2016], and the Thingi10K dataset [Zhou and Jacobson 2016]. These datasets contain abundant complex shapes from household objects to industrial components. For each target object, we settle it in a random orientation on the ground. We subsequently resize it to fit within a bounding box with a maximum side length of 5.5cm, centered at the initial grasp location. The resized object is then incorporated into our object dataset used for grasping. The resulting object dataset contains a total of 543 objects, with 446 allocated for training and 97 for testing. For a thorough gallery of object dataset, please see Figure 12 in supplemental material. We further test our method on over 5000 unseen objects collected from various datasets to demonstrate the generalization superiority of our pin-pression gripper compared to other alternatives. We also collect and contribute a set of 108 objects challenging for passively grasping approaches, categorized as the Chal-H and Chal-T datasets, to emphasize the necessity of the adaptive grasping approach.

Simulation setup. We evaluate the grasping performance in a simulation environment established using the PyBullet physics simulator [Coumans and Bai 2016]. Within this environment, a flat surface serves as the ground, an object is sampled from the target object dataset, and our pin-pression gripper is present. Before executing the simulation, the gripper is translated to a fixed position with its initial configuration, and the object is placed between the

two fingers of the gripper. Considering that variations in end load can impact the grasping outcome, we assume a consistent mass of all target objects to 50 grams each with a uniform density to focus on the gripper’s adaptability to the target object geometry. Following Fit2form [Ha et al. 2020], we set the lateral friction coefficient of 0.2, and the rolling friction coefficient of 0.001. We simulate the grasping process by manipulating the pins with a force of 0.5N applied during grasping. Our simulation environment simultaneously allows both the GtL and GwL grasping modes. To verify grasp success, we lift the object to a height of 30cm and then check if it has dropped. After a simulation episode is done, the gripper is reset to its initial state.

Evaluation metrics. As discussed in Section 4, our objective is to achieve adaptive grasping with both effectiveness and efficiency. To evaluate the grasp quality, we compute the *grasp success rate* (S) for the entire testing set, with the simulator determining whether the target object is dropped or not after the grasping task concludes. For all successful grasps, we calculate the average *generalized Q1* [Liu et al. 2020] as an analytical metric representing the overall physical stability. The $Q1$ value is greater than or equal to 0, and a higher value indicates a more stable grasp. To measure the grasping efficiency, we record the *running time* (T). We also measure the ratios of grasps through the GtL and GwL modes to intuitively understand the gripper’s grasping behavior.

5.2 Ablations on RL Components

To verify the impact of each component, we conduct ablation studies by systematically removing several key components. The corresponding quantitative results are presented in Table 1. We use the full version of our method as a benchmark, denoted as ‘Ours’.

Flexibility of action design. Implementing an action design that incorporates both GtL and GwL motions is essential for enabling the policy network to learn adaptive grasping skills that are both effective and efficient. We individually train the policies under each grasping mode. As depicted in Table 1, when only GtL motion is allowed for execution, represented as ‘Ours ($\chi_1 = 0$)’, the policy network achieves the highest success rate of 84.53% on the testing set but exhibits the lowest efficiency in terms of grasping task running time. This is primarily attributed to decoupling the actions of grasping and lifting objects into two independent stages, leading to additional task running costs about waiting for the finish of ground adjustments. On the other hand, when only GwL motion is considered, denoted as ‘Ours ($\chi_1 = 1$)’, the policy network yields the lowest success rate of 61.86% on the testing set, despite the shortest runtime. This is due to adjusting the object’s pose in the air being prone to dropping, unlike GtL, which has the ground as support and only needs to execute a static grasping. Compared with these two alternatives, our approach flexibly obtains an excellent balance between efficiency and efficacy, outperforming the baseline ‘Ours ($\chi_1 = 1$)’ by a large margin in success rate and surpassing the performance of the baseline ‘Ours ($\chi_1 = 0$)’ significantly in terms of task running cost.

Guidance of reward design. We adopt both grasp reward r_{grasp} and efficiency reward r_{effici} to inform the policy of grasping preferences. The grasping reward r_{grasp} is a combination of generalized

Table 1. Ablation studies of our method, where S denotes the success rate of the entire test or train set; $Q1$ is the mean generalized $Q1$ value of all successful grasps; T presents the average running time for all objects, totally 97 objects for the testing set and 446 for the training set; GwL (%) is the ratio of the testing objects using GwL mode; GtL (%) means the ratio of the grasping objects using GtL mode; We bold the best results in terms of S , $Q1$, and T .

Method	Testing set (Unseen objects)					Training set (Seen objects)				
	$S(\%) \uparrow$	$Q1 \uparrow$	$T(s) \downarrow$	GwL (%)	GtL (%)	$S(\%) \uparrow$	$Q1 \uparrow$	$T(s) \downarrow$	GwL (%)	GtL (%)
Ours	82.47	0.3456	6.54	64.95	35.05	82.96	0.3506	6.30	71.30	28.70
Ours ($\chi_1 = 0$)	84.53	0.3587	7.65	0.00	100.00	88.56	0.3561	7.59	0.00	100.00
Ours ($\chi_1 = 1$)	61.86	0.3603	6.00	100.00	0.00	74.44	0.3586	6.00	100.00	0.00
w/o G	58.76	0.3495	6.01	98.97	1.03	52.24	0.3515	6.01	99.77	0.23
w/o $Q1$	78.35	0.3412	6.53	78.35	21.65	81.16	0.3503	6.39	70.63	29.37
w/o r_{time}	78.35	0.3569	9.80	30.93	69.07	80.72	0.3474	9.07	44.84	55.16
w/o r_{len}	78.35	0.3554	6.70	50.52	49.48	80.49	0.3511	6.48	58.97	41.03
w/o r_{step}	76.29	0.3545	6.75	11.34	88.66	72.42	0.3483	6.72	21.75	78.25
w/o o	72.16	0.3385	6.21	75.26	24.74	78.03	0.3524	6.23	69.73	30.27
w/o g	81.44	0.3471	7.55	0.00	100.00	74.89	0.3446	6.81	0.00	100.00
w/o f	75.25	0.3550	6.71	5.15	94.85	79.37	0.3512	6.67	20.63	79.37
only f	69.07	0.3602	6.45	57.73	42.27	78.48	0.3533	6.55	59.42	40.58
w/o $world$	73.20	0.3384	6.23	89.69	10.31	79.82	0.3534	6.28	79.37	20.63
w/o $one\ hot$	79.38	0.3563	6.80	87.63	12.37	77.80	0.3489	6.72	80.49	19.51
w/o $pin\ pos$	77.32	0.3550	7.36	18.56	81.44	75.11	0.3542	7.03	27.13	72.87
RGB-D image	73.19	0.3609	6.87	15.46	84.54	71.75	0.3578	6.79	24.22	75.78
w/o CL	70.10	0.3537	6.06	95.88	4.12	74.66	0.3567	6.07	93.72	6.28
w/o pre	80.41	0.3375	6.76	57.73	42.27	81.84	0.3567	6.61	59.42	40.58
w/o $CL \& pre$	71.13	0.3457	6.19	80.41	19.59	78.48	0.3531	6.20	79.15	20.85
Reward CL	78.35	0.3345	7.14	21.65	78.35	79.82	0.3401	7.13	21.52	78.48

$Q1$, success signal G , and execution time r_{time} while we adopt r_{len} and r_{step} serving as efficiency reward to encourage grasping without unnecessary pin movements. We conduct an ablation study by separately removing these reward terms to investigate the impact of each component. The conducted experiments include: ‘w/o G ’ eliminates the success signal G ; ‘w/o $Q1$ ’ removes the generalized $Q1$ used for measuring grasp stability; ‘w/o r_{time} ’ omits r_{time} , designed for encouraging a decrease in execution time; ‘w/o r_{len} ’ involves the removal of r_{len} , designed for encouraging shortening the pin movements, and ‘w/o r_{step} ’ eliminates r_{step} , designed for encouraging few grasp steps. We can observe that disabling each single reward component adversely affects the performance of the grasping policy, both in the training and testing sets. The policy trained without G attains the lowest success rate among individual reward component ablations, emphasizing its pivotal role in guiding successful grasps. This importance arises from the primary goal of preventing the object from falling during pin movements. The results of ‘w/o $Q1$ ’ reflect that the removal of the $Q1$ reward decreases the grasp stability compared with ‘Ours’. Disabling the execution time reward r_{time} incurs the highest task running cost. Meanwhile, rewards r_{len} and r_{step} , designed to minimize pin movements, also contribute to the quick completion of the task.

Necessity of state representation. We underscore the significance of the proposed state representation, which integrates object information, gripper information, and gripper-object interaction, by separately training policies with disabling corresponding components. Specifically, ‘w/o o ’ disables the object information o , ‘w/o

g ’ disables the gripper information g , and ‘w/o f ’ disables the state representation of the gripper-object interaction f . We also provide results with only the state representation of the gripper-object interaction f , denoted as ‘only f ’. The removal of either g or f reduces the success rate while inclining the policy towards utilizing only the more time-consuming GtL method to accomplish tasks. The underlying rationale is that both g and f contain information about the gripper itself, removing either of them would diminish the policy’s ability to assess the current situation. This makes it more challenging to execute adaptive adjustments in the air.

To further validate the rationality of the detailed state representations mentioned in Section 4.2, we conduct three additional ablation versions, including ‘w/o $world$ ’ denoted as the elimination of utilizing the global coordinate system to represent the gripper information g and gripper-object interaction information f ; ‘w/o $one\ hot$ ’ as the removal of one-hot indicator f_c in f , and ‘w/o $pin\ pos$ ’ omitting the pin positions in the world frame f_p and the object frame f_o . The decreases in the success rate of these experiments validate the importance of providing sufficient information for grasping execution. Additionally, we conduct a comparison to an alternative state representation using RGB-D images. From the results, one can observe that when using only RGB-D images as the full state, our method achieves a success rate of 73.19% on the testing set and 71.75% on the training set. This suggests that our method performs well even with this single-view partial observation, though not as well as the designed representation of state. We will elaborate on

Table 2. Success rate comparisons on artificially generated CAD and scanned object datasets. All shapes are unseen for our trained policies.

Method	CAD models			Scanned models		
	ShapeNet	Thing10K	ABC	YCB	BigBIRD	KIT
WSG50	43.28%	41.34%	40.28%	58.22%	47.20%	41.09%
Fit2form	78.82%	76.96%	68.12%	78.21%	84.00%	80.62%
Random	22.37%	28.86%	23.34%	35.44%	31.20%	24.03%
Passive	60.36%	67.51%	56.94%	60.76%	68.80%	55.81%
Ours	81.54%	80.42%	76.91%	81.01%	88.00%	87.60%

how our state representation can be applied in physical practice in Section 5.4.

Significance of curriculum learning. One of our key contributions is the introduction of a curriculum learning technique directing the policy model to gain proficiency in both GtL and GwL skills. To highlight the significance of our curriculum learning approach, we conduct specific ablations on different stages of the curriculum. We stipulate that ‘w/o *CL*’ denotes training without the two-stage curriculum; ‘w/o *pre*’ represents two-stage training without the pre-collected GwL experiences, ‘w/o *CL* & *pre*’ removes both the two-stage curriculum and the pre-collected experiences, and ‘Reward *CL*’ represents a curriculum framework based on reward shaping, where all grasping rewards except execution time are initially applied, and the time penalty is later introduced. As shown in Table 1, we can see that removing the two-stage curriculum impairs the flexibility of the policy to switch grasping modes. The policy achieves a successful grasp only through the more challenging GwL motions, which significantly reduce the grasp success rate. Eliminating the pre-collected GwL experiences leads to a relatively lower success rate on the testing set, given the lack of prior experience in air adjustments to guide policy learning. The same reasons apply to ‘w/o *CL* & *pre*’ which learns policies without guidance about both ground adjustments and air adjustments. ‘Reward *CL*’ also yields a lower success rate than our curriculum design, highlighting that our two-stage curriculum is better suited for facilitating learning across GtL and GwL skills.

5.3 Comparison Results.

We design an adaptive geometry-aware pin-pression gripper that can adjust its finger shapes according to the variations of the geometry and pose of the target object. To demonstrate the necessity of learned grasping policies and the effectiveness of our gripper, we make comparisons with other alternative grippers and the passive grasping approach.

Comparisons with alternative grippers. For general gripper grasping, previous methods often utilize either universal parallel-jaw grippers for general-purpose manipulation, designed without consideration of the target geometry, or task-specific grippers, which are geometry-aware and execute grasping by closing all gripper fingers. To facilitate a more comprehensive comparison, we categorize the baseline variants based on these two gripper design approaches and evaluate their success rates separately. For the general-purpose grippers, we compare our method with the WSG50

gripper [SCHUNK 2023], which features specialized components on its fingertips to enhance friction and increase the contact area for stable grasping. For the task-specific grippers, we compare our method with a 3D generative design approach, Fit2form [Ha et al. 2020], which generates pairs of finger shapes to execute grasps for fixed object poses. The experiment setting for baselines closely follows the implementation from Fit2Form and performs the top-down grasp. We also compare our method with multi-finger robot grippers, such as the two-finger gripper (EZGripper), the three-finger gripper (Barrett hand), and the dexterous hand (Shadow hand) (see Table 2 and Section C in supplemental material). Additionally, to further demonstrate the effectiveness of our learned policies, we compare our method with two other approaches: the random grasping policy, which randomly executes pin movements throughout the entire task process, and the passive grasping method, which adjusts finger shapes passively by directly applying external force.

We evaluate the grasp success rate of each approach on both Computer-Aided Design (CAD) and real-world scanning. For artificially generated CAD models, we test on the ShapeNet dataset, the Thing10K dataset, and the ABC dataset. For the scanned shapes, we test on the YCB dataset, the BigBIRD dataset, and the KIT dataset. Specifically, we randomly select 1511 shapes from ShapeNet, 1542 shapes from Thing10K, and 1728 shapes from ABC, as well as using all available shapes from the YCB dataset, the BigBIRD dataset, and the KIT dataset. Note that all shapes are unseen for the policies that only learn on the training set from Section 5.2.

Table 2 reports the quantitative comparisons to those alternative baselines on six different datasets. We also provide visual comparisons between our method and these baselines in Figure 8. Obviously, our method consistently outperforms all the baselines on all the datasets. We observe that these alternative grippers underperform against variations in object geometry or poses. In contrast, our method is capable of adaptively modifying its finger shape, guided by the interaction between the gripper and the object. It can also be observed that the random grasping policy may produce sparse and insufficient contacts between objects and the gripper fingers, failing to achieve perfect force closure for the target objects. Additionally, our method also achieves good performance across different grasping directions in terms of success rate (See Figure 3 in supplemental material). Overall, these comparative results further underscore the advantages of our geometry-aware pin-pression gripper design.

Comparison with passive grasping method. Previous efforts turn to directly introducing external force to passively adjust finger shapes [Robotics 2019; Scott 1985]. These gripper fingers typically operate in a fixed mode, facing challenges for shape variations. For example, flat objects usually offer limited contact areas for the gripper, especially when lying on the ground. Objects with inclined surfaces or tetrahedral shapes also present difficulties due to inadequate force closure. Inspired by these, we construct a challenging dataset to further demonstrate the necessity of our online grasping approach, i.e., grasping objects with adaptive gripper adjustment. The principle for constructing the dataset is that the objects are either flat or have inclined surfaces. We finally obtain a Chal-H dataset with 50 flat shapes and a Chal-T dataset consisting of objects with inclined surfaces or tetrahedron-like shapes. All shapes are

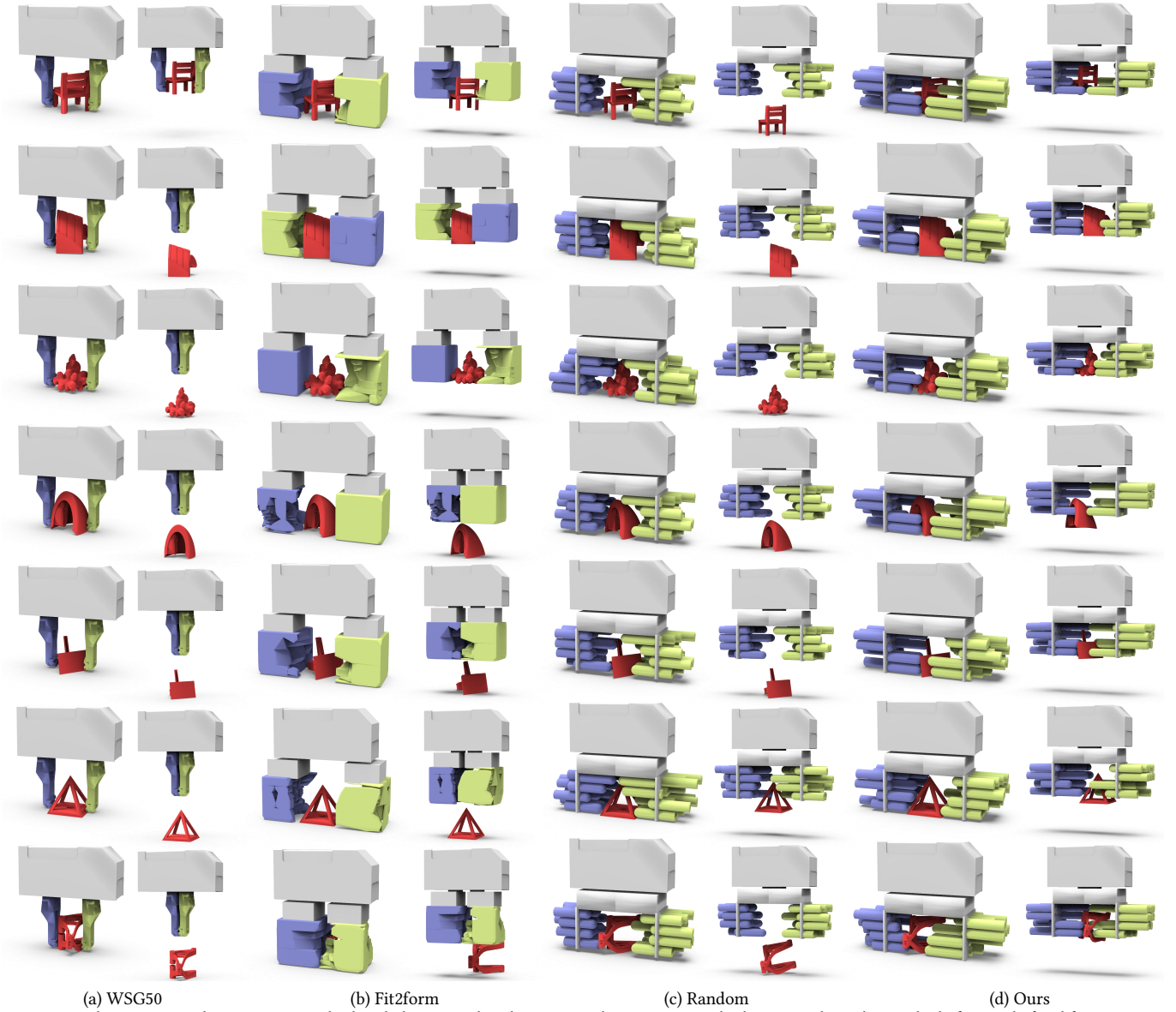


Fig. 8. Visual comparisons between our method and alternative baselines on six datasets. For each object, we show the results before and after lifting it. Note that our method is capable of generating successful grasps for both artificially generated CAD models and real-world scanned models.

Table 3. Comparison with the WSG50 gripper and the Passive Grasp method of directly extending all pins. Here, Chal-H represents the category of flat objects, while Chal-T represents objects with inclined surfaces or tetrahedron-like shapes.

Method	Train		Test		Chal-H				Chal-T			
	S (%) ↑	Q1 ↑	S (%) ↑	Q1 ↑	S (%) ↑	Q1 ↑	GwL (%)	GtL (%)	S (%) ↑	Q1 ↑	GwL (%)	GtL (%)
WSG50	39.24	0.2181	36.08	0.2226	26.00	0.2012	0.00	100.00	22.42	0.2217	0.00	100.00
Passive Grasp	58.97	0.3414	60.82	0.3560	36.00	0.3146	0.00	100.00	44.82	0.3280	0.00	100.00
Active Grasp (Ours)	82.96	0.3506	82.47	0.3456	72.00	0.3264	28.00	72.00	63.79	0.3526	75.87	24.13

unseen for the grasping policies during the training. More details of the challenge dataset can be found in Section B of supplemental material. We compare our learned policies with directly extending all pins to passively conform to the shape of the target object. The

quantitative comparisons are reported in Table 3, and the qualitative comparisons are illustrated in Figure 9.

Table 3 reveals that our learned policies exhibit superior adaptability in comparison to the conventional parallel jaw gripper (WSG50)

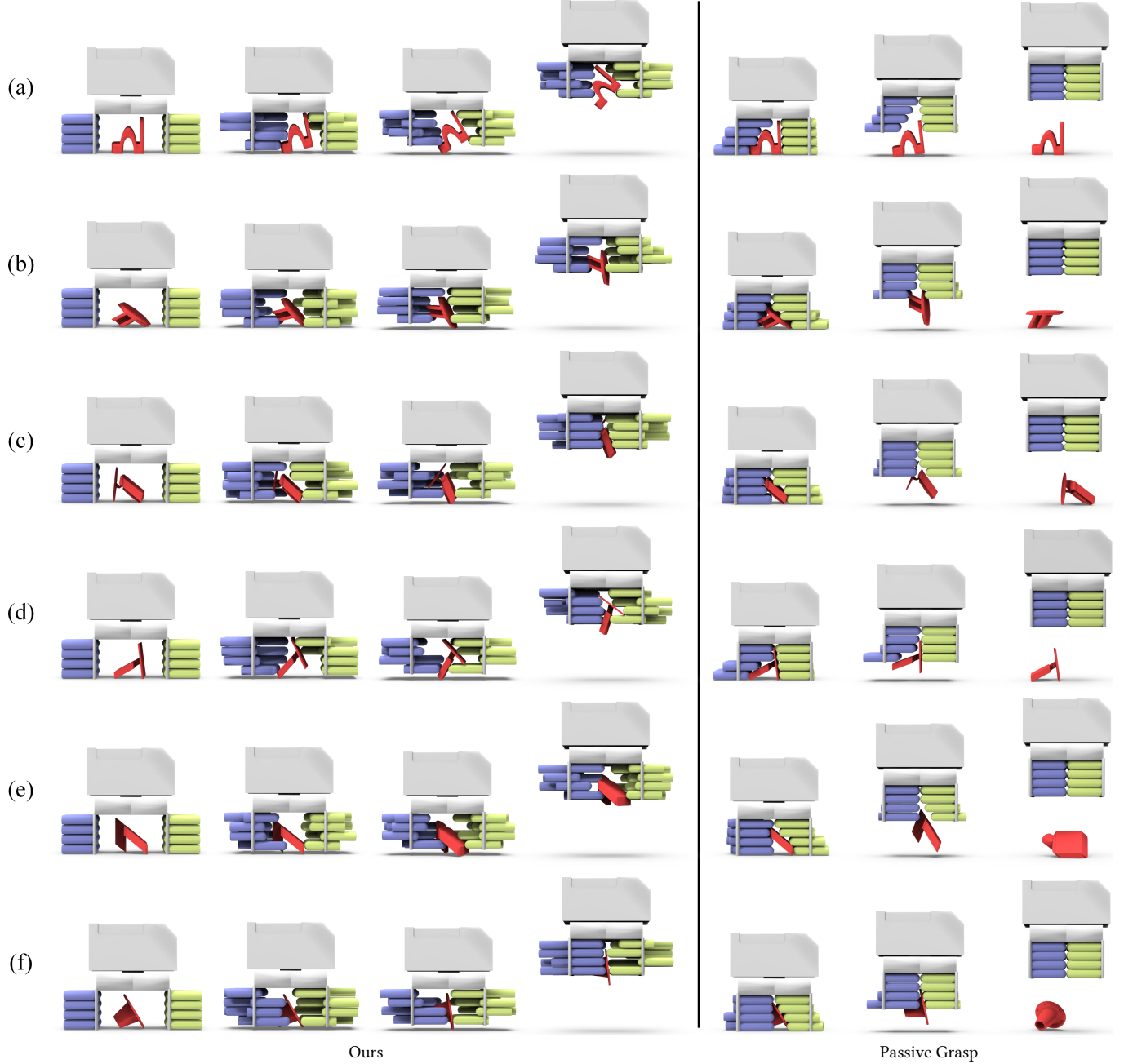


Fig. 9. Visualization of our approach (left) and the passive grasping method with all pins extended (right). We display the initial configurations of the gripper and several key frames of the grasping processes. For the Passive Grasp method, we showcase the moment of object and gripper detachment.

and passive grasping. Our gripper achieves 72.00% grasp success rate on flat objects and 63.79% on tetrahedron-like shapes, compared to 36.00% and 44.82% for Passive Grasp, respectively. WSG50 grippers also achieve low success rates on both datasets, i.e., 26.00% for Chal-H and 22.42% for Chal-T, demonstrating the necessity of our adaptive gripper design.

As illustrated in Figure 9, the passive grasping approach struggles due to the unidirectional extension of all pins, which often fails to establish effective force closure, causing the object to be pushed out

of the gripper and dropped. It can be observed from Figure 9 that our method is capable of grasping the target objects through online adjustments. Throughout the GwL phase, the object’s pose evolves through in-hand rotation, and the grasp configuration undergoes refinement for better force closure through dynamic adjustments. This also underscores the advantages of active grasping.

Additionally, we uncover an intriguing phenomenon by observing the proportion between GwL and GtL motion. For objects featuring flat shapes, the GtL skill is utilized more frequently (constituting

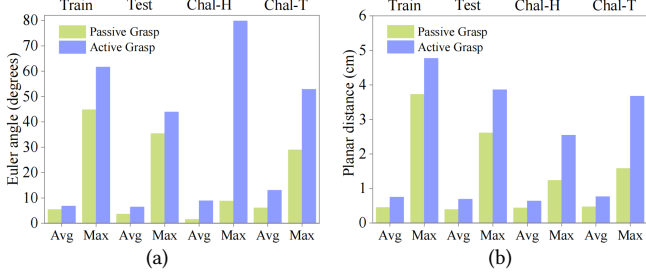


Fig. 10. Flexibility analysis on Active Grasp (Ours) and Passive Grasp methods in successfully grasped objects. (a) statistics the Euler angle variation (measured in degrees); (b) statistics positional variation along the xoy-plane (planar distance measured in centimeters). For each dataset, we statistic both the mean (Avg) and maximum (Max) values.

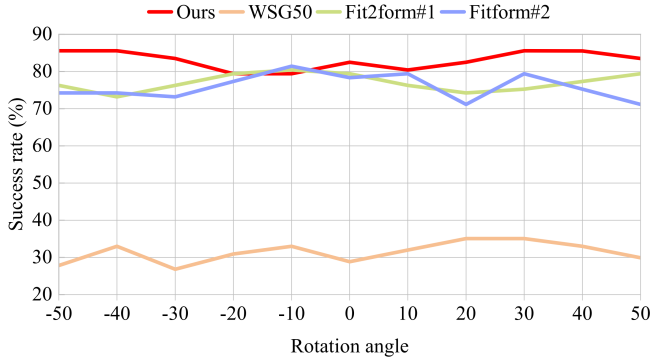


Fig. 11. Comparisons of the robustness to object rotations. The test objects undergo various rotations around the z-axis.

72.00% of cases), thereby facilitating the precise identification of optimal contact points. In contrast, for objects characterized by inclined surfaces or tetrahedron-like shapes, our algorithm tends to the utilization of GwL motion (accounting for 75.87% of instances) with in-air adjustments, to alter the relative pose between the object and the gripper, thereby ensuring more effective grasping. This highlights the capability of our approach in shape perception and adaptability.

5.4 Analysis of Pin-Pression Gripper Design

Gripper flexibility analysis. To illustrate the flexibility of our pin-pression gripper, we measure the variations of Euler angles and planar distances of the successfully grasped objects during the grasping procedure. We compare our method with the Passive Grasp method and we measure these pose changes before and after grasping. We calculate the Euclidean distance in the horizontal xoy plane and measure the planar shifts in centimeters. For the Euler angle, we convert the quaternions of the objects into rotation matrices, and then compute the angular difference between the two rotation matrices. As shown in Figure 10, our method exhibits richer pose variation than passive grasping, proving that our RL-based policy enhances grasping performance through in-hand adjustments.

Robustness to object rotation. To further demonstrate the robustness of our approach to object rotation, we compare its performance against the WSG50 gripper and Fit2form method, measuring

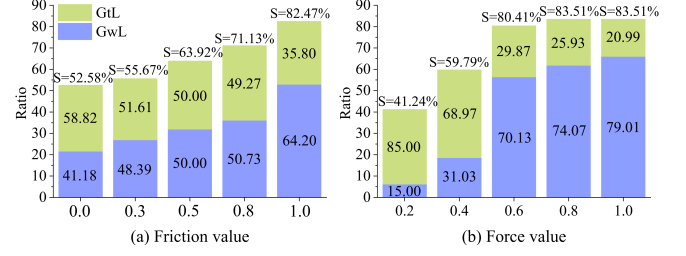


Fig. 12. The influences of gripper friction and grasp force on the performance of our method. We evaluate the grasp success rate and the percentages of both GtL and GwL motions, across different dynamic parameters.

success rate at different levels of object rotation. Specifically, all objects are rotated by $\pm 10^\circ$, $\pm 20^\circ$, $\pm 30^\circ$, $\pm 40^\circ$, and $\pm 50^\circ$ around the z-axis, respectively, and then the corresponding success rates are calculated. We make sufficient comparisons with the Fit2form method. For each object, we measure the grasping success rate by utilizing different gripper shapes specifically generated for the corresponding rotation (Fit2form#1). We also employ a consistent gripper shape for grasping an object with its various rotations (Fit2form#2). The success rates of these methods are reflected in Figure 11.

It can be observed that our adaptive pin-pression gripper is nearly always superior in handling object rotations, even if a significant object rotation is applied. The general-purpose WSG50 gripper cannot fit well to dynamic variations of object rotations due to the fixed finger shape. This results in consistently exhibiting the worst performance under perturbations of different rotations. We also note that the Fit2form method is sensitive to the rotation of objects, even for Fit2form#1 where the gripper shapes are specifically generated for the corresponding rotation. More visualization results about the robustness of our pin-pression gripper to arbitrary object poses can be found in Section D of supplemental material.

Friction and grasp force analysis. We also analyze the impacts of gripper friction and grasp force on the performance of our learned policies, aiming to unveil interesting aspects of grasping behavior. We evaluate our method on the grasp success rate and the percentages of GtL and GwL motions. In the friction analysis, we vary the gripper friction from 0.0 to 1.0 and retrain the policy network accordingly. The results with different gripper friction parameters are illustrated in Figure 12a. As anticipated, the success rate consistently improves as we increase the friction parameter, suggesting that gripper friction plays a crucial role in achieving a successful grasp. Moreover, an interesting observation in grasping mode is noteworthy: with an increase in friction, there is a notable rise in the percentage of GwL motion. This suggests that our method retains the capability to flexibly adopt appropriate grasping modes, coordinating pin motion with the target object to successfully grasp it. For the grasp force analysis, we evaluate a total of five grasp forces. The related results of our method under different grasp forces are shown in Figure 12b. It can be seen that grasp force is also a pivotal factor in the performance of our method for adaptively manipulating the gripper toward a successful grasp. We also evaluate the robustness

Table 4. Performance of our pin-pression gripper with various pin densities.

Pin density	$S(\%) \uparrow$	$Q1 \uparrow$	$T(s) \downarrow$
3×3	50.51	0.3416	594.24
4×4	82.47	0.3456	634.38
5×5	88.66	0.3458	688.67

Table 5. Sim-to-Sim transfer experiment. We evaluate the robustness of our learning-based approach by transferring it to a different physics simulator Isaac Gym [Makoviychuk et al. 2021].

Method	$S(\%) \uparrow$	Time cost (s) \downarrow	
		Computation	Simulation
WSG50	47.42%	0.00	765.81
Fit2form	70.10%	477.56	806.95
Ours	73.19%	0.79	784.47

of our method to variations in object mass and friction. These experiments are conducted to assess the success rate of our method on the testing data (see Figure 5 in supplemental material).

Exploration of pin density. We explore pin arrays of different densities, including 3×3 , 4×4 , and 5×5 . We maintain the overall area of the gripper and only redistribute the pins for the different densities. We retrain the 3×3 and 5×5 gripper configurations and report the grasp success rate and the sum running time cost in Table 4. In our pursuit to design a pin-pression gripper that performs well across a wide variety of shapes while minimizing the number of pin actuators, we found that a 4×4 grid size is sufficiently flexible to achieve a general grasping effect, with an average success rate over 82.00% and significantly outperform other density alternatives.

Sim-to-Sim transfer. Transferring grasping capabilities to simulation environments with different physics is of utmost importance for validating the generality of our approach, encompassing both the hardware design of our pin-pression gripper and our learning-based grasping algorithm. Different simulators may exhibit variations in physical dynamics, such as subtle differences in collision detection, object modeling, and numerical integration methods. To show the robustness of our method, we compare it with other alternative methods by directly transferring them to a different physics simulator. Specifically, we replicate the gripper grasping environment from PyBullet [Coumans and Bai 2016], employing the open-source Bullet Physics engine, to a new test scene in Isaac Gym [Makoviychuk et al. 2021] with the PhysX engine [NVIDIA 2020]. We evaluate both the grasp success rate and the sum running time cost for all test instances in Isaac Gym, and we report the results in Table 5. Our geometry-aware approach sustains considerable grasping performance in the new simulation environment, surpassing other methods in terms of success rate. Furthermore, our method requires less computational and simulation time compared to the Fit2form method, which is known for its time-consuming nature for customized gripper shape generation.

Sim-to-Real transfer. We further validate the performance of our approach on a real robotic platform. Transferring grasping capabilities to real-world environments presents significantly greater challenges for both gripper manufacturing and grasping policy learning compared to simulations. On the one hand, we must ensure that the physically manufactured gripper aligns with our design, particularly for the automatic and accurate control of the pin’s extension and retraction. On the other hand, we need to ensure that the grasping policy trained in simulation can be effectively transferred to the real world.

To achieve this, we use an electric actuator as the pin of the gripper finger, which allows us to automatically and accurately control and read the pin’s movement distance (*pin reading*). We then physically fabricate a pin-pression gripper featuring 4×4 pin arrays, ensuring it is proportionally consistent with our gripper design and exact in size. This gripper is also equipped with an in-hand RGB-D camera for capturing the object shape. To adapt our grasping policy to the real-world environment, we employ a two-stage teacher-student training paradigm [Chen et al. 2022]. First, we train a teacher policy network using reinforcement learning with full state information captured in simulation. Subsequently, we train a student policy network via DAGGER algorithm [Ross et al. 2011b], which takes virtual RGB-D observations and pin readings as inputs and is optimized to match the actions predicted by the teacher policy network. Lastly, we use real-world observations, including pin readings from each actuator and RGB-D images from the camera, as inputs to the student policy network, thereby transferring the grasping capability to the real world.

The physical experiment of sim-to-real transfer is illustrated in Figure 13. We showcase several key steps of dynamically adjusting the extension and retraction of pins throughout the grasping process, applied to five object models. As illustrated, target objects are successfully grasped and lifted by the dynamic grasping skill of our physically manufactured pin-pression gripper. This confirms the effectiveness and practicality of our approach in real-world applications. For more details on the experiment implementation, hardware specifications, and the results of the physical experiment, please refer to Section E in supplemental material.

6 DISCUSSION AND FUTURE WORK

We concurrently investigate gripper design and dynamic grasping skills for 3D objects featuring diverse geometry and poses. We have developed an innovative pin-pression gripper capable of adaptively adjusting its shape to grasp objects with various geometries and topologies and under arbitrary poses. We also propose an RL-based method to learn adaptive grasping policies along with a curriculum learning technique to facilitate online grasping using different grasping modes. Through extensive quantitative and qualitative experiments on diverse novel object datasets, our method demonstrates the capability to handle complex shapes and achieve dynamic robot grasping with a high success rate and efficiency. We also physically manufacture the pin-pression gripper based on the design and highlight sim-to-real results on a real robot platform, confirming that our method can be effectively transferred to real-world applications.

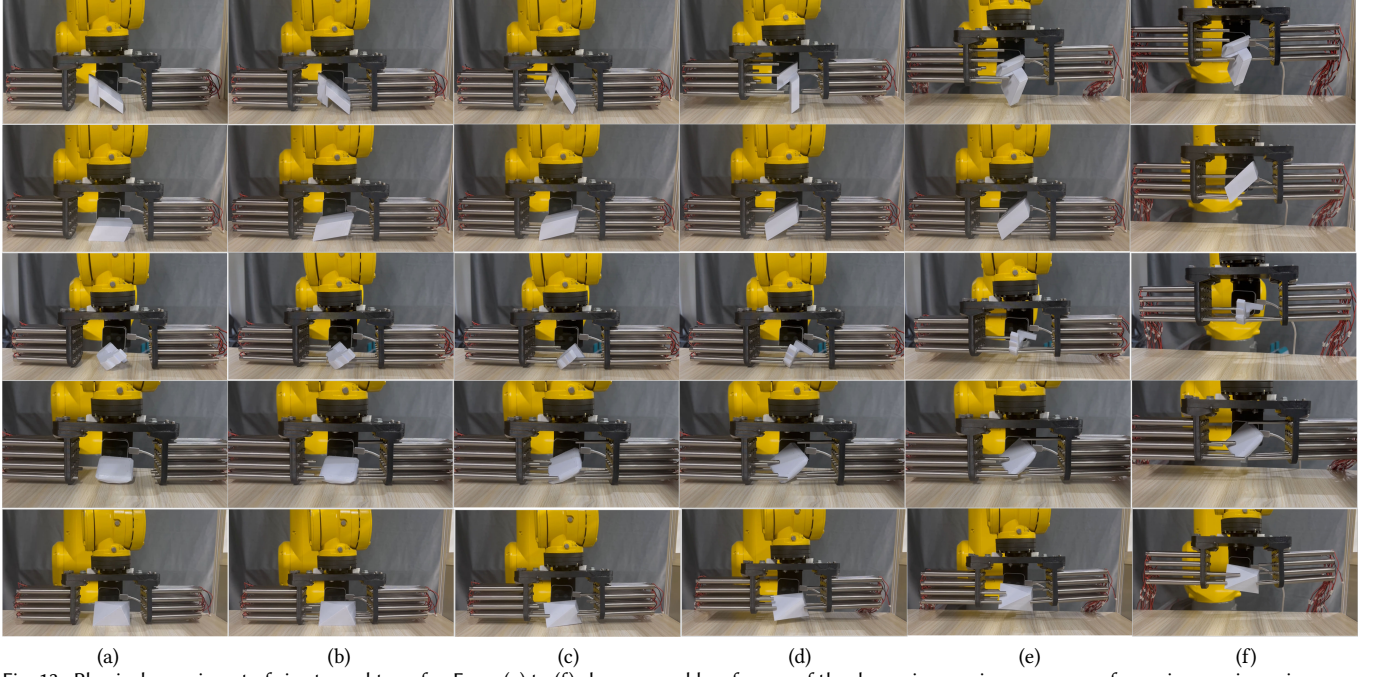


Fig. 13. Physical experiment of sim-to-real transfer. From (a) to (f) show several key frames of the dynamic grasping processes of our pin-pressure gripper on five object models. The gripper approaches the target object from the top, forms a basic closure against the object with pin movements, and lifts the object while further performing in-hand re-orientation to improve the grasping stability.

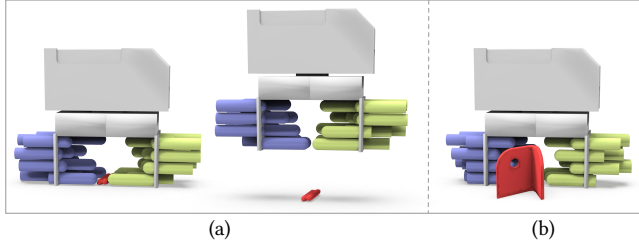


Fig. 14. Failure cases. Our gripper is not adept at handling certain extremely thin shapes (a), due to the cylindrical shape formulation of the pins and the limited pin grain. The same reasons may also lead to objects with curved surfaces (b) being pushed out.

As an initial endeavor to develop a novel pin-pressure gripper to tackle the significant challenges of adaptive grasping, our work still has certain limitations. Firstly, to achieve adaptive grasping, the pins on our gripper necessitate a broader range of adjustments, leading to the scales of target objects being limited. Secondly, our gripper may encounter difficulties with extreme cases, such as thin objects like chopsticks (refer to Figure 14a), owing to the constrained space available for the gripper to perform the adjustments. We also observe that certain objects with curved surfaces might be pushed out as a result of the force generated along the x-axis by the interaction of pins with their surfaces (see Figure 14b).

Additionally, our grasping process is a multi-objective optimization that simultaneously considers task success, grasp stability, and efficiency. Since we prioritize success and stability, the efficiency reward is given a lower weight, allowing some redundant actions. Further improving the execution efficiency of the pin-pressure gripper is an important area for future work. In parallel, we are engaged

in applying our gripper to real-world applications in daily life scenarios [Li et al. 2024] and industrial automation [Zhao et al. 2023].

ACKNOWLEDGMENTS

This work was supported in parts by NSFC (62325211, 62495081, 62272082, 12494554, 62132021), the Joint Fund General Project of Liaoning Provincial Department of Science and Technology (2024-MSLH-352), and the Major Program of Xiangjiang Laboratory (No. 23XJ01009).

REFERENCES

- John R Amend, Eric Brown, Nicholas Rodenberg, Heinrich M Jaeger, and Hod Lipson. 2012. A positive pressure universal gripper based on the jamming of granular material. *IEEE Transactions on Robotics* (2012).
- OpenAI: Marcin Andrychowicz, Bowen Baker, Maciej Chociej, Rafal Jozefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, et al. 2020. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research* (2020).
- Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. 2017. Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine* (2017).
- Lucian Balan and Gary M. Bone. 2003. Automated Gripper Jaw Design and Grasp Planning for Sets of 3D Objects. *Journal of Field Robotics* (2003).
- Richard Bellman. 1957. A Markovian decision process. *Journal of Mathematics and Mechanics* (1957).
- Antonio Bicchi. 1995. On the closure properties of robotic grasping. *The International Journal of Robotics Research* (1995).
- Cristian Bodnar, Adrian Li, Karol Hausman, Peter Pastor, and Mrinal Kalakrishnan. 2019. Quantile qt-opt for risk-aware vision-based robotic grasping. *arXiv preprint arXiv:1910.02787* (2019).
- R.C. Brost and R.R. Peters. 1996. Automatic design of 3-d fixtures and assembly pallets. In *IEEE International Conference on Robotics and Automation*.
- Russell G Brown and Randy C Brost. 1999. A 3-D modular gripper design tool. *IEEE Transactions on Robotics and Automation* (1999).
- Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. 2015. Shapenet:

- An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012* (2015).
- R. Qi Charles, Hao Su, Mo Kaichun, and Leonidas J. Guibas. 2017. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*.
- Shuaijun Chen, Jinxi Wang, Wei Pan, Shang Gao, Meili Wang, and Xuequan Lu. 2023. Towards uniform point distribution in feature-preserving point cloud filtering. *Computational Visual Media* (2023).
- Tao Chen, Megha H. Tippur, Siyang Wu, Vikash Kumar, Edward H. Adelson, and Pulkit Agrawal. 2022. Visual dexterity: In-hand reorientation of novel and complex object shapes. *Science Robotics* (2022).
- Sammy Christen, Shreyas Hampali, Fadime Sener, Edoardo Remelli, Tomas Hodan, Eric Sauser, Shugao Ma, and Bugra Tekin. 2024. Diffh2o: Diffusion-based synthesis of hand-gesture interactions from textual descriptions. In *SIGGRAPH Asia 2024*.
- Fu-Jen Chu, Ruinian Xu, and Patricio A Vela. 2018. Real-world multiobject, multigrasp detection. *IEEE Robotics and Automation Letters* (2018).
- Erwin Coumans and Yunfei Bai. 2016. Pybullet, a python module for physics simulation for games, robotics and machine learning. URL <http://pybullet.org> (2016).
- Yuhong Deng, Xiaofeng Guo, Yixuan Wei, Kai Lu, Bin Fang, Di Guo, Huaping Liu, and Fuchun Sun. 2019. Deep reinforcement learning for robotic pushing and picking in cluttered environment. In *IEEE International Conference on Intelligent Robots and Systems*.
- Nils Dengler, David Großklaus, and Maren Bennewitz. 2022. Learning Goal-Oriented Non-Prehensile Pushing in Cluttered Scenes. In *IEEE International Conference on Intelligent Robots and Systems*.
- Amaury Depierre, Emmanuel Dellandréa, and Liming Chen. 2018. Jacquard: A Large Scale Dataset for Robotic Grasp Detection. In *IEEE International Conference on Intelligent Robots and Systems*.
- Aaron M Dollar and Robert D Howe. 2010. The highly adaptive SDM hand: Design and performance evaluation. *The International Journal of Robotics Research* (2010).
- Guoguang Du, Kai Wang, Shiguo Lian, and Kaiyong Zhao. 2021. Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: a review. *Artificial Intelligence Review* (2021).
- Adrien Ecoffet, Joost Huizinga, Joel Lehman, Kenneth O Stanley, and Jeff Clune. 2019. Go-explode: a new approach for hard-exploration problems. *arXiv preprint arXiv:1901.10995* (2019).
- Hao-Shu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. 2020. Graspnet-1billion: A large-scale benchmark for general object grasping. In *IEEE Conference on Computer Vision and Pattern Recognition*.
- Nima Fazeli, Miquel Oller, Jiajun Wu, Zheng Wu, Joshua B Tenenbaum, and Alberto Rodriguez. 2019. See, feel, act: Hierarchical learning for complex manipulation skills with multisensory fusion. *Science Robotics* (2019).
- Hong Fu, Haokun Yang, Weishu Song, and Wenzeng Zhang. 2017. A novel cluster-tube self-adaptive robot hand. *Robotics and Biomimetics* (2017).
- Hong Fu and Wenzeng Zhang. 2019. The development of a soft robot hand with pin-array structure. *Applied Sciences* (2019).
- Moritz Geilinger, Roi Poranne, Ruta Desai, Bernhard Thomaszewski, and Stelian Coros. 2018. Skaterbots: Optimization-based design and motion synthesis for robotic creatures with legs and wheels. *ACM Transactions on Graphics* (2018).
- Huy Ha, Shubham Agrawal, and Shuran Song. 2020. Fit2Form: 3D Generative Model for Robot Gripper Form Design. In *Conference on Robotic Learning*.
- Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, G. Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, P. Abbeel, and Sergey Levine. 2018. Soft Actor-Critic Algorithms and Applications. *arXiv preprint arXiv:1812.05905* (2018).
- M. Honarpardaz, M. Tarkian, J. Ölvander, and X. Feng. 2017. Finger design automation for industrial robot grippers: A review. *Robotics and Autonomous Systems* (2017).
- Muhammad Babar Imtiaz, Yuansong Qiao, and Brian Lee. 2023. Prehensile and non-prehensile robotic pick-and-place of objects in clutter using deep reinforcement learning. *Sensors* (2023).
- Yan-Bin Jia. 2004. Computation on Parametric Curves with an Application in Grasping. *The International Journal of Robotics Research* (2004).
- Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. 2018. Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation. *arXiv preprint arXiv:1806.10293* (2018).
- Daniel Kappler, Jeannette Bohg, and Stefan Schaal. 2015. Leveraging big data for grasp planning. In *IEEE International Conference on Robotics and Automation*.
- Marios Kiatos and Sotiris Malassiotis. 2019. Grasping unknown objects by exploiting complementarity with robot hand geometry. In *Computer Vision Systems*.
- Milin Kodnongbua, Ian Good, Yu Lou, Jeffrey Lipton, and Adriana Schulz. 2022. Computational design of passive grippers. *ACM Transactions on Graphics* (2022).
- Thomas R Kurfess et al. 2005. *Robotics and automation handbook*. CRC press Boca Raton, FL.
- Michelle A Lee, Yuke Zhu, Peter Zachares, Matthew Tan, Krishnan Srinivasan, Silvio Savarese, Li Fei-Fei, Animesh Garg, and Jeannette Bohg. 2020. Making sense of vision and touch: Learning multimodal representations for contact-rich tasks. *IEEE Transactions on Robotics* (2020).
- Ian Lenz, Honglak Lee, and Ashutosh Saxena. 2015. Deep learning for detecting robotic grasps. *The International Journal of Robotics Research* (2015).
- Sergey Levine, Peter Pastor, Alex Krizhevsky, and Deirdre Quillen. 2016. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The International Journal of Robotics Research* (2016).
- Wenhao Li, Zhiyuan Yu, Qijin She, Zhinan Yu, Yuqing Lan, Chenyang Zhu, Ruizhen Hu, and Kai Xu. 2024. LLM-enhanced Scene Graph Learning for Household Rearrangement. In *SIGGRAPH Asia 2024*.
- Min Liu, Zherong Pan, Kai Xu, Kanishka Ganguly, and Dinesh Manocha. 2020. Deep Differentiable Grasp Planner for High-DOF Grippers. (2020).
- Jeffrey Mahler, Jacky Liang, Sherdil Niyaz, Michael Laskey, Richard Doan, Xinyu Liu, Juan Aparicio Ojea, and Ken Goldberg. 2017. Dex-Net 2.0: Deep Learning to Plan Robust Grasps with Synthetic Point Clouds and Analytic Grasp Metrics. (2017).
- Jeffrey Mahler, Florian T. Pokorny, Brian Hou, Melrose Roderick, Michael Laskey, Mathieu Aubry, Kai J. Kohlhoff, Torsten Kröger, James J. Kuffner, and Ken Goldberg. 2016. Dex-Net 1.0: A cloud-based network of 3D objects for robust grasp planning using a Multi-Armed Bandit model with correlated rewards. *IEEE International Conference on Robotics and Automation* (2016).
- Viktor Makovychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, N. Rudin, Arthur Allshire, Ankur Handa, and Gavriel State. 2021. Isaac Gym: High Performance GPU-Based Physics Simulation For Robot Learning. (2021).
- Alexis Maldonado, Ulrich Klank, and Michael Beetz. 2010. Robotic grasping of unmodeled objects using time-of-flight range data and finger torque information. In *IEEE International Conference on Intelligent Robots and Systems*.
- Christian Mandery, Ömer Terlemez, Martin Do, Nikolaus Vahrenkamp, and Tamim Asfour. 2016. Unifying Representations and Large-Scale Whole-Body Motion Databases for Studying Human Motion. *IEEE Transactions on Robotics* (2016).
- Takahiro Miki, Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. 2022. Learning robust perceptive locomotion for quadrupedal robots in the wild. *Science Robotics* (2022).
- Andrew T Miller and Peter K Allen. 2004. Graspit! a versatile simulator for robotic grasping. *IEEE Robotics & Automation Magazine* (2004).
- An Mo and Wenzeng Zhang. 2017. Pin array hand: a universal robot gripper with pins of ellipse contour. In *Robotics and Biomimetics*.
- An Mo and Wenzeng Zhang. 2019. A universal robot gripper based on concentric arrays of rotating pins. *Science China Information Sciences* (2019).
- Douglas Morrison, Peter Corke, and Jürgen Leitner. 2018. Closing the loop for robotic grasping: a real-time, generative grasp synthesis approach. In *Robotics: Science and Systems*.
- Arsalan Mousavian, Clemens Eppner, and Dieter Fox. 2019. 6-dof graspnet: Variational grasp generation for object manipulation. In *IEEE International Conference on Computer Vision*.
- Jungwook Mun, Khang Truong Giang, Yunghee Lee, Nayoung Oh, Sejoon Huh, Min Kim, and Sungho Jo. 2023. HybGrasp: A Hybrid Learning-to-Adapt Architecture for Efficient Robot Grasping. *IEEE Robotics and Automation Letters* (2023).
- Sanmit Narvekar, Bei Peng, Matteo Leonetti, Jivko Sinapov, Matthew E Taylor, and Peter Stone. 2020. Curriculum learning for reinforcement learning domains: A framework and survey. *Journal of Machine Learning Research* (2020).
- NVIDIA. 2020. *Nvidia PhysX*. <https://developer.nvidia.com/physx-sdk>
- Zherong Pan, Duo Zhang, Changhe Tu, and Xifeng Gao. 2021. Planning of power grasps using infinite program under complementary constraints. *IEEE Robotics and Automation Letters* (2021).
- Lerrel Pinto and Abhinav Gupta. 2016. Supersizing self-supervision: Learning to grasp from 50K tries and 700 robot hours. In *IEEE International Conference on Robotics and Automation*.
- Deirdre Quillen, Eric Jang, Ofir Nachum, Chelsea Finn, Julian Ibarz, and Sergey Levine. 2018. Deep Reinforcement Learning for Vision-Based Robotic Grasping: A Simulated Comparative Evaluation of Off-Policy Methods. In *IEEE International Conference on Robotics and Automation*.
- Aravind Rajeswaran, Vikash Kumar, Abhishek Gupta, Giulia Vezzani, John Schulman, Emanuel Todorov, and Sergey Levine. 2017. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. *arXiv preprint arXiv:1709.10087* (2017).
- Joseph Redmon and Anelia Angelova. 2015. Real-time grasp detection using convolutional neural networks. In *IEEE International Conference on Robotics and Automation*.
- Empire Robotics. 2019. VERSABALL GRIPPER. <https://www.empirerobotics.com> (2019).
- Robotiq. 2022. *Hand-E Adaptive Gripper*. https://assets.robotiq.com/website-assets/support_documents/document/Hand-E_Manual_Generic_PDF_20220111.pdf
- Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. 2011a. A reduction of imitation learning and structured prediction to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics*.
- Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. 2011b. A reduction of imitation learning and structured prediction to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics*.

- Andreas Schroeffer, Christoph Rehekampff, and Tim C Lueth. 2019. An Automated Design Approach for Task-Specific two Finger Grippers for Industrial Applications. In *Robotics and Biomimetics*.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).
- SCHUNK. 2023. *WSG Universal gripper*. https://schunk.com/us/en/gripping-systems/parallel-gripper/wsg/c/PGR_820
- Peter B Scott. 1985. The ‘Omnigripper’: A form of robot universal gripper. *Robotica* (1985).
- Lin Shao, Fabio Ferreira, Mikael Jorda, Varun Nambiar, Jianlan Luo, Eugen Solowjow, Juan Aparicio Ojea, Oussama Khatib, and Jeannette Bohg. 2020. Unigrasp: Learning a unified model to grasp with multifingered robotic hands. *IEEE Robotics and Automation Letters* (2020).
- Quanquan Shao, Jie Hu, Weiming Wang, Yi Fang, Wenhui Liu, Jin Qi, and Jin Ma. 2019. Suction grasp region prediction using self-supervised learning for object picking in dense clutter. In *IEEE International Conference on Mechatronics System and Robots*.
- Purvash Sharma and Damian Valles. 2020. Deep convolutional neural network design approach for 3D object detection for robotic grasping. In *Annual Computing and Communication Workshop and Conference*.
- Qijin She, Ruizhen Hu, Juzhan Xu, Min Liu, Kai Xu, and Hui Huang. 2022. Learning high-DOF reaching-and-grasping via dynamic representation of gripper-object interaction. *ACM Transactions on Graphics* (2022).
- Jun Shintake, Vito Cacucciolo, Dario Floreano, and Herbert Shea. 2018. Soft robotic grippers. *Advanced Materials* (2018).
- Arjun Singh, James Sha, Karthik S Narayan, Tudor Achim, and Pieter Abbeel. 2014. Big-bird: A large-scale 3d database of object instances. In *IEEE International Conference on Robotics and Automation*.
- Yanan Song, Liang Gao, Xinyu Li, and Weiming Shen. 2020. A novel robotic grasp detection method based on region proposal networks. *Robotics and Computer-Integrated Manufacturing* (2020).
- Zejia Su, Haibin Huang, Chongyang Ma, Hui Huang, and Ruizhen Hu. 2023. Point cloud completion via structured feature maps using a feedback network. *Computational Visual Media* (2023).
- Weikang Wan, Haoran Geng, Yun Liu, Zikang Shan, Yaodong Yang, Li Yi, and He Wang. 2023. UniDexGrasp++: Improving Dexterous Grasping Policy Learning via Geometry-aware Curriculum and Iterative Generalist-Specialist Learning. *IEEE International Conference on Computer Vision* (2023).
- Lirui Wang, Yu Xiang, Wei Yang, Arsalan Mousavian, and Dieter Fox. 2022. Goal-auxiliary actor-critic for 6d robotic grasping with point clouds. In *Conference on Robot Learning*.
- Weijia Wang, Xiao Liu, Hailing Zhou, Lei Wei, Zhigang Deng, Manzur Murshed, and Xuequan Lu. 2024a. Noise4Denoise: Leveraging noise for unsupervised point cloud denoising. *Computational Visual Media* (2024).
- Yongliang Wang, Kamal Mokhtar, Cock Heemskerk, and Hamidreza Kasaei. 2024b. Self-supervised learning for joint pushing and grasping policies in highly cluttered environments. In *IEEE International Conference on Robotics and Automation*.
- Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. 2018. Dynamic Graph CNN for Learning on Point Clouds. *ACM Transactions on Graphics* (2018).
- Walter Wohlking, Aitor Aldoma, Radu B. Rusu, and Markus Vincze. 2012. 3DNet: Large-scale object class recognition from CAD models. In *IEEE International Conference on Robotics and Automation*.
- Yinzen Xu, Weikang Wan, Jialiang Zhang, Haoran Liu, Zikang Shan, Hao Shen, Ruicheng Wang, Haoran Geng, Yijia Weng, Jiayi Chen, et al. 2023. Unidexgrasp: Universal robotic dexterous grasping via learning diverse proposal generation and goal-conditioned policy. In *IEEE Conference on Computer Vision and Pattern Recognition*.
- Zhenjia Xu, Beichun Qi, Shubham Agrawal, and Shuran Song. 2021. Adagrasp: Learning an adaptive gripper-aware grasping policy. In *IEEE International Conference on Robotics and Automation*.
- Zhengrong Xue, Han Zhang, Jingwen Cheng, Zhengmao He, Yuanchen Ju, Changyi Lin, Gu Zhang, and Huazhe Xu. 2024. Arraybot: Reinforcement learning for generalizable distributed manipulation through touch. In *IEEE International Conference on Robotics and Automation*.
- Andy Zeng, Shuran Song, Stefan Welker, Johnny Lee, Alberto Rodriguez, and Thomas Funkhouser. 2018. Learning synergies between pushing and grasping with self-supervised deep reinforcement learning. In *IEEE International Conference on Intelligent Robots and Systems*.
- Pengyu Zhang, Dong Wang, and Huchuan Lu. 2024. Multi-modal visual tracking: Review and experimental comparison. *Computational Visual Media* (2024).
- Yunbo Zhang, Alexander Clegg, Sehoon Ha, Greg Turk, and Yuting Ye. 2023. Learning to Transfer In-Hand Manipulations Using a Greedy Shape Curriculum. In *Computer Graphics Forum*.
- Allan Zhao, Jie Xu, Mina Konaković-Luković, Josephine Hughes, Andrew Spielberg, Daniela Rus, and Wojciech Matusik. 2020. Robogrammar: graph grammar for terrain-optimized robot design. *ACM Transactions on Graphics* (2020).
- Hang Zhao, Zherong Pan, Yang Yu, and Kai Xu. 2023. Learning physically realizable skills for online packing of general 3D shapes. *ACM Transactions on Graphics* (2023).
- Hang Zhao, Juzhan Xu, Kexiong Yu, Ruizhen Hu, Chenyang Zhu, and Kai Xu. 2025. Deliberate Planning of 3D Bin Packing on Packing Configuration Trees. *arXiv preprint arXiv:2504.04421* (2025).
- Qingnan Zhou and Alec Jacobson. 2016. Thingi10k: A dataset of 10,000 3d-printing models. *arXiv preprint arXiv:1605.04797* (2016).
- Wenyang Zhou, Lu Yuan, and Taijiang Mu. 2024. Multi3D: 3D-aware multimodal image synthesis. *Computational Visual Media* (2024).
- Xiangyang Zhu and Jun Wang. 2003. Synthesis of force-closure grasps on 3-D objects based on the Q distance. *IEEE Transactions on Robotics and Automation* (2003).
- Yuke Zhu, Ziyu Wang, Josh Merel, Andrei Rusu, Tom Erez, Serkan Cabi, Saran Tunyasuvunakool, János Kramár, Raia Hadsell, Nando de Freitas, et al. 2018. Reinforcement and imitation learning for diverse visuomotor skills. *arXiv preprint arXiv:1802.09564* (2018).
- Berk Çalli, Arjun Singh, Aaron Walsman, Siddhartha S. Srinivasa, P. Abbeel, and Aaron M. Dollar. 2015. The YCB object and Model set: Towards common benchmarks for manipulation research. *IEEE International Conference on Advanced Robotics* (2015).