

RoTri-Diff: A Spatial Robot–Object Triadic Interaction-Guided Diffusion Model for Bimanual Manipulation

Zixuan Chen^{1,3}, Nga Teng Chan^{4,3}, Yiwen Hou³, Chenrui Tie³, Zixuan Liu³, Haonan Chen³, Junting Chen³, Jieqi Shi², Yang Gao², Jing Huo^{1†}, Lin Shao^{3†}

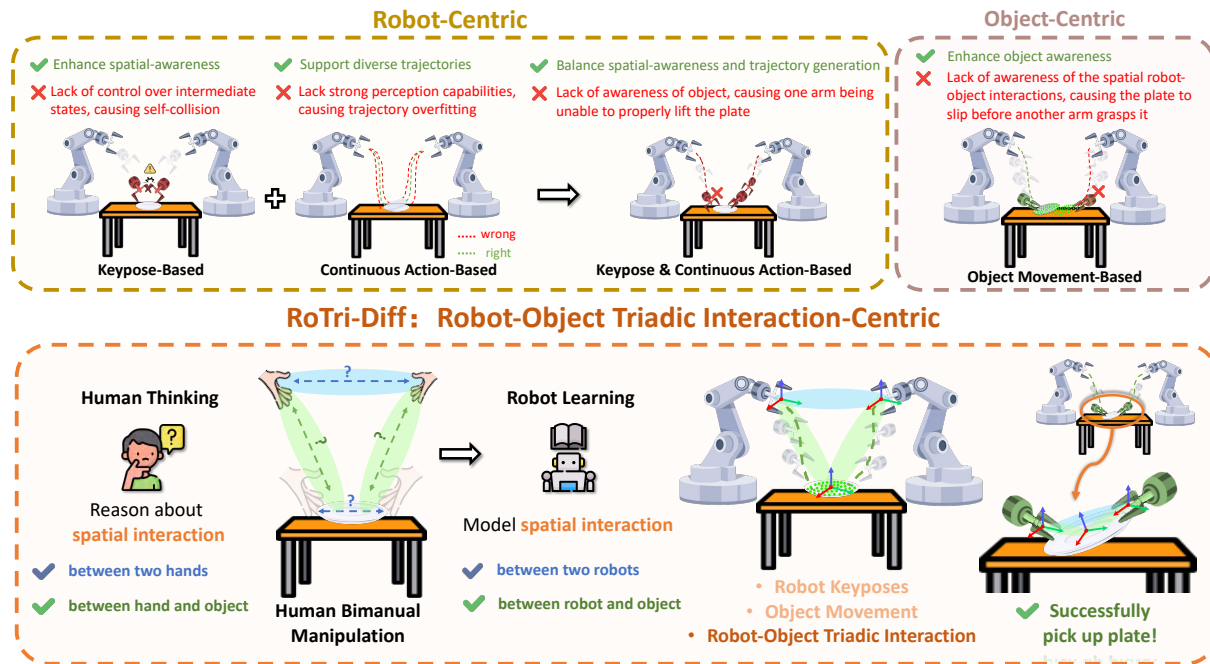


Fig. 1: We present **RoTri-Diff**, a diffusion-based framework for bimanual imitation learning that centers on robot–object triadic interaction (*RoTri*). By explicitly modeling and leveraging the relative 6D pose relations between the two arm end-effectors and the manipulated objects, it achieves stable performance on bimanual tasks requiring fine-grained coordination.

Abstract—Bimanual manipulation is a fundamental robotic skill that requires continuous and precise coordination between two arms. While imitation learning (IL) is the dominant paradigm for acquiring this capability, existing approaches, whether robot-centric or object-centric, often overlook the

dynamic geometric relationship among the two arms and the manipulated object. This limitation frequently leads to inter-arm collisions, unstable grasps, and degraded performance in complex tasks. To address this, in this paper we explicitly model the Robot–Object Triadic Interaction (*RoTri*) representation in bimanual systems, by encoding the relative 6D poses between the two arms and the object to capture their spatial triadic relationship and establish continuous triangular geometric constraints. Building on this, we further introduce **RoTri-Diff**, a diffusion-based imitation learning framework that combines *RoTri* constraints with robot keyposes and object motion in a hierarchical diffusion process. This enables the generation of stable, coordinated trajectories and robust execution across different modes of bimanual manipulation. Extensive experiments show that our approach outperforms state-of-the-art baselines by 10.2% on 11 representative RL-Bench2 tasks and achieves stable performance on 4 challenging real-world bimanual tasks. Project website: <https://rotri-diff.github.io/>.

†Corresponding Author

This work was completed during Zixuan and Nga Teng’s visiting at the National University of Singapore.

¹Zixuan Chen and Jing Huo are with the School of Computer Science, Nanjing University, China. Emails: chenzx@nju.edu.cn, huojing@nju.edu.cn

²Jieqi Shi and Yang Gao are with the School of Intelligence Science and Technology, Nanjing University, China. Emails: isjieqi@nju.edu.cn, gaoy@nju.edu.cn

³Yiwen Hou, Chenrui Tie, Zixuan Liu, Haonan Chen, Junting Chen and Lin Shao are with the School of Computing, National University of Singapore, Singapore. Email: {yiwenhou, chenrui.tie, zixuanliu}@u.nus.edu, {chenhaonan, junting.chen, linshao}@u.nus.edu

⁴Nga Teng Chan is with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, China. Email: ntchanab@connect.ust.hk

This work is supported in part by New Generation Artificial Intelligence-National Science and Technology Major Project (2025ZD0122904), National Natural Science Foundation of China (62192783, 62276128, 62506153), Jiangsu Science and Technology Major Project (BG2025035), the Fundamental Research Funds for the Central Universities(KG202514) and the Collaborative Innovation Center of Novel Software Technology and Industrialization.

I. INTRODUCTION

Bimanual manipulation is a fundamental robotic capability, essential for executing complex, human-like tasks that demand fine-grained dual-arm coordination [1]–[5]. Imitation learning (IL) has become the dominant paradigm for acquiring this skill [4], [6]–[13], and existing approaches fall into two main categories. The first is robot-centric IL, where mod-

els learn directly from demonstration trajectories: they either predict sparse robot keyposes for high-level planning [10], [14], executed by motion planners [15], [16], or generate dense continuous action sequences for direct control [17], [18]. Hybrid methods combine these paradigms [19], [20], using keyposes to guide continuous action generation. More recently, object-centric IL methods [9], [21] enrich robot-centric IL by incorporating the manipulated object’s motion as auxiliary guidance.

Despite significant progress, existing approaches still face critical limitations. As shown in the upper part of Fig. 1, current paradigms struggle in the “pick plate” task that requiring fine-grained dual-arm coordination (one arm tilting the plate while the other grasps it). Keypose-based methods enhance spatial awareness but, due to sparse supervision and poor control of intermediate states, often yield inaccurate trajectories and inter-arm collisions. Continuous action-based methods reproduce diverse motions, yet their reliance on dense supervision leads to overfitting, weakening perception and generalization. Hybrid methods combining keyposes and continuous actions seek a balance but fail to capture the object’s dynamic state, causing one arm to be unable to tilt the plate properly. Object-centric extensions enhance object awareness by leveraging movement information (e.g., object pointflow), but they overlook explicit robot–object interactions, which may lead to failures, such as a plate slipping before the other arm completes the grasp. Overall, these limitations make bimanual systems unstable in tasks that demand high-precision coordination.

In contrast, when performing bimanual operations, humans naturally sustain a continuous awareness of the **spatial triadic relationship**, that is, both the relation between the two hands and the relation between each hand and the manipulated object. Such triadic reasoning implicitly encodes inter-hand distances, hand–object relations, and their dynamic interactions, thereby ensuring stable and collision-free coordination. Consistently, findings from UMI [22] show that providing policies with the relative pose between grippers is crucial for achieving effective bimanual coordination. In the domain of dexterous manipulation, cutting-edge works such as $\mathcal{D}(\mathcal{R}, \mathcal{O})$ [23] and RobustDexGrasp [24] have also demonstrated that explicitly modeling hand–object interactions can significantly improve manipulation performance. Taken together, we argue that constructing a reasonable spatial triadic interaction for bimanual system and incorporating it into bimanual policies can enhance manipulation stability.

Motivated by this insight, we introduce **RoTri-Diff**, a novel IL framework guided by the spatial **Robot–Object Triadic Interaction** (*RoTri*). At its core, *RoTri* models the triadic spatial configuration formed by the two end-effectors and the manipulated object. By uniformly encoding their relative 6D pose, it establishes a continuous triangular geometric constraint that provides structured guidance for feasible and stable bimanual coordination. As shown in Fig. 1, RoTri-Diff is the first bimanual IL framework that integrates three key guidance signals: robot keyposes for long-horizon planning, object movement for capturing physical dynamics,

Method	Imitation Learning Guides			Prediction Type
	Robot Keyposes	Object Movement	Robot–Object Interaction	
ACT [17]	×	×	×	Continuous Actions
PerAct ² [10]	✓	×	×	Keyposes
DP3 [18]	×	×	×	Continuous Actions
AnyBimanual [14]	✓	×	×	Keyposes
3D Diffuser Actor [19]	✓	×	×	Keyposes & Continuous Actions
PPI [9]	✓	✓	×	Keyposes & Continuous Actions
RoTri-Diff (Ours)	✓	✓	✓	Keyposes & Continuous Actions

TABLE I: Comparison of representative bimanual imitation learning methods.

and the core *RoTri* representation for maintaining stable spatial relations. Architecturally, RoTri-Diff is a hierarchical diffusion model that can generate action sequences while maintaining consistency from global planning to fine-grained execution. The process includes three stages: (i) simultaneously predicting object pointflow and a continuous *RoTri* segment; (ii) generating keypose actions based on the predicted object pointflow and the *RoTri* at the keypose timesteps from (i); and (iii) integrating the three signals produced in (i) and (ii) to generate continuous action sequences. We extensively evaluate RoTri-Diff in both simulation and the real world. On the RL Bench2 bimanual benchmark [10], it achieves an **10.2% higher** success rate across 11 representative tasks compared to state-of-the-art baselines. Furthermore, in four challenging real-world bimanual manipulation tasks, RoTri-Diff demonstrates robust and stable performance.

In summary, our contributions are threefold:

- We introduce the concept of **RoTri** for bimanual manipulation, a triadic interaction representation that explicitly encodes spatial relations between the two arms and the object, enabling stable bimanual actions.
- We introduce **RoTri-Diff**, a hierarchical diffusion model that synergistically integrates robot keyposes, object dynamics, and *RoTri* constraints to generate spatially and temporally consistent trajectories.
- We provide extensive empirical validation, achieving state-of-the-art performance on 11 RL Bench2 tasks and robust real-world execution in four bimanual manipulation scenarios.

II. RELATED WORK

A. Imitation Learning for Bimanual Manipulation

Current imitation learning (IL) approaches for bimanual manipulation can be broadly categorized into two paradigms [10], [11], [14], [17]. The first is robot-centric, which directly models and predicts robot states. It includes: (i) keypose-based methods [10], [14] that predict discrete waypoints and rely on motion planners for execution; (ii) continuous action-based methods [4], [17], [18] that generate dense control sequences using Transformers or diffusion models; and (iii) hybrid methods [19], [20] that combine high-level keypose prediction with low-level trajectory generation. The second is object-centric, which uses object motion as the primary guidance signal for action generation [9], [21]. For instance, PPI [9] leverages 3D point-level object flow to enhance spatial grounding and manipulation precision. Despite their differences, both paradigms fail to explicitly

model the dynamic spatial relationship between the two arms and the object, limiting performance in contact-rich tasks requiring tight coordination and collision avoidance. To address this gap, we introduce the *RoTri* representation as a unified guidance signal. As shown in Table I, our method integrates robot-object interactions, robot keyposes, and object motion within a hierarchical framework, enabling robust prediction of both end-effector keyposes and continuous actions in complex bimanual scenarios.

B. Diffusion Models for Manipulation

Diffusion models, such as Denoising Diffusion Probabilistic Models (DDPM) [25], have achieved remarkable success in generative domains including image [26], [27] and video synthesis [28], [29]. Inspired by these advances, recent work has applied diffusion models to robotic action generation via imitation learning. These methods typically learn policies that iteratively denoise action trajectories, conditioned on perceptual inputs from 2D images [30]–[34] or 3D point clouds [18], [19], [35], [36]. In the context of bimanual manipulation, diffusion-based approaches have further explored unified action spaces [4], constrained image-conditioned policies [8], kinematics-aware action generation [12], and object-flow guidance [9]. While these advances mark important progress, they still inherit the key limitation outlined above: the lack of explicit modeling of the spatial triadic interactions between the arms and the object. In contrast, our proposed RoTri-Diff is the first diffusion-based framework to explicitly incorporate this relational geometry. By conditioning the denoising process on the *RoTri* representation, our hierarchical model significantly improves the stability and coherence of generated bimanual actions.

III. METHOD

In this section, we detail the overall architecture of **RoTri-Diff**. We start with a brief problem formulation (Sec. III-A), then describe visual perception and Robot-Object Triadic Interaction (*RoTri*) modeling (Sec. III-B), including 3D semantic feature construction, initial object point cloud acquisition, and building the unique *RoTri* representation from the 6D poses of the two arm end-effectors and the object. Next, we cover key imitation learning (IL) guidance signals: keyposes, object pointflow, and the *RoTri* relationship (Sec. III-C). We then present the action prediction module (Sec. III-D), a hierarchical, diffusion-based Transformer with synergistic attention for precise action generation. Finally, we detail the training and inference procedures (Sec. III-E). Fig. 2 provides an overview of our proposed framework.

A. Problem Formulation

Given a set of bimanual manipulation demonstrations $\mathcal{D} = \{(o_t, a_t)_{t=0}^{T_i}\}_{i=0}^N$, our goal is to train a RoTri-Diff network ϵ_θ that maps visual observations o to actions a . At timestep t , our model takes as input the language instruction and RGBD images from K cameras, and outputs a sequence of h^c continuous actions $a_t^c = \{a_{t:t+h^c-1}\}$, where each action a_t represents the target gripper poses and openness for

both the left and right manipulators. The model incorporates three key imitation learning (IL) guidance signals: Keyposes, Object Pointflow, and the Robot-Object Triadic Interaction (*RoTri*) representation. A keypose timestep t^k is defined as a turning point in the trajectory with significant changes in the grippers’ openness and the arms’ joint states [10], [19], [35], [37]. The first signal, Keyposes, specifies the target end-effector poses for the subsequent h^k keyposes, $a_t^k = \{a_{t^k}^k\}_{i=1}^{h^k}$. The second, Object Pointflow, defines the positions of N_q object point cloud points at the next h^k keyposes, $F \in \mathbb{R}^{h^k \times N_q \times 3}$. Finally, the third signal, the *RoTri* representation $R_t \in \mathbb{R}^{21}$, precisely quantifies the 6D relative poses of the two arms and the object.

B. Visual Perception and RoTri Modeling

Visual Perception. As is seen in Fig. 2(a), to obtain 3D semantic features that integrate both semantic and geometric information, we adopt the 3D scene representation method from PPI [9]. We first preprocess the raw point clouds through cropping and downsampling. We then use the DINOv2 model [38] to extract pixel-wise semantic features from multi-view 2D images. These features are fused via a weighted sum, with weights determined by the point’s distance from the projected surface. To mitigate computational burden, we use a PointNet++ dense encoder [39] to downsample the scene points into a compact representation $S_t \in \mathbb{R}^{N_s \times (3+D)}$. This compressed representation retains key geometric and semantic details while enhancing local point relationships through the set abstraction of PointNet++.

To enable the model to focus on capturing overall object motion rather than global absolute coordinates, thereby ensuring robust and generalized pointflow estimation even when the object’s position is out of distribution, we use the same initial object point cloud sampling strategy as in PPI [9], which approximates the conditional distribution $p(F|F_0)$. Here, F_0 represents the initial point cloud sampled from the manipulated object in the first frame. To acquire F_0 , we use the Grounding DINO [40] and SAM [41] models to generate an object mask from a language prompt and an image. We then randomly sample $N_q = 200$ pixels from the mask and obtain their 3D coordinates $F_0 \in \mathbb{R}^{N_q \times 3}$. Consistent with the setup in PPI [9], this sampling operation is performed only once per episode.

RoTri Modeling. The core of our method is to capture the dynamic triadic interaction formed by the two arms and the manipulated objects. We define the Robot-Object Triadic Interaction (*RoTri*) vector as a compact representation of this relationship. To obtain the initial *RoTri* vector, we first acquire the 6D absolute poses of the two arm end-effectors and the object at the initial timestep. We then compute the relative poses between each pair and concatenate them into a single vector: $R_0 = [p_{\text{left} \rightarrow \text{right}}^0, p_{\text{left} \rightarrow \text{obj}}^0, p_{\text{right} \rightarrow \text{obj}}^0] \in \mathbb{R}^{21}$, where each 7-dimensional component consists of a 3D position and a 4D quaternion (x, y, z, w) . For training supervision, this representation is computed at every step of the expert trajectory, yielding the ground-truth sequence $\{R_t\}_{t=0}^T$. For tasks with multiple manipulated objects, we extend *RoTri*

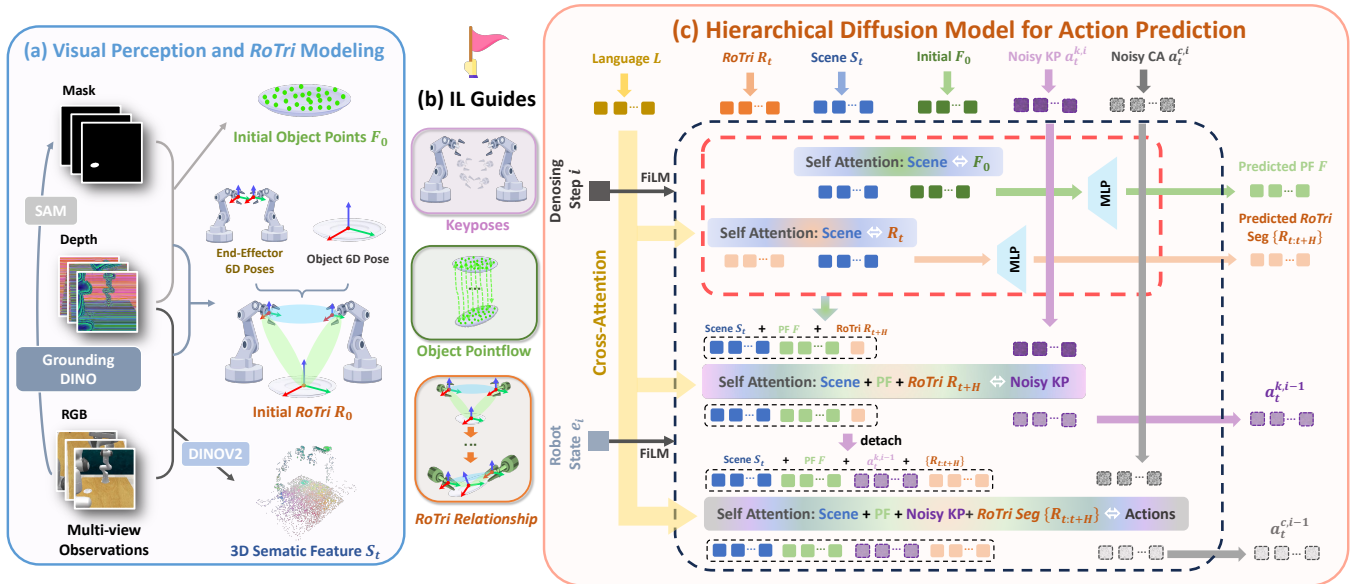


Fig. 2: **Overview of RoTri-Diff.** (a) **Visual Perception and RoTri Modeling:** Extracting the initial object point cloud F_0 , 3D semantic features S_t , and the initial RoTri representation R_0 from multi-view observations. (b) **Imitation Learning Guidance Signals:** Three complementary signals used for supervision: Keypose, Object Pointflow, and the RoTri Relationship. (c) **Hierarchical Diffusion Model:** The model concurrently predicts object pointflow and autoregressively predicts a future RoTri segment. These predictions then serve as dynamic conditions to guide the denoising and generation of keyposes and continuous actions within a synergistic attention module.

modeling to handle variable interactions while maintaining a fixed-size representation. We construct a set of RoTri vectors, each describing the triadic interaction with one object, and aggregate them using a permutation-invariant Transformer encoder. The encoder outputs a high-dimensional embedding summarizing the most relevant interactions, which is then projected back into a fixed 21-dimensional vector. This allows the model to dynamically focus on the relevant objects while remaining scalable and robust to varying object counts.

C. IL Guidance Signals

Keypose & Object Pointflow. Following PPI [9], we generate two key IL guidance signals. First, we employ the heuristic algorithm [37] to extract keyposes $a_{t_k}^k$ from expert trajectories, which are defined as points where joint velocity approaches zero or gripper open/close events occur, and these serve as targets to guide continuous action generation. Second, to address the occlusion issue caused by object motion, we leverage the object’s 6D pose (obtained from simulation [10] or the real world [42], [43]) to transform initial object point cloud F_0 to their future keypose positions $F_{t_k} \in \mathbb{R}^{N \times 3}$. These transformed points serve as the ground truth for pointflow supervision and eliminate the need for real-time pose estimation during inference.

RoTri Relationship. Instead of directly predicting the complete RoTri trajectory, we adopt a more robust approach by learning its dynamic evolution through approximating the conditional distribution $p(R|R_0)$. Specifically, the model predicts in an incremental fashion: at each timestep, it predicts the change in the RoTri representation, ΔR_t . The subsequent RoTri representation R_t is then obtained by accumulating this predicted change onto the previous timestep’s representation,

i.e., $R_t = R_{t-1} + \Delta R_t$. This predictive paradigm shifts the learning focus from inferring complex absolute poses to capturing relative interaction dynamics.

D. Hierarchical Diffusion Model for Action Prediction

Observation Encoder. The observation encoder of RoTri-Diff processes and encodes multi-modal inputs, including the 3D semantic features, language instruction, robot states, initial object points, and the initial RoTri representation into a unified latent space. The language instruction is processed by a CLIP-based text encoder, while the robot states (which include pose and gripper state for both arms), initial object points, and the initial RoTri representation are each independently projected into the latent space via separate three-layer MLPs. These encoded features are then integrated to form the final observational conditioning for the model.

Action Prediction. As shown in Fig. 2(c), at timestep t and denoising step i , RoTri-Diff integrates RoTri tokens R_t , scene tokens S_t , language tokens L , initial object points tokens F_0 , and noised action tokens $a_t^{k,i}$ and $a_t^{c,i}$. Each token comprises a latent embedding and a 3D position. Following 3D Diffuser Actor [19] and PPI [9], we apply Relative Self-Attention with Rotary Positional Embedding to effectively leverage the relative 3D spatial information between tokens. The robot’s proprioception e_t and denoising timesteps i affect the attention through Feature-wise Linear Modulation (FiLM) [44]. All tokens first attend to the language tokens L via Parallel Attention. Subsequently, the model initiates two parallel processes: **1) Object pointflow (PF) prediction:** In a one-shot process, initial object points tokens F_0 attend to scene tokens S_t . An MLP then predicts a 12D representation (xyz coordinates for subsequent keyposes) for each object

points. **2) Auto-regressive RoTri prediction:** This process follows a hierarchical auto-regressive prediction mechanism, iterating on a keypose-by-keypose basis. The model auto-regressively predicts the *RoTri* segment $\{R_{t:t+H}\}$ for the next H timesteps. Each prediction is conditioned on the previous keypose step’s *RoTri* state (initially R_0) attending to scene tokens and immediately serves as a dynamic condition for action generation. Rather than a single-shot prediction, the model proceeds step-wise across keypose segments, predicting each segment’s *RoTri* trajectory conditioned on the previous segment’s final *RoTri* (or R_0 for the first segment).

The predicted *RoTri* segment conditions keypose action generation. Noised keypose tokens $a_t^{k,i}$ attend to the object PF, scene, and the specific *RoTri* token at the keypose timestep (R_{t+H}) via Relative Self-Attention. An MLP maps the output to the action dimension, supervised by keypose noise $\epsilon_t^{k,i}$. Then, noised continuous action tokens $a_t^{c,i}$ attend to all prior features, including the new keypose and full *RoTri* segment. Another MLP predicts continuous noise $\epsilon_t^{c,i}$ for supervision. The final state of the predicted *RoTri* segment R_{t+H} feeds into the next iteration, completing the auto-regressive chain.

E. Implementation Details

Training. For the RoTri-Diff network ϵ_θ , all losses are computed via an L_1 norm. Given the conditioning inputs (3D features S_t , language L , initial points F_0 , initial *RoTri* R_0 , and proprioception e_t), the loss functions are:

$$\begin{aligned}\mathcal{L}_c &= \|\epsilon_\theta(S_t, L, F_0, R_0, e_t, a_t^{c,i}, i) - \epsilon_c\|_1 \\ \mathcal{L}_k &= \|\epsilon_\theta(S_t, L, F_0, R_0, e_t, a_t^{k,i}, i) - \epsilon_k\|_1 \\ \mathcal{L}_{pf} &= \|\epsilon_{\theta, pf}(S_t, L, F_0, e_t, i) - F\|_1 \\ \mathcal{L}_{rotri} &= \|\epsilon_{\theta, rotri}(S_t, L, R_0, e_t, i) - \Delta R_t\|_1\end{aligned}$$

Here, $a_t^{c,i}$ and $a_t^{k,i}$ are noised actions, while ϵ_c and ϵ_k are the ground truth noise. $\epsilon_{\theta, pf}$ and $\epsilon_{\theta, rotri}$ denote the prediction heads for the object PF F and the change in RoTri representation ΔR_t . The total training loss is:

$$\mathcal{L}_\theta = w_c \mathcal{L}_c + w_k \mathcal{L}_k + w_{pf} \mathcal{L}_{pf} + w_{rotri} \mathcal{L}_{rotri},$$

with weights $w_c, w_k = 0.05$ and $w_{pf}, w_{rotri} = 1.0$. In all experiments, we use a DDPM noise schedule with 1000 training timesteps. For tasks from the RL Bench benchmark, we train for 500 epochs; for real-world tasks, we train for 1000 epochs. We use a batch size of 64, an AdamW optimizer with a learning rate of $1e-4$, and a cosine decay learning rate scheduler, and conduct model training on eight A5000 GPUs. We select the checkpoint with the lowest average validation loss for evaluation.

Inference. At the start of an episode ($t = 0$), we sample $N_q = 200$ object points F_0 and compute the initial *RoTri* representation R_0 , which serve as global conditions. At each key decision point t , the policy generates actions by starting with random noise $\hat{a}_t^N \sim \mathcal{N}(0, \mathbf{I})$ and performing an iterative hierarchical diffusion process. In each denoising step, the model predicts the future PF F and *RoTri* segment

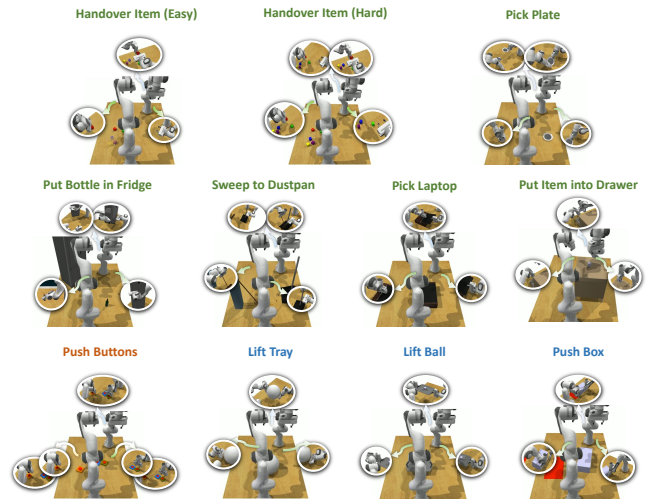


Fig. 3: Visualization of the 11 RL Bench2 tasks.

$\{R_{t:t+H}\}$ as dynamic guidance. After a set number of steps (1000 for DDPM in sim, 20 for DDIM in real), the process yields a clean sequence of 50 continuous actions. The entire generated action sequence is then sent to the robot controller for complete execution until the next keypose is reached, at which point the planning cycle repeats.

IV. EXPERIMENTS

In this section, we perform a series of experiments aimed at addressing the following questions:

- Q1:** How well does RoTri-Diff perform on complex bimanual manipulation tasks in simulation, and can it outperform state-of-the-art methods?
- Q2:** What critical role does the *RoTri* representation play? Among keypose guidance, continuous action prediction, and the synergy between the two, which application of *RoTri* is most critical for improving model performance?
- Q3:** Can RoTri-Diff maintain stable performance on real-world bimanual tasks that require precise spatial coordination and strict action constraints?

A. Simulation Results

Simulation Setup. To answer **Q1**, we perform extensive simulation experiments in RL Bench2 [10], a bimanual manipulation benchmark built on CoppeliaSim featuring diverse tasks with varying coordination demands. As shown in Fig. 3, we select eleven challenging tasks, grouped into three bimanual coordination types [10]: 1) **Symmetric Coordination:** Both arms move simultaneously on separate objects. The challenge lies in parallel operation within a shared workspace, requiring continuous perception of spatial relations to plan collision-free trajectories. Task: *Push Buttons*. 2) **Synchronous Coordination:** Both arms jointly manipulate a single object. The difficulty is precise coordination of relative poses to prevent drops or collisions. Tasks: *Lift Tray*, *Lift Ball*, *Push Box*. 3) **Asynchronous Coordination:** Arms perform dependent sequential actions on the same object. The challenge is capturing accurate spatiotemporal interaction to ensure reliable execution while avoiding drops

	Avg. Success (%)	push buttons	lift ball	lift tray	push box	handover item (easy)	handover item (hard)	pick laptop	put item into drawer	sweep to dustpan	pick plate	bottle in fridge
ACT [17]	5.9	4.0	38.3	1.3	67.0	0.0	0.0	0.0	1.7	0.0	0.0	0.0
PerAct ² [10]	16.8	47.0	50.0	60.0	62.0	19.7	11.0	36.7	49.7	2.0	4.0	3.0
DP3 [18]	-	-	27.0	6.3	39.3	4.7	-	6.0	0.0	98.7	-	-
AnyBimanual [14]	39.9	73.0	36.0	67.0	46.0	44.0	15.0	7.0	50.0	67.0	8.0	26.0
3DDA [19]	-	-	87.3	76.0	54.7	44.7	-	40.7	52.7	96.7	-	-
PPI [9]	70.7	92.0	89.3	92.0	96.7	62.7	37.3	46.3	79.7	98.7	0.0	82.6
RoTri-Diff (ours)	80.9 (10.2%↑)	97.0\pm1.73	95.7\pm1.15	94.3\pm1.15	95.0\pm1.73	73.3\pm2.89	52.3\pm2.52	66.0\pm3.46	87.0\pm1.73	96.67\pm2.89	40.7\pm1.15	92.0\pm1.73

TABLE II: **Performance on 11 RL Bench2 tasks.** AnyBimanual is a multi-task policy evaluated with a single checkpoint across all tasks, whereas the other methods are single-task policies that report results from the best checkpoint for each task [9], [20]. RoTri-Diff is evaluated with three seeds per task, reporting the mean and standard deviation.

or collisions. Tasks: *Put Button in Fridge*, *Sweep to Dustpan*, *Pick Laptop*, *Put Item into Drawer*, *Handover Item (Easy)*, *Handover Item (Hard)*, *Pick Plate*.

Baselines. We compare RoTri-Diff with a diverse set of baselines to highlight its advantages across different levels. ACT [17] and DP3 [18] directly predict continuous action sequences without leveraging high-level guidance. PerAct² [10] and AnyBimanual [14] employ a Perceiver-based architecture [45] to predict discrete keyposes. 3D Diffuser Actor (3DDA) [19] further integrates diffusion to predict both keyposes and full action trajectories. PPI [9] combine keypose and object-movement guidance, achieving strong performance on tasks with dynamic objects. DP3 and 3DDA are evaluated on only a subset of the 11 tasks.

Metric. Our method is evaluated with 100 rollouts per task under randomized initial states. We report the success rate (%) for each task as well as the average across all tasks, where results are averaged over three independent checkpoints and presented with mean and standard deviation.

Overall Performance. As shown in Table II, our RoTri-Diff model achieves an average success rate of **80.9%** across 11 tasks. This result not only significantly outperforms all baselines but also represents a **10.2% improvement** over the previous state-of-the-art. Methods that rely solely on continuous action prediction, such as ACT and DP3, perform poorly on tasks requiring fine-grained control; for instance, DP3 achieves only 0.0% on *put item into drawer*. Similarly, keypose-based methods like PerAct² and AnyBimanual struggle with challenging coordination tasks. On *handover item (hard)*, for example, AnyBimanual achieves only 15.0%, whereas our RoTri-Diff reaches 52.3%. Although 3DDA combines keyposes with continuous actions, it lacks explicit modeling of object dynamics. As a result, it underperforms significantly in asynchronous coordination tasks: on *handover item (easy)*, it achieves only 44.7%, compared to RoTri-Diff’s 73.3%. Similarly, on *put item into drawer*, RoTri-Diff achieves 87.0%, far exceeding 3DDA’s 52.7%.

Against the state-of-the-art hybrid baseline PPI, our RoTri-Diff demonstrates clearer advantages. While PPI is the first to introduce object movement guidance, it completely fails on the highly constrained *pick plate* task (0.0%), whereas RoTri-Diff achieves 40.7%. On *handover item (easy)* and *put item into drawer*, RoTri-Diff also surpasses PPI by large margins, reaching 73.3% vs. 62.7% and 87.0% vs. 79.7%, respectively. In summary, by explicitly modeling the

Method	push buttons	lift tray	pick plate	bottle in fridge
<i>RoTri</i> (Keypose Only)	93.7	92.6	29.3	90.3
<i>RoTri</i> (Continuous Only)	95.3	88.7	21.3	86.0
RoTri-Diff	97.0	94.3	40.7	92.0

TABLE III: **Ablation Study.** Comparison of two architectural variants with RoTri-Diff, confirming the necessity of combining keypose and continuous *RoTri* guidance in a full hierarchical design.

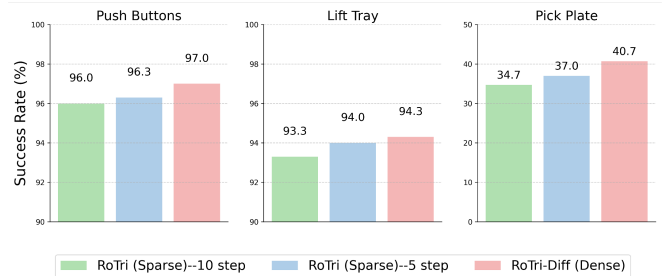


Fig. 4: **Ablation Study.** Comparing Dense with Sparse (5/10 step) variants shows that higher guidance density consistently yields superior task success.

RoTri relationship, our method enables the system to directly perceive the spatial relation among the two arms and the manipulated object, thereby enhancing its performance in tasks that require precise coordination and spatial interaction.

Ablation Study. To address Q2, we conduct three ablation studies to examine the role of *RoTri* representations and their synergy with the hierarchical prediction architecture. We design three variants: ***RoTri* (Keypose Only)**: This variant uses *RoTri* only at the keypose timestep to guide high-level keypose prediction, while continuous actions are generated from predicted keyposes, scene features, and point cloud flow. It tests whether *RoTri* mainly benefits high-level planning or also continuous execution. ***RoTri* (Continuous Only)**: This variant removes the keypose prediction module and directly uses the full *RoTri* relationship to generate continuous action sequences, examining whether *RoTri* alone can replace the hierarchical structure. ***RoTri* (Sparse)**: This variant studies the effect of temporal density by providing *RoTri* guidance at fixed intervals (every 5 or 10 timesteps) between keyposes, evaluating how dense step-wise guidance affects performance.

As shown in Table III, both representation and hierarchy

are essential. In the Keypose Only variant, performance on *Pick Plate* drops to 29.3% (vs. 40.7% for RoTri-Diff), showing that continuous constraints are needed for smooth, fine-grained actions. Conversely, the Continuous Only variant reaches 88.7% on *Lift Tray* but only 21.3% on *Pick Plate*, suggesting keyposes provide a crucial predictive anchor for complex tasks. Fig. 4 further highlights the importance of temporal guidance density. RoTri-Diff consistently outperforms sparse variants: for example, on *Push Buttons*, success drops from 97.0% (dense) to 96.3% (5-step) and 96.0% (10-step), with larger gaps in precision-heavy tasks such as *Pick Plate* (40.7% vs. 37.0%). This shows that dense per-timestep *RoTri* guidance prevents drift and error accumulation. Overall, the ablation confirms that combining explicit *RoTri* representation, hierarchical prediction, and dense temporal guidance is key to robust bimanual coordination.

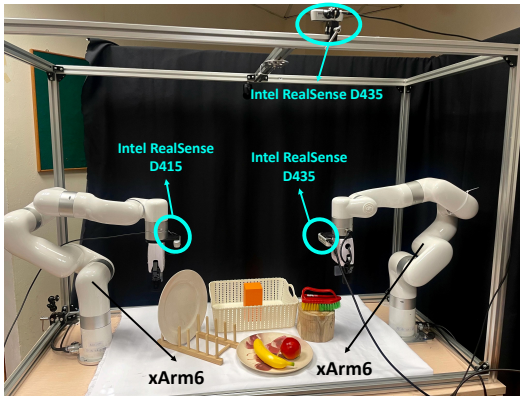


Fig. 5: Real-world experimental setup.

B. Real-world Results

Real-world Setup. To answer Q3, we evaluate the proposed method on two xArm6 robots, with the setup shown in Fig. 5. Each scene includes two Eye-on-Hand cameras and one Eye-on-Base RealSense camera. Four real-world tasks are designed, covering the three coordination types: 1) **Pick Tomato & Banana** (Symmetric Coordination): Both arms simultaneously grasp a tomato and a banana from the table and place them into a fruit bowl. 2) **Pick Plate** (Asynchronous Coordination): One arm tilts the plate by pressing its edge, while the other grasps the lifted side. 3) **Wash Plate** (Asynchronous Coordination): One arm retrieves a plate from the shelf and positions it, while the other grasps a brush and wipes the plate. 4) **Lift Basket** (Synchronous Coordination): Both arms collaboratively lift a basket with stacked blocks and place it onto a wooden post while keeping the blocks upright. The task processes are illustrated in Fig. 6. Demonstrations are collected via the Meta Quest 2 teleoperation platform: 50 for “Lift Basket” and “Pick Plate”, and 20 for the other tasks.

Metric. We report the total number of successful outcomes of RoTri-Diff across four real-world manipulation tasks, summing the successful trials (1 if the task is completed, 0 otherwise) over 5 trials per task. This total reflects overall performance across all trials.

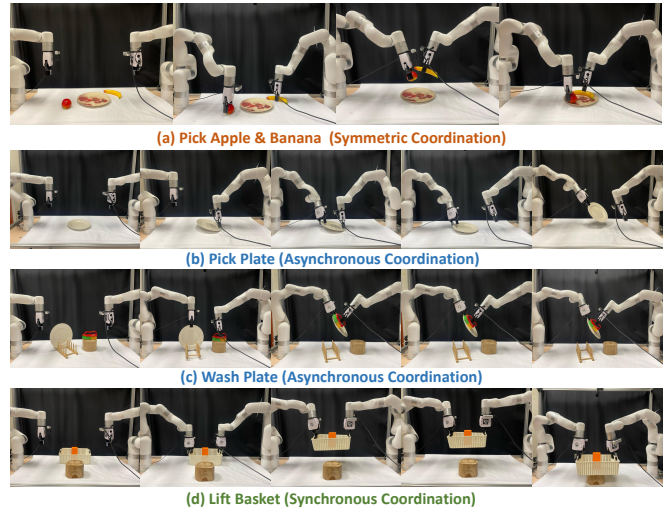


Fig. 6: Task process visualizations in 4 real-world tasks.

Method	Pick Tomato & Banana	Pick Plate	Wash Plate	Lift Basket
RoTri-Diff	5/5	3/5	4/5	4/5

TABLE IV: Real-world experiment results.

Real-world Performance. As shown in Table IV, we evaluate RoTri-Diff across four real-world manipulation tasks using the number of successful trials (out of 5 trials per task). In the “Pick Tomato & Banana” task, RoTri-Diff achieves 5 successful trials out of 5, demonstrating its ability to plan and execute two independent parallel grasps within a shared workspace while avoiding collisions. In the “Pick Plate” and “Wash Plate” tasks involving complex sequential manipulation, RoTri-Diff achieves 3 out of 5 and 4 out of 5 successful trials, respectively, highlighting its capability to handle tasks with strict temporal dependencies and precise bimanual coordination. Notably, in the “Pick Plate” task, one arm first tilts and stabilizes the plate before the other grasps the lifted side, imposing strict spatial constraints between the two arms and the object. In the “Lift Basket” task, RoTri-Diff achieves 4 successful trials out of 5, indicating its ability to maintain stable and synchronous spatial coordination between the two arms and the object under load. Taken together, these results demonstrate that RoTri-Diff can robustly address the diverse complexities of real-world bimanual manipulation.

V. CONCLUSION AND LIMITATION

We propose RoTri-Diff, a diffusion-based imitation learning framework for bimanual manipulation that explicitly models the robot–object triadic interaction (*RoTri*) for the first time. The *RoTri* representation encodes the relative 6D poses between the two end-effectors and the manipulated object, thereby capturing their triadic spatial relationship through continuous triangular geometric constraints. Built upon this representation, RoTri-Diff integrates robot keyposes, object pointflow, and *RoTri* constraints within a hierarchical diffusion process to generate temporally and spatially stable bimanual actions. Extensive experiments show that

RoTri-Diff not only outperforms state-of-the-art baselines across simulation tasks with different coordination modes, but also achieves stable performance in 4 real-world challenging tasks, paving the way toward human-level bimanual manipulation in diverse scenarios.

Despite its strengths, RoTri-Diff remains limited by its reliance on rigid-body assumptions and accurate 6D pose estimation, restricting generalization to deformable objects and unstructured environments. Future work will aim to establish a more general triadic interaction representation capable of handling deformable objects, supporting cross-embodiment transfer, and adapting to diverse bimanual platforms.

REFERENCES

- [1] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi *et al.*, “Openvla: An open-source vision-language-action model,” in *CoRL*, 2024.
- [2] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu *et al.*, “Rt-1: Robotics transformer for real-world control at scale,” in *Arxiv*, 2022.
- [3] D. Driess, F. Xia, M. S. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu *et al.*, “Palm-e: An embodied multimodal language model,” in *Arxiv*, 2023.
- [4] S. Liu, L. Wu, B. Li, H. Tan, H. Chen, Z. Wang, K. Xu, H. Su, and J. Zhu, “Rdt-1b: a diffusion foundation model for bimanual manipulation,” in *ICLR*, 2025.
- [5] Y. Mu, T. Chen, Z. Chen, S. Peng, Z. Lan, Z. Gao, Z. Liang, Q. Yu, Y. Zou, M. Xu *et al.*, “Robotwin: Dual-arm robot benchmark with generative digital twins,” in *CVPR*, 2025.
- [6] Z. Chen, Z. Ji, J. Huo, and Y. Gao, “Scar: Refining skill chaining for long-horizon robotic manipulation via dual regularization,” in *NeurIPS*, 2024.
- [7] D. Yu, H. Xu, Y. Chen, Y. Ren, and J. Pan, “Bike: Keypose-conditioned consistency policy for bimanual robotic manipulation,” in *Arxiv*, 2024.
- [8] I. Liu, C. Arthur, J. Chen, G. Sukhatme, and D. Seita, “D-coda: Diffusion for coordinated dual-arm data augmentation,” in *Arxiv*, 2025.
- [9] Y. Yang, Z. Cai, Y. Tian, J. Zeng, and J. Pang, “Gripper keypose and object pointflow as interfaces for bimanual robotic manipulation,” in *RSS*, 2025.
- [10] M. Grotz, M. Shridhar, T. Asfour, and D. Fox, “Peract2: A perceiver actor framework for bimanual manipulation tasks,” in *Arxiv*, 2024.
- [11] J. Grannen, Y. Wu, B. Vu, and D. Sadigh, “Stabilize to act: Learning to coordinate for bimanual manipulation,” in *CoRL*, 2023.
- [12] Q. Lv, H. Li, X. Deng, R. Shao, Y. Li, J. Hao, L. Gao, M. Y. Wang, and L. Nie, “Spatial-temporal graph diffusion policy with kinematic modeling for bimanual robotic manipulation,” in *CVPR*, 2025, pp. 17 394–17 404.
- [13] Z. Chen, J. Yin, Y. Chen, J. Huo, P. Tian, J. Shi, Y. Hou, Y. Li, and Y. Gao, “Deco: Task decomposition and skill composition for zero-shot generalization in long-horizon 3d manipulation,” 2025.
- [14] G. Lu, T. Yu, H. Deng, S. S. Chen, Y. Tang, and Z. Wang, “Anybimanual: Transferring unimanual policy for general bimanual manipulation,” in *ICCV*, 2025.
- [15] I. A. Sucas, M. Moll, and L. E. Kavraki, “The open motion planning library,” in *IEEE Robotics & Automation Magazine*, vol. 19, no. 4. IEEE, 2012, pp. 72–82.
- [16] B. Sundaralingam, S. K. S. Hari, A. Fishman, C. Garrett, K. Van Wyk, V. Blukis, A. Millane, H. Oleynikova, A. Handa, F. Ramos *et al.*, “Curobo: Parallelized collision-free robot motion generation,” in *ICRA*, 2023.
- [17] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, “Learning fine-grained bimanual manipulation with low-cost hardware,” in *RSS*, 2023.
- [18] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu, “3d diffusion policy,” in *RSS*, 2024.
- [19] T.-W. Ke, N. Gkanatsios, and K. Fragkiadaki, “3d diffuser actor: Policy diffusion with 3d scene representations,” in *CoRL*, 2024.
- [20] N. Gkanatsios, J. Xu, M. Bronars, A. Mousavian, T.-W. Ke, and K. Fragkiadaki, “3d flowmatch actor: Unified 3d policy for single- and dual-arm manipulation,” 2025.
- [21] Y. Su, X. Zhan, H. Fang, Y.-L. Li, C. Lu, and L. Yang, “Motion before action: Diffusing object motion as manipulation condition,” in *IEEE Robotics and Automation Letters*. IEEE, 2025.
- [22] C. Chi, Z. Xu, C. Pan, E. Cousineau, B. Burchfiel, S. Feng, R. Tedrake, and S. Song, “Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots,” in *RSS*, 2024.
- [23] Z. Wei, Z. Xu, J. Guo, Y. Hou, C. Gao, Z. Cai, J. Luo, and L. Shao, “ $\mathcal{D}(\mathcal{R}, \mathcal{O})$ grasp: A unified representation of robot and object interaction for cross-embodiment dexterous grasping,” in *ICRA*. IEEE, 2025, pp. 4982–4988.
- [24] H. Zhang, Z. Wu, L. Huang, S. Christen, and J. Song, “RobustDex-Grasp: Robust dexterous grasping of general objects,” in *CoRL*, 2025.
- [25] J. Ho, A. Jain, and P. Abbeel, “Denosing diffusion probabilistic models,” in *NeurIPS*, 2020.
- [26] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach, “Sdxl: Improving latent diffusion models for high-resolution image synthesis,” in *ICLR*, 2024.
- [27] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” in *CVPR*, 2022.
- [28] Z. Wang, C. Lu, Y. Wang, F. Bao, C. Li, H. Su, and J. Zhu, “Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation,” in *NeurIPS*, vol. 36, 2023.
- [29] W. Peebles and S. Xie, “Scalable diffusion models with transformers,” in *ICCV*, 2023.
- [30] A. Z. Ren, J. Lidard, L. L. Ankile, A. Simeonov, P. Agrawal, A. Majumdar, B. Burchfiel, H. Dai, and M. Simchowitz, “Diffusion policy optimization,” in *Arxiv*, 2024.
- [31] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song, “Diffusion policy: Visuomotor policy learning via action diffusion,” in *RSS*, 2023.
- [32] X. Ma, S. Patidar, I. Haughton, and S. James, “Hierarchical diffusion policy for kinematics-aware multi-task robotic manipulation,” in *CVPR*, 2024.
- [33] V. Vosylius, Y. Seo, J. Uruç, and S. James, “Render and diffuse: Aligning image and action spaces for diffusion-based behaviour cloning,” in *RSS*, 2024.
- [34] Y. Wang, Y. Zhang, M. Huo, T. Tian, X. Zhang, Y. Xie, C. Xu, P. Ji, W. Zhan, M. Ding *et al.*, “Sparse diffusion policy: A sparse, reusable, and flexible policy for robot learning,” in *CoRL*, 2025.
- [35] Y. Chen, Z. Chen, J. Yin, J. Huo, P. Tian, J. Shi, and Y. Gao, “Gravmad: Grounded spatial value maps guided action diffusion for generalized 3d manipulation,” in *ICLR*, 2025.
- [36] Y. Ze, Z. Chen, W. Wang, T. Chen, X. He, Y. Yuan, X. B. Peng, and J. Wu, “Generalizable humanoid manipulation with improved 3d diffusion policies,” in *Arxiv*, 2024.
- [37] S. James and A. J. Davison, “Q-attention: Enabling efficient learning for vision-based robotic manipulation,” in *IEEE Robotics and Automation Letters*, 2022.
- [38] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, R. Howes, P.-Y. Huang, H. Xu, V. Sharma, S.-W. Li, W. Galuba, M. Rabbat, M. Assran, N. Ballas, G. Synnaeve, I. Misra, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, “Dinov2: Learning robust visual features without supervision,” in *Arxiv*, 2023.
- [39] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, “Pointnet++: Deep hierarchical feature learning on point sets in a metric space,” in *Arxiv*, 2017.
- [40] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, C. Li, J. Yang, H. Su, J. Zhu *et al.*, “Grounding dino: Marrying dino with grounded pre-training for open-set object detection,” in *Arxiv*, 2023.
- [41] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, “Segment anything,” in *Arxiv*, 2023.
- [42] B. Wen, J. Tremblay, V. Blukis, S. Tyree, T. Müller, A. Evans, D. Fox, J. Kautz, and S. Birchfield, “Bundlesdf: Neural 6-dof tracking and 3d reconstruction of unknown objects,” in *CVPR*, 2023, pp. 606–617.
- [43] W. Bowen, Y. Wei, K. Jan, and B. Stan, “FoundationPose: Unified 6d pose estimation and tracking of novel objects,” in *CVPR*, 2024.
- [44] E. Perez, F. Strub, H. de Vries, V. Dumoulin, and A. Courville, “Film: Visual reasoning with a general conditioning layer,” in *AAAI*, 2017.
- [45] A. Jaegle, S. Borgeaud, J.-B. Alayrac, C. Doersch, C. Ionescu, D. Ding, S. Koppula, D. Zoran, A. Brock, E. Shelhamer *et al.*, “Perceiver io: A general architecture for structured inputs & outputs,” in *ICLR*, 2022.