MOP: A Multimodal Object-aware Policy for Robotic Manipulation via Geometry-guided Fusion and Trajectory Prediction

***Abstract*—3D imitation learning has demonstrated capability in diverse visuomotor tasks but often struggles with small objects or high-precision manipulation due to geometric sparsity. To address this, we propose the Multimodal Object-aware Policy (MOP), a novel framework that incorporates a geometry-guided fusion module to adaptively integrate 2D semantic features with 3D geometry for precise control. Additionally, we introduce a lightweight Object Position Prediction (OPP) module that serves as an auxiliary supervision signal, eliminating the need for expensive motion capture systems during training. We evaluate our policy across 56 tasks on 3 simulation benchmarks. Experimental results demonstrate that MOP significantly outperforms baselines, achieving higher success rates with lower variance and efficient inference. Real-world experiments on 4 tasks further validate the robustness and transferability of our approach. The project resources are available at https://reshmouqi01.github.io/MOPpage/.**

***Index Terms*—Learning from Demonstration, Deep Learning in Grasping and Manipulation, Perception for Grasping and Manipulation.**

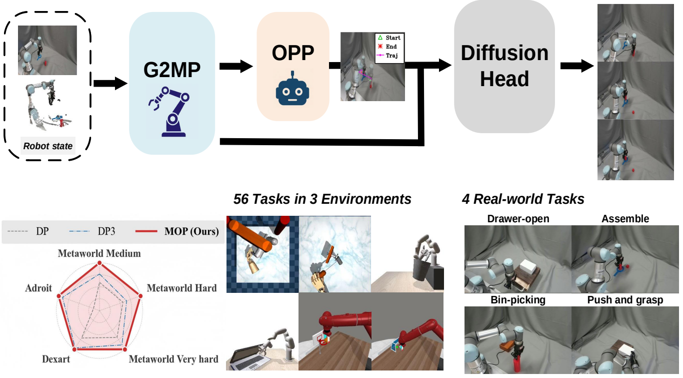

Fig. 1: **Multimodal object-aware policy (MOP)** is a multimodal imitation learning method, enabling high accuracy and strong robustness on various simulation benchmarks and real-world tasks.

I. Introduction

In the dynamic and unstructured real-world environments, imitation learning enables the robots to learn human-like manipulation skills. To better understand the physical world, robotic perception systems play an essential role in the embodied intelligence. By incorporating 2D visual perception, transformer-based and diffusion-based imitation learning architectures are adopted to predict the desired actions [1, 2]. Recent works have shown that leveraging 3D representations with point clouds [3–5] and voxels [6] can significantly improve the robotic manipulations in terms of accuracy and robustness, compared to the 2D counterpart.

Prior 3D diffusion strategies are capable of solving conventional manipulation tasks, however, when confronted with small objects or high-precision tasks [7], they may not efficiently address these challenging tasks due to the limited spatial understanding. As such, rich 3D semantics and accurate geometry are required in more complex environments. Recent work [8] adopts NeRF for pre-training to learn 3D representations, in order to accomplish difficult robotic manipulation. To redeem the insufficient geometric information, the pre-trained large vision-language models [9, 10] or language-guided high-level planning [11, 12] are leveraged to understand the semantics, but these approaches could not guarantee low-level geometric alignment. Existing contributions such as [13, 14] fuse the 2D semantics with 3D geometry for spatial feature alignment, and consider point cloud and image features to be equally important. Such consideration neglects dynamic modality combinations in favor of dense semantics, resulting in reduced accuracy and computational efficiency. Therefore, adaptive multimodal fusion is an important means to enhance efficiency and robustness [15]. In robotic manipulation tasks, leveraging object motion priors is crucial for integrating perception with control. To capture fine-grained motion cues, some approaches such as transferable affordance [16], trajectory modeling for videos [17] and 3D scene flow [18] have been studied in the literature. With the help of an expensive motion capture (MoCap) system, [19] designs two cascaded diffusion processes for object pose prediction and action generation, which limits the system scalability and inference speed.

To address the aforementioned issues, this work proposes a multimodal object-aware policy (MOP) (Fig. 1), which is a novel imitation learning framework. Different from existing designs, MOP develops a geometry-guided multimodal perception module and a lightweight object position prediction module that eliminates the dependency on external motion capture (MoCap) systems, collectively enabling efficient and robust robotic manipulation. To comprehensively assess MOP performance, we conduct simulations across 56 tasks on 3 simulation benchmarks including Adroit [20], DexArt [7] and Meta-World [21]. The results confirm that MOP substantially improves the performance and accelerates inference, compared to the baseline methods. The real-robot experiments further demonstrate the efficacy of our method. The main contributions of this work are as follows:

- We propose a geometry-guided multimodal perception (G2MP) module to enable robust multimodal perception. Through employing an asymmetric architecture with adaptive gating, G2MP dynamically regulates the injection of RGB semantics into 3D geometric representations. This mechanism upweights visual representations to compensate for sparsity in challenging tasks. Consequently, MOP effectively resolves modality competition and handles complex scenarios.

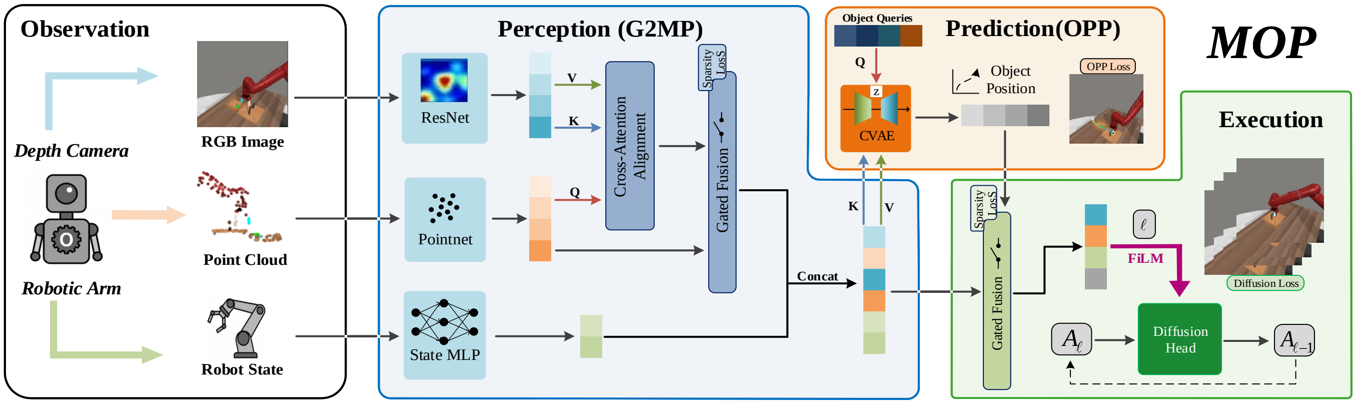

Fig. 2: **Overview of the multimodal object-aware policy (MOP).** Based on observations captured by RGB-D camera and robot state, MOP adopts the geometry-guided multimodal perception (G2MP) for dynamically fusing the semantics and geometric information. In the object position prediction (OPP) module, a transformer-based conditional variational autoencoder (CVAE) is introduced. Finally, conditioned on the fused feature, actions are generated by the diffusion head.

- We design a lightweight Object Position Prediction (OPP) module to efficiently predict 3D object positions. By introducing rich trajectory priors to facilitate the training of the diffusion policy, this module eliminates the need for MoCap systems and enhances fine-grained manipulation capabilities with negligible computational overhead.

II. Related Works

A. Imitation Learning in Robotic Manipulation

Imitation learning aims to acquire the robotic skills from expert demonstrations. Behavioral cloning (BC) [22] is a classic imitation learning approach to directly mapping observations to actions, and formulates the policy learning as a supervised regression problem. Due to the compounding errors [23], BC cannot efficiently learn long-horizon tasks. To cope with this issue, new imitation learning methods such as implicit BC [24] and Transformer-based architecture [1] have been developed, in which the attention mechanism and action chunking are leveraged to capture action trajectories. Recently, the diffusion-based imitation learning [2–5] has gained great research attention as a new policy paradigm. Inspired by the success of achieving image generation and semantic scene synthesis tasks [25], diffusion policies iteratively denoise random noise into action sequence. Later work [26] supports fast sampling of diffusion policies for goal-specific action generation. However, effective observation encoder needs to be designed for precise manipulation when deploying the diffusion policies.

B. Visual Representation in Robot Learning

Visual representations need to be comprehensively understood in the robotic systems. Early research primarily utilizes large-scale image datasets (e.g., ImageNet) for pre-training 2D convolutional neural networks (CNNs) [27] or vision [28]. Recent works [29–31] study visual pre-training for robotic manipulation tasks, to improve the generalization of 2D policies through video or language supervision. Meanwhile, 3D representation learning is emerging to enable robot's capability of understanding spatial geometry. Voxel and NeRF can help solve complex tasks with geometric reasoning [6, 32], but they lead to high computational costs. The point cloud based methods [3, 4, 33] are SE(3)-equivariant [34] and accelerate the inference through directly processing the sparse points. To leverage both the geometric and semantic representations, multimodal fusion has been adopted in [9, 13, 35, 36]. Current fusion mechanisms primarily employ naïve concatenation [10] or flow representation [37] for fusing RGB semantics with 3D geometry. Although conditioning mechanism has been employed for image generation [31, 38], the integration of semantics and geometric information needs to be delicately tailored for efficient robotic perception.

C. Motion Guidance and Trajectory Prediction

The appropriate object-centric representation is essential in the robotic manipulation domain [39]. Compared with the end-to-end approaches that map pixels directly to actions, incorporating object state representations significantly enhances policy robustness. Early works [40, 41] have considered object 6D pose estimation for robotic manipulation. To reduce reliance on precise geometric models, the keypoint based methods such as [39, 42] localize semantic points of interest to simplify object state representation; the scene flow based methods such as [37] estimate the object surface normals to guide motion planning. Long-term point tracking technologies have been developed in [43–45], where complex motion of objects are captured through tracking sparse keypoints. These works highlight that compared to pursuing nearly complete pose estimation, learning low-dimensional trajectories or keypoint representations not only helps reduce computational burden, but also delivers adequate state information to guide manipulation policies.

III. Method

The proposed MOP is an object position guided multimodal policy as shown in Fig. 2. Differing from the existing multimodal imitation learning methods, the geometry-guided MOP is capable of avoiding the modality competence in an adaptive multimodal fusion manner. MOP consists of three components: i) Geometry-guided multimodal perception (G2MP) decouples 2D semantics and 3D geometric structure, to provide robust state representations; ii) Object position prediction (OPP) module estimates the object positions and thus provides the geometric priors; iii) Diffusion-based policy generates the precise robotic actions.

A. Geometry-Guided Multimodal Perception (G2MP)

In the unstructured environments, robotic perception needs to comprehensively understand both the object geometry and visual semantics. Therefore, we first adopt unimodal encoders for complementary strengths.

1) Spatial Softmax based Position Encoder: The encoder utilizes ResNet-18 based backbone. Different from the global average pooling that encodes the visual semantics into a 1D feature [13], spatial softmax pooling is adopted to maintain pixel position information. Assuming that each pixel corresponds to a 2D coordinate μ, and the image feature map extracted by the ResNet-18 is $\mathbf{f} \in \mathbb{R}^{C\times H\times W}$ with C feature channels and the spatial dimensions H, W, the keypoint coordinate μ_c for each feature channel c is as follows:

$$\mu_c = \sum_{h=1}^{H}\sum_{w=1}^{W} \mu\left(h,w\right) \frac{e^{f_{c,h,w}/\tau}}{\sum\limits_{i=1}^{H}\sum\limits_{j=1}^{W} e^{f_{c,i,j}/\tau}}, \tag{1}$$

where $f_{c,h,w}$ is the activation for each feature channel c at $\mu\left(h,w\right)$, τ is the temperature. Thus we can transform the image feature into C keypoint coordinates denoted by $\mathbf{F}_v \in \mathbb{R}^{C\times 2}$.

2) Feature Extraction from Point Clouds: Given the observed point clouds $\mathbf{P} \in \mathbb{R}^{N\times 3}$, we use a lightweight DP3 encoder [4], namely the geometric feature $\mathbf{F}_g$ is obtained after a three-layer MLP and max-pooling operations.

3) Gated Geometry-Query Fusion: To address the modality competence issue for multimodal fusion, we propose a geometry-guided and geometry-query gated fusion mechanism. Specifically, the gated fusion feature $\mathbf{Z}_{\text{en}}$ is obtained in a residual manner:

$$\mathbf{Z}_{\text{en}} = \mathbf{F}_g + \alpha\tilde{\mathbf{F}}, \tag{2}$$

where the learnable gate α ensures that geometric feature is dominant, and $\tilde{\mathbf{F}}$ is expressed as:

$$\tilde{\mathbf{F}} = \text{Attention}\left(\mathbf{Q} = \mathbf{W}_{\mathcal{Q}}\mathbf{F}_g, \mathbf{K} = \mathbf{W}_{\mathcal{K}}\mathbf{F}_v, \mathbf{V} = \mathbf{W}_{\mathcal{V}}\mathbf{F}_v\right) \tag{3}$$

in which $\mathbf{Q}$, $\mathbf{K}$ and $\mathbf{V}$ are the query, key and value, respectively, $\mathbf{W}_{\mathcal{Q}}$, $\mathbf{W}_{\mathcal{K}}$ and $\mathbf{W}_{\mathcal{V}}$ are linear projections. During the training, the Modality-Dropout is adopted to ensure that the model keeps geometry-dominated, i.e., with probability p, randomly zero out image features to force the network to learn robust geometric representations. When point cloud is sufficient, the gate α automatically reduces reliance on semantic information. Conversely, the network actively extracts visual semantics to enhance feature expressiveness when the geometric information is insufficient. Finally, the gated fusion feature $\mathbf{Z}_{\text{en}}$ and the encoded robot state are concatenated to form observation representations $\mathcal{O}$.

B. Object Position Prediction (OPP)

In the complex environments, the use of object motion information improves task execution. The work [19] employs a MoCap system for demonstration collection and presents MBA policy consisting of two cascaded diffusion processes for predicting the object poses and generating robot actions, which lead to high acquisition costs and inference delay in practice. As such, we introduce OPP, a lightweight CVAE based module for object position prediction without the assistance of MoCap system. Given observations $\mathcal{O}$, OPP predicts future object position information of T_{obj} steps, namely $\mathcal{P} = \{\mathbf{p}_t\}_{t=1}^{T_{\text{obj}}}$ with $\mathbf{p}_t = (x_t, y_t, z_t)$. Incorporating $\mathcal{P}$ via residual gating yields the conditional inputs of the diffusion policy:

$$\mathcal{I}_{\text{policy}} = \mathcal{O} + \beta \cdot \text{MLP}(\mathcal{P}, \mathcal{O}), \tag{4}$$

where the perceptual feature gate β is learnable parameter. Compared to MBA, our OPP module only predicts the object position and does not rely on object pose labeled data. In the following ablation study (Section IV-E), it is confirmed that our MOP variant with the 3D object position information performs comparably to MBA with object pose information (a 9D-vector).

C. Diffusion-based Policy Execution

In the execution module, the noise-free action sequences $\mathcal{A}_0$ are generated by leveraging a conditional diffusion model. Specifically, the observation features $\mathcal{I}_{\text{policy}}$ is injected into the noise prediction network ε_θ as the condition via Feature-wise Linear Modulation (FiLM), the denoising process at the ℓ-th step is performed as follows:

$$\mathcal{A}_{\ell-1} = \frac{1}{\sqrt{\alpha_\ell}}\left(\mathcal{A}_\ell - \frac{1-\alpha_\ell}{\sqrt{1-\bar{\alpha}_\ell}}\varepsilon_\theta\left(\mathcal{A}_\ell, \mathcal{I}_{\text{policy}}, \ell\right)\right) + \sigma_\ell \mathcal{N}\left(\mathbf{0}, \mathbf{I}\right), \tag{5}$$

where α_ℓ and σ_ℓ are the noise schedule [46], $\bar{\alpha}_\ell = \prod_{s=1}^{\ell}\alpha_s$.

D. Optimization Objective

The proposed MOP framework employs the multi-task end-to-end training, to ensure that modules are coordinated and prevent excessive dependence on uncertainty predictions. Therefore, the total loss function is calculated as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{diff}} + \lambda_{\text{opp}}\mathcal{L}_{\text{opp}} + \lambda_{\text{reg}}\mathcal{L}_{\text{sparse}}, \tag{6}$$

where $\mathcal{L}_{\text{diff}}$ is the diffusion loss, $\mathcal{L}_{\text{opp}}$ is the OPP loss with the weight λ_{opp} and the sparsity loss regularizer $\mathcal{L}_{\text{sparse}}$ with the weight λ_{reg}.

1) Diffusion Loss: To train the noise prediction network ε_θ, we use MSE as the objective:

$$\mathcal{L}_{\text{diff}} = \mathbb{E}\left[\left\|\varepsilon^\ell - \varepsilon_\theta(\mathcal{A}_0 + \varepsilon^\ell, \mathcal{I}_{\text{policy}}, \ell)\right\|_2^2\right]. \tag{7}$$

TABLE II: **Comparing MOP (ours) with the baselines across 56 tasks in 3 simulation environments.** We run 3 seeds and report the average success rate (%) and standard deviation of the results.

Method	Adroit(3 tasks)	DexArt(4 tasks)	Meta-World Easy(28 tasks)	Meta-World Medium(11 tasks)	Meta-World Hard(5 tasks)	Meta-World Very Hard(5 tasks)	Average
DP [2]	25.8±4.0	40.3±2.4	81.2±2.5	52.9±4.0	47.9±5.0	54.3±4.2	64.4±3.3
DP3 [4]	74.9±1.4	51.1±3.3	84.8±3.0	63.9±4.4	53.2±6.0	65.5±5.5	73.2±3.7
Mamba Policy [47]	76.2±3.0	51.4±3.5	84.3±2.6	62.2±4.2	58.1±3.7	63.9±3.8	73.1±3.2
MBA [19]	80.8±2.4	54.4±4.2	85.3±1.7	65.2±4.7	61.3±5.2	67.9±4.4	75.2±3.1
MOP (ours)	81.3±2.3	52.7±3.5	85.6±1.5	78.6±2.6	73.0±3.6	72.1±2.6	79.3±2.2

2) OPP Loss: The CVAE-based OPP loss includes the trajectory reconstruction term and KL divergence term for regularizing the latent space $z \in \mathcal{Z}$:

$$\mathcal{L}_{\text{opp}} = \sum_{t=1}^{T_{\text{obj}}} \left\| \mathbf{p}_t - \mathbf{p}_t^{\text{GT}} \right\|_1 + \beta_{\text{KL}} D_{\text{KL}} \left(q(z|\mathcal{P}^{\text{GT}}, \mathcal{O}) \parallel \mathcal{N}(\mathbf{0}, \mathbf{I}) \right), \tag{8}$$

where $\mathcal{P}^{\text{GT}} = \{\mathbf{p}_t^{\text{GT}}\}$ are the ground-truth object position information, β_{KL} is the weight and $q(z|\mathcal{P}^{\text{GT}}, \mathcal{O})$ is the probabilistic encoder.

3) Sparsity Loss: Since system robustness can be further enhanced by introducing a dual residual learning mechanism, we adopt the sparsity loss to regularize the geometric feature gate α and the perceptual feature gate β:

$$\mathcal{L}_{\text{sparse}} = \|\alpha\|_2^2 + \|\beta\|_2^2 \,. \tag{9}$$

IV. Simulation and Real-World Experiments

A. Simulation Experiment Setup

Simulation Benchmarks. We consider the total 56 manipulation tasks in 3 simulation environments for performance evaluation:

- Adroit [20]: A 28-DoF dexterous arm-hand manipulator solves the tasks in the MuJoCo physics simulator [48]. We select 3 challenging tasks including door opening, hammer and repositioning a pen.
- DexArt [7]: A 22-DoF dexterous arm-hand manipulator solves the tasks in the SAPIEN physical simulator [49]. Following [7], we consider 4 tasks including Faucet, Bucket, Laptop and Toilet. These tasks focus on the interactions with articulated objects.
- Meta-World [21]: A 7-DoF Sawyer arm with the gripper solves the tasks in the MuJoCo physics simulator. We consider 49 tasks, to evaluate broad task generalization.

Evaluation Metric. We run 3 seeds for each experiment with seed number 0, 42, 66, and evaluate 20 episodes to calculate the success rate at each seed. The average success rate of the top five checkpoints is reported. The settings for the 3 simulation benchmarks are detailed as follows:

- Adroit: We use 10 expert demonstrations per task. Each model is trained for 3000 epochs and evaluated every 200 epochs.
- DexArt: We use 50 expert demonstrations per task. Each model is trained for 3000 epochs and evaluated every 200 epochs.
- Meta-World: We use 10 expert demonstrations per task. Each model is trained for 1000 epochs and evaluated every 100 epochs.

Baselines. To comprehensively exhibit the performance behavior of the proposed MOP method, we compare it with four strong diffusion-based imitation learning policies. These baselines cover different visual observations, model architectures and inference strategies:

- DP [2]: An image-based imitation learning method utilizes the CNN and transformer as the backbones.
- DP3 [4]: A point cloud based imitation learning method utilizes a light-weight Pointnet to obtain the 3D representations.
- Mamba Policy [47]: A state space model (SSM) based 3D policy introduces Mamba architecture as the denoising network backbone.
- MBA [19]: An object-centric imitation learning method adopts two cascaded diffusion processes to predict the object motion and generate object motion guided actions with the help of a MoCap system.

Simulation settings. We use a single NVIDIA RTX 4090 GPU to train all the models. The detailed hyperparameters for all the baselines and our MOP method are concluded in Table I. Since only the baseline MBA and our MOP also cover the object dynamics, the object motion/position prediction horizon is set to be the same value for the sake of fairness in these two policies.

TABLE I: Hyperparameters for all the baselines and our MOP method.

Hyperparameters	Value	Method
Observation horizon	2	All
Action prediction horizon	16	All
Execution horizon	8	All
Motion/position prediction horizon	16	MBA /**MOP (ours)**
Visual encoder output dimension	128	All 3D policies
Diffusion steps (training/inference)	100/10	All
Batch size	128	All
Learning rate	1×10^{-4}	All

B. Simulation Results

Table II shows clear differences that our MOP achieves higher average success rate (%) with lower standard deviation across most evaluated tasks in 3 environments than the baselines, which confirms the strong robustness and generalization of our approach. In particular, MOP improves the average success rates by at least 10% in the Meta-World tasks with the medium and hard levels of difficulty compare to MBA, and reaches the best performance in Adroit tasks.

To evaluate the learning efficiency of the proposed model, we sample 8 simulation tasks with different levels of difficulty from Meta-World benchmark. Fig. 3 confirms that MOP

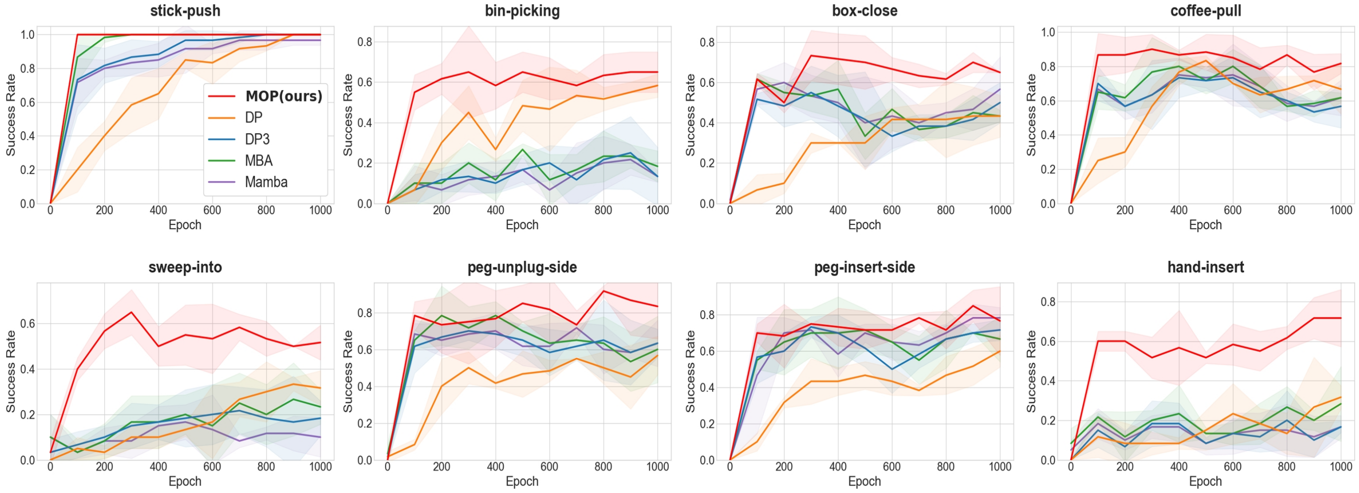

Fig. 3: **Learning curves of MOP and the baselines on Meta-World benchmark.** MOP achieves faster convergence and higher success rates than the baselines.

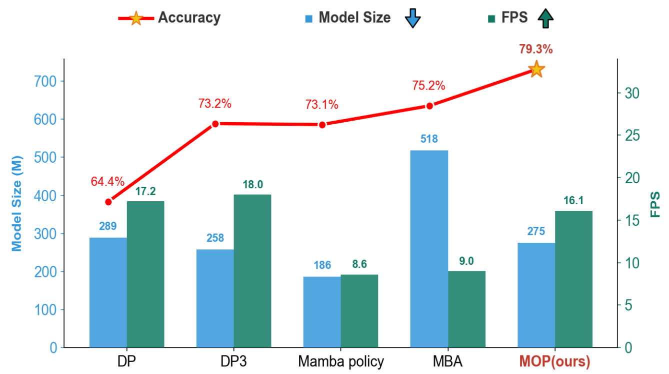

Fig. 4: **Effects of different models on the accuracy and inference speed (FPS) via a single NVIDIA RTX 4090 GPU. MOP (ours)** achieves the highest accuracy with reasonable inference speed.

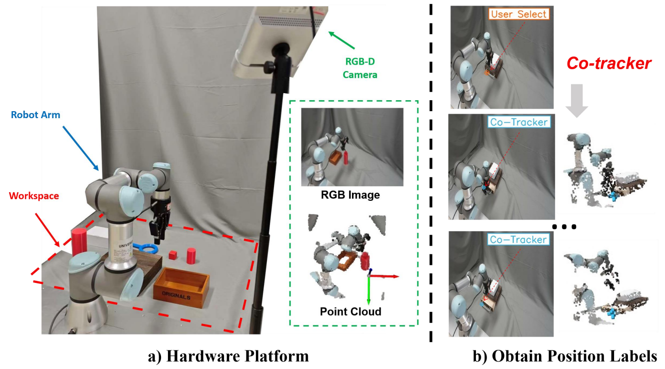

Fig. 5: **Real-world robot setup.** 6-DoF UR3 arm with a Robotiq 2F-85 gripper and Orbbec Femto Mega RGB-D camera to capture observations during the execution process; Cotracker [45] is employed to obtain object position labels.

converges rapidly through leveraging the proposed G2MP and OPP modules, and outperforms the baselines across these tasks. In **sweep-into** and **hand-insert** tasks, unlike existing methods that suffer from low efficiency and limited accuracy, MOP significantly improves accuracy due to its capability of representations learning and efficient guidance for action generation. In **bin-picking** task with small object, point cloud downsampling in existing 3D policies [4, 19, 47] leads to the geometric information loss and thus reduces accuracy; our MOP fuses 2D semantics with 3D features and can efficiently capture key spatial information in the complex scene.

In real-world environments, the deployment of robot learning requires lower inference latency since the closed-loop control may be frequent. While accuracy (average success rate) is pivotal, high computational efficiency is also preferred in practice. Fig. 4 shows that MOP achieves the highest accuracy (79.3%) by using 275M parameters, significantly less than MBA (518M parameters), and offers 1.8x inference speed compared to MBA. Meanwhile, with slightly less inference speed, MOP improves the accuracy by 6.1% over DP3 (258M parameters) and 14.9% over DP (289M parameters). Although Mamba Policy has the lowest number of model parameters (186M), its inference is the slowest (only 8.6 FPS) and accuracy (73.1%) is also decreased.

C. Real-world Experiments Setup

To evaluate MOP's accuracy and generalization in practice, we conduct the real-world experiments across 4 manipulation tasks.

Platform. The real robot system consists of a 6-DoF UR3 arm with a Robotiq 2F-85 gripper and Orbbec Femto Mega RGB-D camera to capture observations, as shown in Fig. 5a). To train the OPP module, the object position labels are collected with the help of Cotracker (Fig. 5b)) [45]. Training and inference are performed by using a single NVIDIA RTX 4090 GPU and a single RTX 4060 GPU, respectively.

Baselines and Tasks. We compare MOP with 3D policies

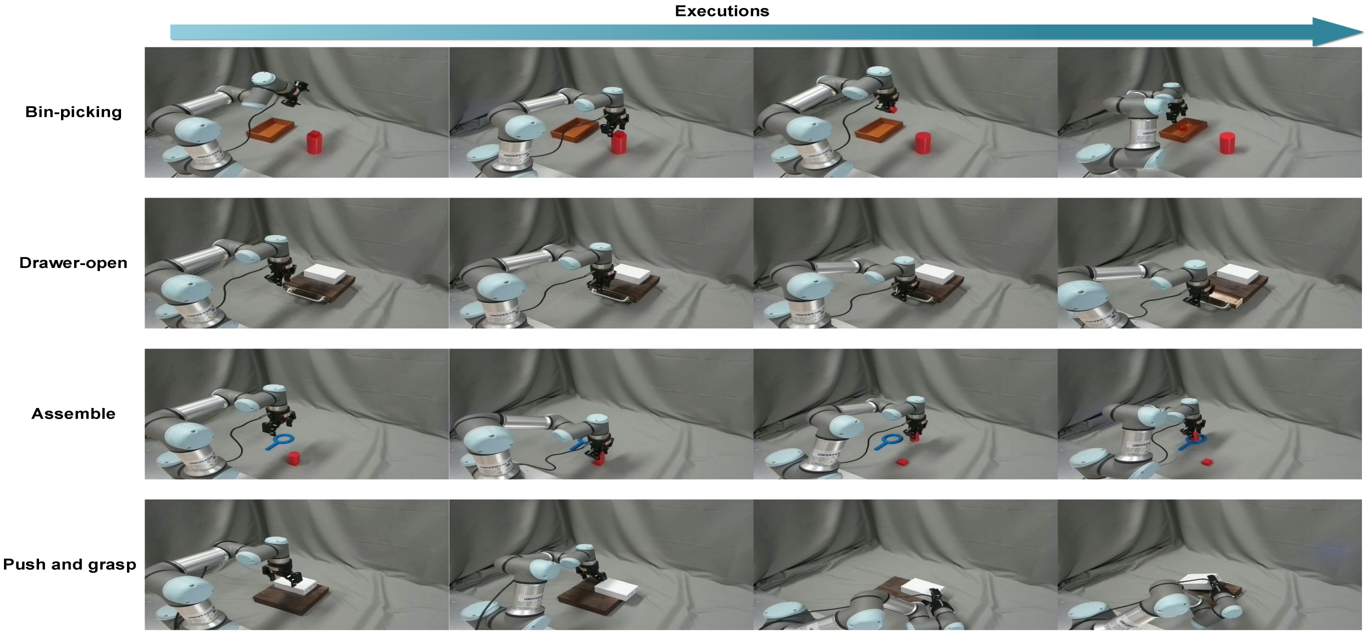

Fig. 6: Examples of execution steps across 4 real-world manipulation tasks.

TABLE III: Real-world performance comparisons between **MOP** and the baselines across 4 manipulation tasks.

Method	Bin-picking	Drawer-open		Assemble		Push-grasp	
	Succ. (%) ↑	Collision ↓	Succ. (%) ↑	Grasp Succ. (%) ↑	Succ. (%) ↑	Stage I Succ. (%) ↑	Succ. (%) ↑
DP3 [4]	20	11	30	70	30	40	30
Mamba Policy [47]	15	11	35	75	25	40	35
MOP (ours)	40	9	55	75	50	65	60

including DP3 [4] and Mamba policy [47][1]. In Fig. 6, we describe 4 real-world tasks as follows:

- **Bin-picking** (Small object): Pick a small cubic object and place it into a bin. In this task, point cloud downsampling often leads to the loss of small object representation.
- **Drawer-open** (Articulated object): Open a drawer under random drawer positions. This task involves interactions with articulated object. Collision between the gripper and drawer needs to be reduced.
- **Assemble** (High-precision task): Grasp a cylindrical object with a diameter of 40mm and move it to the working space, and insert it into a circular ring with an inner diameter of 60mm.
- **Push-grasp** (multi-stage task): It consists of two stages: i) In Stage I, push a cuboid object to the edge until it overhangs; and ii) In the final stage, grasp it.

We collect 50 expert demonstrations via teleoperation, and conduct 20 independent trials for each task.

D. Real-world Results

We evaluate MOP against DP3 and Mamba Policy on 4 complex tasks. Table III shows that MOP significantly improves the success rate across all the tasks, compared to the baselines. In **Bin-picking** task, MOP achieves a success rate twice that of DP3; In **Drawer-open** task, MOP improves the success rate by 20% over Mamba Policy. In **Assemble** task, our MOP enables highly precise manipulation, hence it improve the success rate by 20% over DP3. In addition, leveraging MOP increases the **Push-grasp** accuracy by at least 25% over Mamba Policy.

E. Ablation Study

To analyze the effects of key modules in MOP, we conduct ablation experiments on 4 tasks in Meta-World environment, namely **coffee-pull (c-p)**, **sweep-into (s-i)**, **peg-unplug-side (p-u-s)** and **hand-insert (h-i)**. All the experiments follow the same simulation setting: All the models are trained across 3 seeds; each model is trained for 1000 epochs, and 20 episodes are tested every 100 epochs. Besides the baseline methods DP3 and MBA, we create 3 MOP variants:

- MOP w/o OPP: In this variant, we exclude the OPP module while retaining the G2MP module. This design evaluates the policy's performance without explicit object position and trajectory priors.
- MOP w/o RGB: In this variant, we remove the image encoder and the G2MP fusion module, reducing the policy to a unimodal point cloud baseline. This setting allows us to evaluate the differences between our lightweight OPP module and MBA with 9D object pose information.
- MOP w/ Concat: In this variant, we replace the adaptive gating module with straightforward concatenation. Here, semantic and geometric features are simply concatenated, treating both modalities as equally informative.

[1] As shown in Tables II and III, our MOP outperforms MBA [19] in Adroit and Meta-World benchmarks. Since MBA requires high-dimensional object motion data supported by MoCap system, its real-world performance is not evaluated in our low-cost platform.

TABLE IV: Ablation study on key modules in Meta-World environment.

Method	c-p	s-i	p-u-s	h-i	Average
MOP (ours)	91.0±4.6	60.7±1.5	90.3±4.2	67.7±5.1	77.4
MOP w/o OPP	87.0±3.3	57.7±2.4	82.7±0.9	61.0±6.2	72.1
MOP w/o RGB	75.0±2.5	24.7±2.9	77.7±2.5	23.0±2.9	50.1
MOP w/ Concat	80.7±2.4	61.3±1.3	59.3±1.3	61.7±3.3	65.8
DP3 [4]	73.0±6.6	22.3±8.4	72.0±2.0	19.7±6.1	46.8
MBA [19]	76.3±5.1	23.7±10.8	76.0±4.4	26.3±6.7	50.6

Our ablation study in Table IV shows that overall, MOP can comprehensively exploit multimodal benefits and thus achieve the highest accuracy, compared to its variants and baselines. MOP w/o RGB performs similarly to MBA, which demonstrates that our lightweight OPP module can efficiently replace the expensive 9D object pose supervision approach and own fast inference speed. In addition, MOP improves the accuracy by 11.6% over its variant MOP w/ Concat, which confirms that the adaptive gated fusion mechanism introduced by MOP indeed addresses the modality competence.

F. Analysis of Adaptive Gating Mechanism

To further investigate the adaptive capabilities of the G2MP module across tasks with varying geometric features, we conduct an in-depth analysis of two representative tasks from the Meta-World environment, namely **Window-close** (involving large objects with dense point cloud features) and **Bin-picking** (involving small objects with sparse point cloud features).

Fig. 7 reveals a significant task-dependent pattern in the gating parameter α (representing the fusion weight of 2D image features). In **Window-close** task, the target object (a drawer) possesses large, continuous 3D surfaces. Geometry-only methods like DP3 can already capture near-perfect feature representations through the point cloud encoder (achieving a 100% success rate). In this case, the G2MP module of MOP automatically recognizes the sufficiency of geometric information, maintaining α at an extremely low level (0.03). This effectively explains MOP's ability to match the accuracy of point cloud-only methods in tasks with rich geometric features. In contrast, during the **Bin-picking** task, extremely sparse point cloud samples of the small object are obtained after point cloud downsampling. This sparsity makes it challenging for the point cloud encoder to capture adequate object features. Consequently, the geometry-only DP3 method exhibits significant performance degradation (success rate drops to 22%). However, our MOP approach via the G2MP module identifies the inadequate 3D geometric representation, and adaptively increases the gating weight α to 0.2, thereby introducing more 2D semantic features extracted by the image encoder. This adaptive semantic compensation substantially enhances performance, boosting the success rate on Bin-picking to 69% (significantly surpassing DP3). The result demonstrates that the G2MP module dynamically tunes the fusion weights of multimodal features based on the assessed reliability of the geometric information. Therefore, MOP enables robust manipulation across tasks of varying geometric complexity.

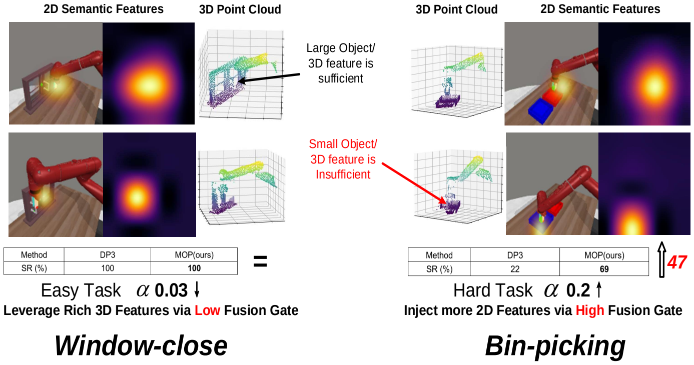

Fig. 7: **Analysis of adaptive gating mechanism.** The proposed G2MP module adaptively and efficiently fuses the 2D semantic features and geometric information, enabling significant success rate (SR) improvement in manipulation tasks involving small objects.

V. Conclusions

In this work, we presented MOP, a novel multimodal object-aware policy designed for efficient and robust robotic manipulation. Addressing the challenge that 3D representations often suffer from information loss when handling small objects, MOP incorporates a geometry-guided multimodal perception module. This module leverages an adaptive gating mechanism to dynamically fuse 2D and 3D features. Furthermore, we introduced a lightweight object position prediction module to inject rich trajectory priors into the policy without relying on external motion capture systems. Extensive experiments across 56 tasks in three simulation environments and 4 representative real-world tasks demonstrate that MOP outperforms strong baselines, exhibiting exceptional capability in handling challenging manipulation tasks.

In future work, the real-world performance boundaries of MOP will be further exploited across more diverse robots, dexterous hands, and tasks. Moreover, the line of this work will be extended to incorporate more control frameworks including combining imitation learning with reinforcement learning for enhancing generalization.

References

[1] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, "Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware," in *Proc. Robot.: Sci. Syst.*, 2023.

[2] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," *Int. J. Rob. Res.*, vol. 44, no. 10–11, pp. 1684–1704, Sep. 2025.

[3] C. Wang, H. Fang, H.-S. Fang, and C. Lu, "RISE: 3D perception makes real-world robot imitation simple and effective," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2024, pp. 2870–2877.

[4] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu, "3D diffusion policy: Generalizable visuomotor policy learning via simple 3D representations," in *Proc. Robot.: Sci. Syst.*, 2024.

[5] T.-W. Ke, N. Gkanatsios, and K. Fragkiadaki, "3D diffuser actor: Policy diffusion with 3D scene representations," in *Proc. Conf. Robot Learn.*, vol. 270, 2025, pp. 1949–1974.

[6] M. Shridhar, L. Manuelli, and D. Fox, "Perceiver-actor: A multi-task transformer for robotic manipulation," in *Proc. Conf. Robot Learn. (CoRL)*, vol. 205, Dec 2023, pp. 785–799.

[7] C. Bao, H. Xu, Y. Qin, and X. Wang, “DexArt: Benchmarking generalizable dexterous manipulation with articulated objects,” in *IEEE/CVF Conf. Computer Vis. Pattern Recognit. (CVPR)*, 2023, pp. 21 190–21 200.

[8] G. Yan, Y.-H. Wu, and X. Wang, “DNAct: Diffusion guided multi-task 3D policy learning,” in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2025, pp. 9464–9471.

[9] B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid *et al.*, “RT-2: Vision-language-action models transfer web knowledge to robotic control,” in *Conf. Robot Learn.*, 2023, pp. 2165–2183.

[10] A. O’Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain *et al.*, “Open x-embodiment: Robotic learning datasets and RT-X models: Open x-embodiment collaboration^{0},” in *IEEE ICRA*, 2024, pp. 6892–6903.

[11] C. Hao, K. Lin, Z. Xue, S. Luo, and H. Soh, “Disco: Language-guided manipulation with diffusion policies and constrained inpainting,” *IEEE Robot. Autom. Lett.*, vol. 10, no. 10, pp. 9726–9733, 2025.

[12] H. Ha, P. Florence, and S. Song, “Scaling up and distilling down: Language-guided robot skill acquisition,” in *Conf. Robot Learn.*, 2023, pp. 3766–3777.

[13] Y. Tang, H. Geng, S. Zang, P. Abbeel, and J. Malik, “Visual-geometry diffusion policy: Robust generalization via complementarity-aware multimodal fusion,” *arXiv preprint arXiv:2511.22445*, 2025.

[14] H. Fang, C. Wang, Y. Wang, J. Chen, S. Xia, J. Lv, Z. He, X. Yi, Y. Guo, X. Zhan *et al.*, “AirExo-2: Scaling up generalizable robotic imitation learning with low-cost exoskeletons,” in *CoRL*, 2025.

[15] Z. Xue and R. Marculescu, “Dynamic multimodal fusion,” in *CVPRW*, 2023, pp. 2575–2584.

[16] S. Wu, Y. Zhu, Y. Huang, K. Zhu, J. Gu, J. Yu, Y. Shi, and J. Wang, “AffordDP: Generalizable diffusion policy with transferable affordance,” in *CVPR*, 2025, pp. 6971–6980.

[17] C. Wen, X. Lin, J. I. R. So, K. Chen, Q. Dou, Y. Gao, and P. Abbeel, “Any-point Trajectory Modeling for Policy Learning,” in *Proc. Robot.: Sci. Syst.*, July 2024.

[18] S. Noh, D. Nam, K. Kim, G. Lee, Y. Yu, R. Kang, and K. Lee, “3D flow diffusion policy: Visuomotor policy learning via generating flow in 3D space,” *arXiv preprint arXiv:2509.18676*, 2025.

[19] Y. Su, X. Zhan, H. Fang, Y.-L. Li, C. Lu, and L. Yang, “Motion before action: Diffusing object motion as manipulation condition,” *IEEE Robot. Autom. Lett.*, vol. 10, no. 7, pp. 7428–7435, 2025.

[20] A. Rajeswaran, V. Kumar, A. Gupta, G. Vezzani, J. Schulman, E. Todorov, and S. Levine, “Learning complex dexterous manipulation with deep reinforcement learning and demonstrations,” in *Proc. Robot.: Sci. Syst.*, 2018.

[21] T. Yu, D. Quillen, Z. He, R. Julian, K. Hausman, C. Finn, and S. Levine, “Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning,” in *Conf. robot learn. (CoRL)*, 2020, pp. 1094–1100.

[22] D. A. Pomerleau, “Alvinn: An autonomous land vehicle in a neural network,” in *NIPS*, 1988.

[23] S. Ross, G. Gordon, and D. Bagnell, “A reduction of imitation learning and structured prediction to no-regret online learning,” in *PMLR*, 2011, pp. 627–635.

[24] P. Florence, C. Lynch, A. Zeng, O. A. Ramirez, A. Wahid, L. Downs, A. Wong, J. Lee, I. Mordatch, and J. Tompson, “Implicit behavioral cloning,” in *Conf. robot learn.*, 2022, pp. 158–168.

[25] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” in *CVPR*, 2022, pp. 10 684–10 695.

[26] M. Reuss, M. Li, X. Jia, and R. Lioutikov, “Goal-conditioned imitation learning using score-based diffusion policies,” in *Robot.: Sci. Syst.*, 2023.

[27] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *CVPR*, 2016, pp. 770–778.

[28] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, “An image is worth 16x16 words: Transformers for image recognition at scale,” in *ICLR*, 2021.

[29] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, “R3M: A universal visual representation for robot manipulation,” in *Conf. Robot Learn.*, 2022.

[30] Y. J. Ma, S. Sodhani, D. Jayaraman, O. Bastani, V. Kumar, and A. Zhang, “VIP: Towards universal visual reward and representation via value-implicit pre-training,” in *ICLR*, 2023.

[31] W. Peebles and S. Xie, “Scalable diffusion models with transformers,” in *CVPR*, 2023, pp. 4195–4205.

[32] Y. Ze, G. Yan, Y.-H. Wu, A. Macaluso, Y. Ge, J. Ye, N. Hansen, L. E. Li, and X. Wang, “GNFactor: Multi-task real robot learning with generalizable neural feature fields,” in *Conf. Robot Learn.*, 2023, pp. 284–301.

[33] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, “Pointnet++: Deep hierarchical feature learning on point sets in a metric space,” *NIPS*, vol. 30, 2017.

[34] A. Simeonov, Y. Du, A. Tagliasacchi, J. B. Tenenbaum, A. Rodriguez, P. Agrawal, and V. Sitzmann, “Neural descriptor fields: Se(3)-equivariant object representations for manipulation,” in *ICRA*, 2022, pp. 6394–6400.

[35] Octo Model Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, C. Xu, J. Luo, T. Kreiman, Y. Tan, L. Y. Chen, P. Sanketi, Q. Vuong, T. Xiao, D. Sadigh, C. Finn, and S. Levine, “Octo: An open-source generalist robot policy,” in *Proc. Robot.: Sci. Syst.*, 2024.

[36] X. Zhang, Y. Jiang, H. Qing, and J. Bai, “Language-conditioned representations and mixture-of-experts policy for robust multi-task robotic manipulation,” *arXiv preprint arXiv:2510.24055*, 2025.

[37] B. Eisner, H. Zhang, and D. Held, “FlowBot3D: Learning 3D articulation flow to manipulate articulated objects,” in *Proc. Robot.: Sci. Syst.*, 2022.

[38] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville, “Film: Visual reasoning with a general conditioning layer,” in *AAAI*, vol. 32, no. 1, 2018.

[39] P. R. Florence, L. Manuelli, and R. Tedrake, “Dense object nets: Learning dense visual object descriptors by and for robotic manipulation,” in *CoRL*, 2018.

[40] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, “Posecnn: A convolutional neural network for 6D object pose estimation in cluttered scenes,” in *RSS*, 2018.

[41] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, and S. Savarese, “DenseFusion: 6D object pose estimation by iterative dense fusion,” in *CVPR*, 2019, pp. 3343–3352.

[42] W. Gao and R. Tedrake, “kPAM 2.0: Feedback control for category-level robotic manipulation,” *IEEE Robot. Automat. Lett.*, vol. 6, no. 2, pp. 2962–2969, 2021.

[43] C. Doersch, Y. Yang, M. Vecerik, D. Gokay, A. Gupta, Y. Aytar, J. Carreira, and A. Zisserman, “Tapir: Tracking any point with per-frame initialization and temporal refinement,” in *ICCV*, 2023, pp. 10 061–10 072.

[44] H. Bharadhwaj, R. Mottaghi, A. Gupta, and S. Tulsiani, “Track2act: Predicting point tracks from internet videos enables diverse zero-shot robot manipulation,” in *ECCV*, 2024.

[45] N. Karaev, I. Rocco, B. Graham, N. Neverova, A. Vedaldi, and C. Rupprecht, “Cotracker: It is better to track together,” in *ECCV*, 2024, pp. 18–35.

[46] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” *NeurIPS*, vol. 33, pp. 6840–6851, 2020.

[47] J. Cao, Q. Zhang, J. Sun, J. Wang, H. Cheng, Y. Li, J. Ma, K. Wu, Z. Xu, Y. Shao *et al.*, “Mamba policy: Towards efficient 3D diffusion policy with hybrid selective state models,” in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2025, pp. 11 359–11 366.

[48] E. Todorov, T. Erez, and Y. Tassa, “MuJoCo: A physics engine for model-based control,” in *IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2012, pp. 5026–5033.

[49] F. Xiang, Y. Qin, K. Mo, Y. Xia, H. Zhu, F. Liu, M. Liu, H. Jiang, Y. Yuan, H. Wang *et al.*, “Sapien: A simulated part-based interactive environment,” in *IEEE/CVF Conf. Computer Vis. Pattern Recognit. (CVPR)*, 2020, pp. 11 097–11 107.