

# Curvature-Guided Flow Matching via SE(3)-Canonical Point Clouds for Robot Imitation Learning

**Abstract**—Imitation learning has emerged as a pivotal paradigm for embodied agents to acquire complex manipulation skills. However, in real-world environments characterized by dynamic occlusions and viewpoint variations, existing 3D point cloud-based policies often suffer from instability and poor generalization, primarily due to the neglect of temporal geometric continuity. To address these challenges, this paper proposes a Canonical Curve Flow Policy for robust robot manipulation. Specifically, we introduce a temporal-aware canonical point cloud representation framework that incorporates SE(3)-equivariance into temporal modeling. By synergistically aggregating cross-frame geometric features, this framework significantly enhances perceptual robustness against unstructured disturbances. Furthermore, to overcome the limitations of linear trajectory assumptions in conventional rectified flow methods, we design a curvature-guided flow matching strategy. This strategy facilitates the learning of probability-aware nonlinear trajectories, enabling the generation process to effectively circumvent low-likelihood regions of the data manifold for precise motion control. Extensive experiments on Robomimic and RLBench benchmarks, as well as real-world robot tasks, demonstrate that the proposed method outperforms state-of-the-art baselines in both convergence efficiency and task success rates. The code and further details are available on the website: <https://curveflow-policy.github.io/>.

**Index Terms**—imitation learning, flow matching, equivariance

## I. INTRODUCTION

ROBOT manipulation has witnessed significant progress through imitation learning, driven by the rapid emergence of generative models [1]–[3]. Visual imitation policies [4], [5], in particular, have demonstrated the capability to learn complex, multi-modal action distributions from expert demonstrations. However, the fundamental bottleneck lies in generalization. While these policies perform well in controlled settings, their performance often degrades substantially when confronted with objects, geometric configurations, or camera viewpoints that lie outside the training distribution. This vulnerability arises because standard data-driven models often act as black-box approximators that rely on memorizing coordinate-dependent patterns or visual textures. Lacking explicit inductive biases for 3D geometry, they struggle to decouple the manipulation logic from global pose variations, leading to brittle extrapolation in unseen scenarios.

To address the generalization bottleneck, recent research has pursued two parallel directions: enhancing 3D geometric

perception and optimizing motion generation. Nevertheless, a critical structural disconnect remains between them.

On the perception side, efforts have focused on extracting geometrically invariant features from high-dimensional observations [6]–[8] to derive effective control policies. Specifically, methods leveraging 3D point clouds [9]–[11] have offered inherent invariance to object appearance and viewpoint shifts, significantly bolstering generalization in unseen scenarios. However, most of these approaches process point clouds in a frame-by-frame manner. This neglect of spatio-temporal continuity renders the policy fragile to sensor noise and dynamic occlusions, as the model fails to maintain a consistent geometric understanding over the temporal dimension.

Simultaneously, on the generation side, diffusion-based methods [9], [12]–[14] have emerged as a powerful paradigm for capturing the inherent multi-modal action distributions in robotic manipulation. However, their computationally expensive iterative sampling limits real-time responsiveness. Consequently, Flow Matching (FM) [15], [16] has gained prominence, achieving breakthrough performance in high-fidelity tasks such as image synthesis and scientific modeling. Technically, FM simplifies the generative process by regressing a velocity field to define an Ordinary Differential Equation (ODE), enabling the efficient transport of probability flow from noise to data. Despite this efficiency, prevailing techniques like Rectified Flow [17], [18] rely on a simplified “linear trajectory assumption.” While fast, this linearity ignores the complex, non-linear geometry of the action manifold. A straight path in the probability space may inadvertently force the generative process (flow trajectory) to traverse low-likelihood regions—areas representing physically implausible or unstable actions. Essentially, both limitations stem from a failure to explicitly model the intrinsic structure of the task—whether it is the SE(3) symmetry in perception or the non-linear nature of the probability transport path.

In this work, we propose CurveFlow, a novel framework that bridges this gap by enforcing structural consistency in both representation and generation. First, to resolve perceptual instability, we introduce a spatio-temporal canonical representation. By integrating an SE(3)-equivariant architecture with a temporal attention mechanism, our method aggregates cross-frame information to construct a stable, appearance-invariant geometric description, effectively filtering out observational disturbances. Second, to generate high-fidelity motions, we propose a curvature-guided flow matching strategy. Unlike standard linear flow methods, our approach explicitly models

the curvature of the flow trajectory. This allows the policy to learn non-linear, probability-aware paths that respect the underlying geometry of the action space, thereby avoiding low-likelihood regions while retaining high inference speed. The main contributions of this work are summarized as follows:

- We introduce a temporal-aware canonical point cloud representation that synergizes SE(3)-equivariance with temporal modeling, significantly enhancing robustness against self-occlusion and viewpoint shifts.
- We present the first integration of curvature-guided flow matching into robot manipulation. By parameterizing the probability transport paths with learnable curvature coefficients, we enable the learning of manifold-aware non-linear trajectories that effectively circumvent low-likelihood regions, thereby outperforming standard linear flow baselines.
- Extensive experiments on 16 simulated tasks (Robomimic and RLBench) and real-world robot platforms demonstrate that CurveFlow achieves superior learning efficiency and success rates compared to state-of-the-art methods.

## II. RELATED WORK

### A. Robot Manipulation Learning from Point Clouds

While reinforcement learning [19] provides a pathway for self-supervised skill acquisition, imitation learning has garnered widespread adoption due to its high sample efficiency and stability in acquiring complex skills from expert demonstrations. Following the successful proposal of Diffusion Policy [4], visual imitation policies based on 2D images have become a dominant paradigm due to the accessibility of rgb data. However, these methods often struggle with depth ambiguity in complex spatial tasks. To address this, 3D point clouds [20], as a rich geometric representation, have been extensively applied to manipulation tasks—including closed-loop policy learning [13], [21], [22] and keypoint-based methods [23]—as they mitigate the limitations of 2D images while avoiding the high computational cost associated with voxel inputs [24].

Specifically, Match Policy [25] frames action reasoning as a point cloud registration problem, thereby achieving high-precision pick-and-place operations for unknown configurations. FP3 [26] achieves strong generalization capabilities through large-scale pre-training on diverse point cloud datasets.

However, these methods predominantly process point clouds as isolated frames, lacking explicit temporal aggregation. This limitation compromises their robustness against sensor noise and dynamic occlusions, a deficiency we address via our spatio-temporally aware canonical representation.

### B. Equivariant Policy Learning for Manipulation

Early works incorporating equivariance primarily focused on leveraging symmetries or equivariances [27]–[29] in 3D Euclidean space to enhance spatial generalization in various robot manipulation tasks. For instance, Ben Eisner *et al.* [30] proposed an SE(3)-equivariant method for relative

pose prediction to achieve precise object placement. Building upon the Diffusion Policy framework, EquiBot [31] integrated a diffusion model with action chunking and incorporated SIM(3)-equivariance into the policy learning architecture using the Vector Neuron [32] framework.

ET-SEED [33] introduced an efficient trajectory-level SE(3)-equivariant diffusion model capable of generating action sequences for complex manipulation tasks. In this work, we investigate the geometric symmetry of policies utilizing point cloud inputs. Specifically, we propose learning a canonical representation that explicitly maps raw 3D observations into a unified, standardized coordinate frame to ensure geometric consistency across varying poses.

### C. Flow-Based Robot Motion Generation

Flow matching [3] is an emerging family of generative models rooted in optimal transport theory. It has been successfully applied to modeling robot action sequences [20], [34], [35]. Its core idea involves estimating a velocity vector field that defines an ODE to generate a probability density path, rather than directly matching a trajectory from noise to the target distribution. By circumventing explicit noise estimation, FM enables faster inference, which is crucial for real-time robot manipulation.

A prominent instantiation, Rectified Flow, was introduced by Liu *et al.* [17] and characterizes the forward process as a linear trajectory. Building on this, AdaFlow [36] employs flow-based generative modeling to represent policies via a state-conditioned ODE. Streaming Flow Policy [37] incrementally integrates the learned velocity field to produce action sequences, effectively reducing distribution shift during modeling. M3P [38] couples 3D point cloud inputs with a MeanFlow paradigm to generate action trajectories in a single network function evaluation.

In contrast to methods that rely on the rigid linear trajectory assumption, our work introduces a curvature-guided framework. By parameterizing the transport path with flexible, learnable coefficients, our approach constructs non-linear trajectories that adaptively align with the intrinsic geometry of the action manifold. This capability allows the model to capture complex kinematic behaviors with higher fidelity, offering a more expressive alternative to standard linear interpolation techniques.

## III. PRELIMINARIES

### A. Problem Formulation

Our objective is to learn an effective mapping from the agent’s observational information and proprioceptive state to action trajectories given limited demonstration data. Consider a demonstration dataset  $D = \{\tau_n\}_{n=1}^N$  consisting of  $N$  expert trajectories, where each trajectory  $\tau_n$  contains a sequence of observation-action pairs. The observations  $o$  and actions  $a$  span multiple timesteps, specifically defined as:

$$\tau_n = \{o_t, a_t\}_{t=1}^T \quad (1)$$

where  $o_t$  represents the observation at time  $t$ , and  $a_t$  represents the action at time  $t$ . Our goal is to learn a policy  $\pi$  that

maps from the past observation window  $O$  to the future action window  $A$ :

$$\pi : O \rightarrow A \quad (2)$$

where  $O = \{o_{t-m+1}, \dots, o_t\}$  consists of  $m$  historical observation steps, and  $A = \{a_{t+1}, \dots, a_{t+n}\}$  contains  $n$  future action steps. In our approach, the observation  $o_t$  is defined to encompass both exteroceptive visual inputs (such as images or point clouds) and proprioceptive feedback, including gripper pose or joint angles.

### B. Equivariance in Policy Learning

Generally, a mapping  $f : X \rightarrow Y$  from an input space  $X$  to an output space  $Y$  is said to be equivariant with respect to a group  $G$  if there exist group representations  $\rho_X$  and  $\rho_Y$  of  $G$  on  $X$  and  $Y$ , respectively, such that:

$$f(\rho_X(g) \cdot x) = \rho_Y(g) \cdot f(x) \quad (3)$$

for all  $g \in G$  and  $x \in X$ . When considering the special Euclidean group  $SE(3)$ , whose elements can be represented as  $4 \times 4$  transformation matrices, a mapping  $f$  is said to be  $SE(3)$ -equivariant if it satisfies the following condition:

$$f(T \cdot o) = T \cdot f(o) \quad \forall T \in SE(3) \quad (4)$$

In the domain of policy acquisition, the function  $\pi$  serves to translate a history of observations  $O$  into a sequence of forthcoming actions  $A$ . Crucially, should a spatial transformation  $T$  be applied to any given observation  $o \in O$ , the policy responds by producing an action sequence that is transformed equivalently:

$$\pi(T \cdot o) = T \cdot \pi(o) \quad (5)$$

Such equivariance ensures that the optimal action remains reconstructible via the corresponding inverse mapping, thereby granting the policy strong generalization capabilities against diverse geometric perturbations, including variations in viewpoint and configuration.

### C. Conditional Flow Matching

Conditional Flow Matching(CFM) is a flow-based method designed to transport data between two probability distributions,  $\pi_0$  and  $\pi_1$ , using an ODE framework. CFM learns a time-dependent ODE model that transforms a sample  $\mathbf{x}_0$  from the initial distribution  $\pi_0$  to a target sample  $\mathbf{x}_1$  from distribution  $\pi_1$ . The forward process is defined by a linear interpolation path:

$$\mathbf{x}_t = (1 - t)\mathbf{x}_0 + t\mathbf{x}_1, \quad t \in [0, 1] \quad (6)$$

Compared to the trajectories in diffusion models, this straight-line path simplifies the transformation. The Flow Matching training objective is achieved by learning a corresponding velocity field  $\mathbf{v}_\Theta(\mathbf{x}_t, t)$  to match the time derivative of this linear trajectory:

$$\mathcal{L}_{FM}(\Theta) = \mathbb{E}_{t, \pi_0(\mathbf{x}_0), \pi_1(\mathbf{x}_1)} \left[ \|\mathbf{v}_\Theta(\mathbf{x}_t, t) - (\mathbf{x}_1 - \mathbf{x}_0)\|_2^2 \right] \quad (7)$$

where the target velocity field  $u_t(\mathbf{x}_t | \mathbf{x}_0, \mathbf{x}_1) = \mathbf{x}_1 - \mathbf{x}_0$  is derived from the forward process, and  $\Theta$  represents the neural network parameters. The learned velocity field  $\mathbf{v}_\Theta(\mathbf{x}_t, t)$  transports data along the straight path without requiring complex temporal discretization.

## IV. TECHNICAL APPROACH

### A. Theoretical Framework of Canonical Representation

In visual imitation learning, while 3D point clouds combined with equivariant architectures can effectively enhance the generalization capability of robotic manipulation policies to unseen objects and scenes, most existing methods lack systematic theoretical foundations and rigorous mathematical formulations. From the perspectives of group theory and differential geometry, we extend the canonical policy framework by introducing a comprehensive theoretical framework for canonical representations. The overall pipeline of our proposed method is illustrated in Fig. 1.

Although point clouds may exhibit apparent diversity due to rigid transformations (e.g., rotations, translations) caused by viewpoint changes and object configuration variations, their intrinsic geometric properties remain invariant. Based on this observation, we can map 3D point cloud observations with different poses to a unified canonical space. We now introduce two core assumptions:

**Assumption 1 (Equivariant Partition)** We postulate that the domain of 3D point clouds admits a decomposition into a finite collection of disjoint subsets, formalized as  $\mathcal{D} = \{\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_p\}$ , where

$$\forall \mathbf{R} \in \mathbb{R}^{3 \times 3}, \forall \mathbf{b} \in \mathbb{R}^{3 \times 1}, \forall i \neq j, \quad \mathcal{G}_i \cap (\mathbf{R}\mathcal{G}_j + \mathbf{b}) = \emptyset \quad (8)$$

This assumption ensures that point clouds sharing the same geometric structure but with different poses can be correctly categorized into the same equivariant subgroup. From a geometric perspective, each equivariant subset  $\mathcal{G}_i$  essentially represents the  $SE(3)$  orbit of a canonical point cloud  $c_i$ :

$$\mathcal{G}_i = \{R \cdot \mathcal{G}_i^{cn} + b \mid (R, b) \in SE(3)\} \quad (9)$$

where  $\mathcal{G}_i^{cn}$  serves as the canonical representative of the orbit  $\mathcal{G}_i$ .

**Assumption 2 (Out-of-Distribution Approximation)** For a point cloud  $X$  sampled from an out-of-distribution task  $T$ , there exists a point cloud  $x_0$  from the demonstration dataset  $D$  and a rigid transformation  $(R, \mathbf{b}) \in SE(3)$ , such that:

$$X = R \cdot x_0 + \mathbf{b} + \epsilon \quad (10)$$

where  $\epsilon$  represents the permissible observation noise and shape variation, satisfying  $\|\epsilon\| \leq \delta$  with  $\delta > 0$  being a bounded error term. This assumption ensures that the diversity of the training dataset  $D$  is sufficient to cover novel observations encountered during testing through rigid transformations.

**Data Canonicalization Pipeline** Building upon the theoretical assumptions above, we propose a complete implementation pipeline for canonical representation, as illustrated in Fig. 2. This pipeline maps point clouds in arbitrary poses to a canonical space through rigorous geometric transformations. For any given point cloud instance  $x \in \mathcal{G}_i$ , the initial

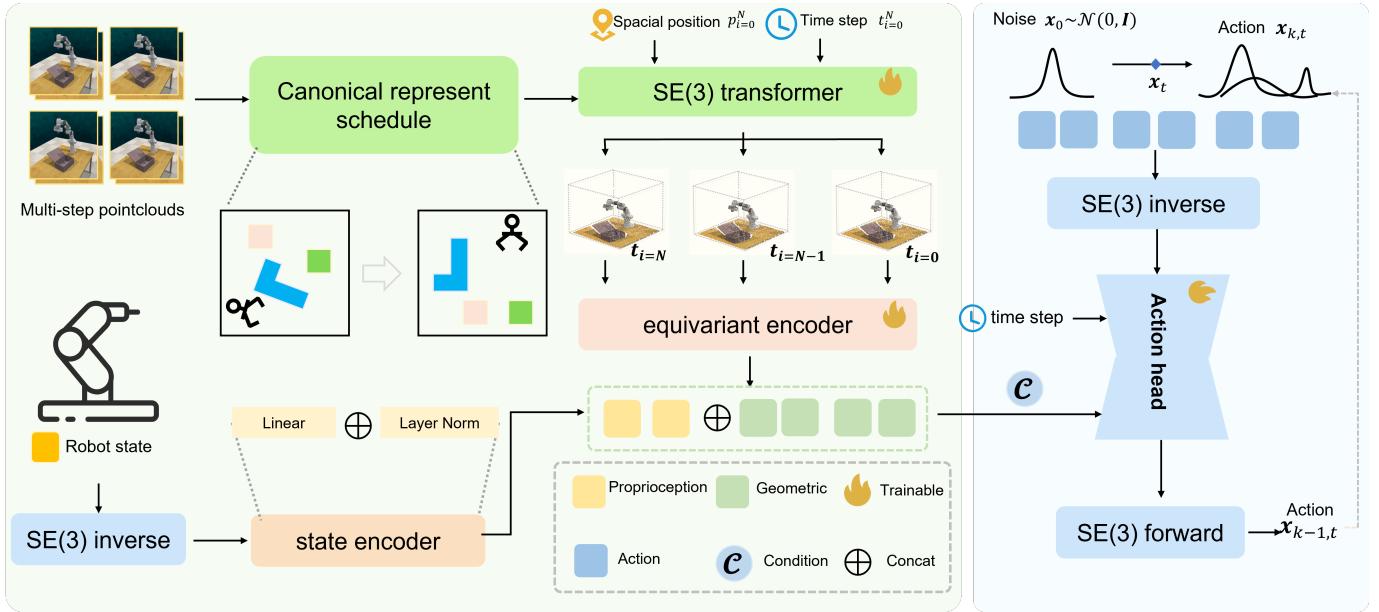


Fig. 1. Overview of the proposed CurveFlow framework. The pipeline integrates a Spatio-Temporal Canonical Representation module (left) for SE(3)-equivariant feature extraction and a Curvature-Guided Flow Matching Action Head (right) for precise motion generation.

preprocessing step involves calculating its geometric centroid  $\bar{x} \in \mathbb{R}^{3 \times 1}$  and performing a global subtraction to generate the decentralized form  $x^{de}$ . Next, we employ an SO(3)-equivariant network  $\Phi$  to estimate the relative rotation of the point cloud, yielding rotation-equivariant vectors  $r_{x1}, r_{x2} \in \mathbb{R}^3$ . The rotation matrix  $R$  is then constructed via the Schmidt orthogonalization process (see Appendix A for proof):

$$R = \begin{bmatrix} r_{x1} & \frac{r_{x2}}{\|r_{x2}\|} & \frac{r_{x1}}{\|r_{x1}\|} \times \frac{r_{x2}}{\|r_{x2}\|} \end{bmatrix} \quad (11)$$

Finally, we apply the inverse of the estimated rotation matrix to obtain the canonical representation of the point cloud:

$$x^{cn} = R^{-1}x^{de} = R^{-1}(x - \bar{x}) \quad (12)$$

To achieve an end-to-end equivariant policy, both the robot state and actions need to be synchronously mapped to the canonical space. Let the robot end-effector pose  $S_t \in SE(3)$  be composed of position  $s_t^{pos} \in \mathbb{R}^{3 \times 1}$  and orientation  $s_t^{ori} \in \mathbb{R}^{3 \times 3}$ , and the action pose  $A_{k,t} \in SE(3)$  be constructed from position  $a_{k,t}^{pos} \in \mathbb{R}^{3 \times 1}$  and orientation  $a_{k,t}^{ori} \in \mathbb{R}^{3 \times 3}$ . The canonical transformation is achieved through the SE(3) inverse operation:

$$S_t^{cn} = T_t^{-1}S_t, \quad A_{k,t}^{cn} = T_t^{-1}A_{k,t} \quad (13)$$

### B. SE(3)-Equivariant Temporal Aggregation Network

While the canonicalization module effectively standardizes individual frames, robust manipulation necessitates a consistent geometric understanding across the temporal dimension. To address this, we introduce an SE(3)-equivariant Transformer that aggregates spatiotemporal features while strictly preserving the intrinsic symmetry of rigid-body motions. Formally, given a point cloud sequence  $\{P_t \in \mathbb{R}^{N \times 3}\}_{t=1}^T$ , our

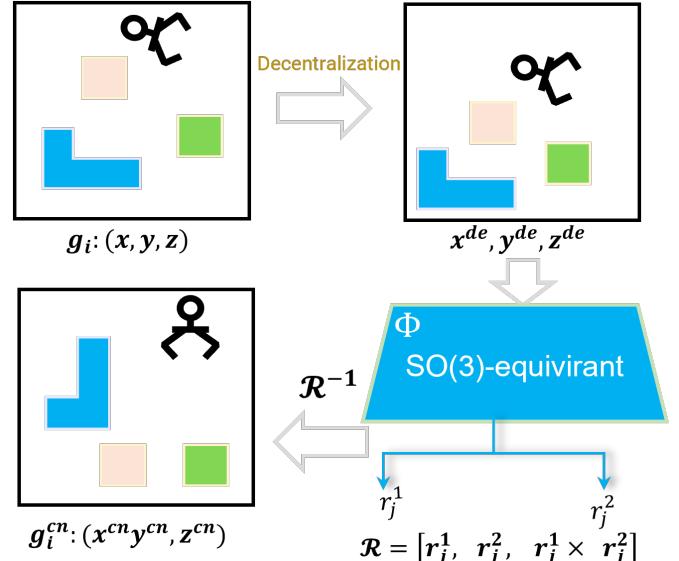


Fig. 2. Data Canonicalization Pipeline. It explicitly maps arbitrary 3D observations into a standardized coordinate system to decouple intrinsic geometry from global pose variations.

network  $\Phi$  ensures that temporal reasoning remains independent of global viewpoint shifts by satisfying the following equivariance property:

$$\Phi(g \cdot P_{1:T}) = g \cdot \Phi(P_{1:T}), \quad \forall g \in SE(3) \quad (14)$$

The aggregation process begins by projecting frame-wise inputs into a unified Vector-Scalar representation. A shared geometric encoder maps the observations at each timestep  $t$  into a feature pair  $h_t = (s_t, v_t)$ , where the decomposition is defined as:

$$h_t = (s_t, v_t), \quad s_t \in \mathbb{R}^{N \times C_s}, \quad v_t \in \mathbb{R}^{N \times 3C_v} \quad (15)$$

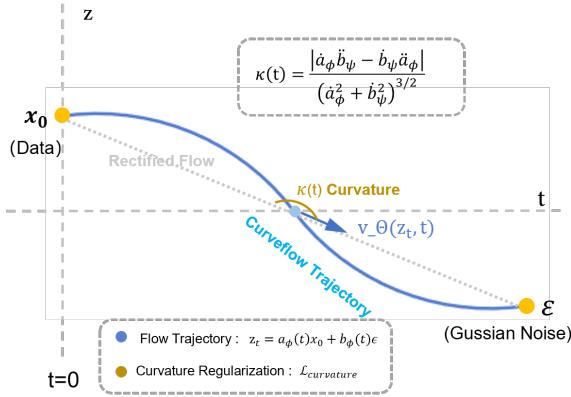


Fig. 3. Schematic illustration of CurveFlow. The diagram visualizes the curved trajectory established between the data sample ( $x_0$ ) and Gaussian noise ( $\epsilon$ ), standing in contrast to the linear path of Rectified Flow (dashed line). The trajectory is defined as  $z_t = a_\phi(t)x_0 + b_\phi(t)\epsilon$ , where the curvature  $\kappa(t)$  quantifies the deviation from linearity.

Here,  $s_t$  encapsulates rotation-invariant semantic information, while  $v_t$  retains directional geometric structures. This separation allows the network to process semantic context and geometric orientation through distinct, principled pathways.

To fuse these features effectively, we employ a dual-stream equivariant attention mechanism. Unlike standard transformers, our attention weights are computed based on pairwise invariant distances to maintain SE(3) consistency:

$$\alpha_{ij} = \text{softmax}_j(\phi(\|x_i - x_j\|)) \quad (16)$$

with  $\phi(\cdot)$  denoting a learnable radial basis kernel. Crucially, feature transport in the vector channel is performed via a geometric update rule that explicitly accounts for relative positions:

$$v'_i = \sum_j \alpha_{ij} \left( W_v v_j + B_v(x_j - x_i) \right) \quad (17)$$

This ensures that the feature aggregation remains equivariant under any rigid transformation.

We further explicitly decouple the attention process into spatial and temporal branches. While the spatial branch models local topology, the temporal branch captures motion dynamics by injecting sequence awareness through a coordinate-independent relative frame offset:

$$\alpha_{ij}^{(t,k)} = \text{softmax}_{j,k}(\phi(\|x_i^t - x_j^k\|) + \psi(t - k)) \quad (18)$$

Finally, the outputs of these spatial and temporal streams are consolidated through an equivariant residual update to produce the final canonical embeddings:

$$h_t^{\text{out}} = h_t + f_{\text{spatial}}(h_t) + f_{\text{temporal}}(h_{1:T}) \quad (19)$$

Comprehensive details regarding the network architecture and layer configurations are provided in Appendix B.

### C. Action Head: Curvature-Guided Flow Matching

In the domain of robot learning, the primary objective is to learn a policy capable of translating a history of observations—comprising 3D point clouds and robot states—into

predicted future action trajectories. After obtaining the canonical representations of the input point clouds and robot proprioceptive states, we process them through a point cloud aggregation encoder module to generate conditional features. These features are combined into a conditioning vector  $c$  that guides the action generation process.

To address the strong constraint of zero curvature in conditional flow matching, we model the action generation process using a curvature-guided flow matching method (as shown in Fig. 3). We define the trajectory  $z_t$  from the initial action distribution  $p_0$  to the target action distribution  $p_1$  as:

$$z_t = a_\phi(t)x_0 + b_\psi(t)\epsilon \quad (20)$$

where  $x_0 \sim p_0$  is the initial action,  $\epsilon \sim p_1$  is the target action or noise, and  $a_\phi(t)$  and  $b_\psi(t)$  are learnable coefficient functions that must satisfy the boundary conditions:

$$a_\phi(0) = 1, b_\psi(0) = 0, a_\phi(1) = 0, b_\psi(1) = 1. \quad (21)$$

This parameterization allows for nonlinear trajectories, whose curvature  $\kappa(t)$  can be expressed as:

$$\kappa(t) = \frac{|\dot{a}_\phi \ddot{b}_\psi - \dot{b}_\psi \ddot{a}_\phi|}{(\dot{a}_\phi^2 + \dot{b}_\psi^2)^{3/2}}. \quad (22)$$

To encourage the learning of smooth trajectories and prevent instability caused by excessive curvature, we introduce a curvature regularization term:

$$\mathcal{L}_{\text{curvature}} = \lambda \int_0^1 (\dot{a}_\phi(t)\ddot{b}_\psi(t) - \dot{b}_\psi(t)\ddot{a}_\phi(t))^2 dt \quad (23)$$

where  $\lambda > 0$  is a hyperparameter for the regularization loss, which adaptively penalizes the “twisting” behavior in the rate of change of the coefficient functions. This penalty is independent of the specific samples  $A_0$  and  $\epsilon$ , thereby ensuring training stability and generalization capability. To computationally evaluate this continuous integral during training, we employ a finite difference approximation scheme over a discretized temporal grid, the details of which are derived in Appendix D.2.

Combined with the data term from conditional flow matching, the total training objective for the action head is:

$$\mathcal{L}_{\text{Curve-FM}} = \mathbb{E}_{t,x_0,\epsilon} \left\| v_\Theta(z_t, t, c) - (\dot{a}_\phi(t)x_0 + \dot{b}_\psi(t)\epsilon) \right\|^2 \quad (24)$$

and

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{Curve-FM}} + \mathcal{L}_{\text{curvature}}. \quad (25)$$

Here  $v_\Theta(z_t, t, c)$  is the velocity field predicted by the neural network, and  $c$  represents the conditional features from the observation encoder. The design details of the U-Net backbone network and the curvature estimation network are provided in Appendix B.

## V. EXPERIMENTS

We systematically evaluate our proposed policy through both simulation and real-world experiments, with the goal of addressing the following research questions: (1) Does our method achieve superior performance and learning efficiency

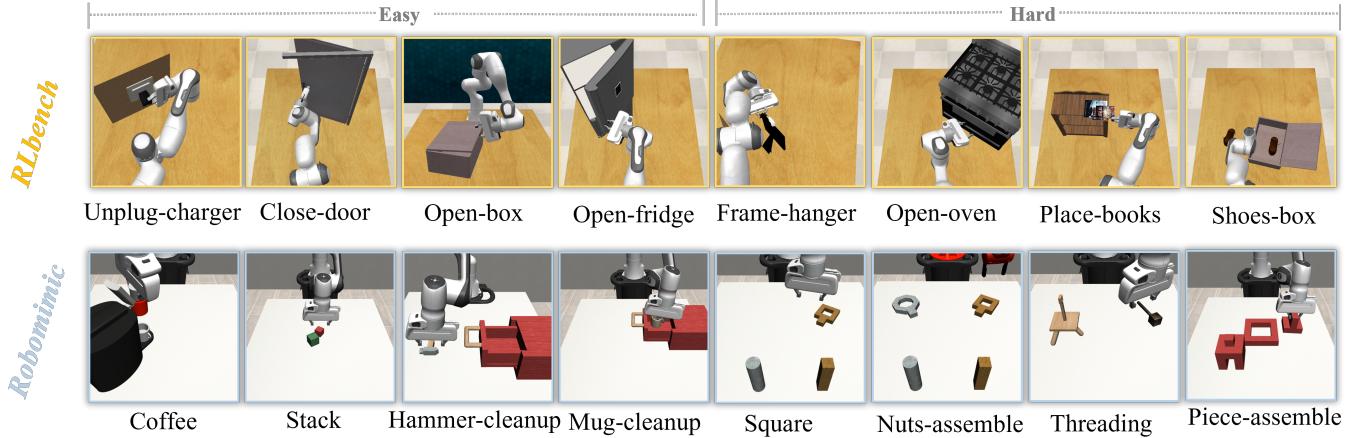


Fig. 4. Visualization of the 16 simulation tasks selected from the RLBench (top row) and RoboMimic (bottom row) benchmarks, covering diverse challenges such as articulated objects and long-horizon manipulation.

TABLE I  
PERFORMANCE COMPARISON OF DIFFERENT METHODS ON MULTIPLE RLBENCH TASKS.

Method	Success Rate (SR) (%)									Mean SR	Delta SR
	unplug charger	close door	open box	open fridge	frame hanger	open oven	place books	shoes box			
DP3 [9]	33.3 ± 4.7	76.0 ± 1.7	98.3 ± 1.5	4.3 ± 2.1	12.3 ± 2.5	0.9 ± 0.3	1.8 ± 0.6	0.0 ± 0.0	28.36	15.29	
iDP3 [14]	35.4 ± 3.6	74.3 ± 2.7	95.7 ± 4.2	5.7 ± 0.6	16.0 ± 2.6	1.3 ± 0.2	2.3 ± 0.4	0.0 ± 0.0	28.84	14.81	
EquiBot [31]	46.3 ± 1.5	53.3 ± 2.6	67.3 ± 3.8	2.8 ± 1.2	9.7 ± 3.8	0.0 ± 0.0	0.8 ± 0.2	0.0 ± 0.0	22.53	21.12	
Canonical-df [10]	55.2 ± 3.4	71.4 ± 4.2	96.1 ± 2.5	13.4 ± 2.3	27.6 ± 3.5	13.2 ± 2.8	11.2 ± 3.4	9.0 ± 1.4	37.14	6.51	
Canonical-fm [10]	57.7 ± 3.2	68.5 ± 3.5	97.4 ± 1.6	17.1 ± 3.2	29.4 ± 2.8	14.5 ± 2.5	13.7 ± 1.1	8.5 ± 2.1	38.35	5.30	
CurveFlow (ours)	61.4 ± 4.3	73.5 ± 4.6	99.1 ± 0.7	31.9 ± 2.9	31.3 ± 2.6	17.9 ± 2.1	18.1 ± 1.8	16.0 ± 3.4	43.65	–	

TABLE II  
PERFORMANCE COMPARISON OF DIFFERENT METHODS ON MULTIPLE ROBOMIMIC TASKS.

Method	Success Rate (SR) (%)								Mean SR	Delta SR
	coffee_d1	stack_d1	hammer_cl_d1	mug_cl_d0	square_d2	nuts_as_d0	threading_d1	piece_assemble_d0		
DP3 [9]	32 ± 12	22 ± 8	54 ± 6	6 ± 2	26 ± 8	18 ± 4	12 ± 2	16 ± 6	21.75	28.50
iDP3 [14]	32 ± 8	28 ± 8	46 ± 8	8 ± 4	22 ± 4	14 ± 4	10 ± 4	16 ± 4	20.75	29.50
EquiBot [31]	28 ± 10	8 ± 4	32 ± 10	12 ± 4	12 ± 4	13 ± 2	6 ± 2	21 ± 5	15.38	34.88
Canonical-df [10]	54 ± 4	78 ± 8	66 ± 6	28 ± 1	14 ± 4	36 ± 3	28 ± 6	39 ± 5	40.38	9.88
Canonical-fm [10]	58 ± 10	84 ± 10	68 ± 6	34 ± 8	16 ± 8	33 ± 1	34 ± 8	42 ± 6	44.00	6.25
CurveFlow (ours)	60 ± 6	78 ± 6	72 ± 8	40 ± 10	28 ± 6	38 ± 2	28 ± 6	48 ± 6	50.25	–

compared to existing point-cloud-based imitation learning approaches? (2) Can our method maintain robust performance under noisy conditions with point cloud perturbations? (3) Is our method applicable to real-world, real-time robotic manipulation tasks?

#### A. Simulation Benchmark

We select RoboMimic and RLbench, two well-established task simulators, to evaluate the performance of our proposed policy. A total of 16 representative tasks are considered, including *hammer cleanup*, *coffee preparation*, *piece assembly*, and *open door*. These tasks cover a wide range of manipulation scenarios, involving rigid objects and articulated objects, as well as complex settings such as long-horizon tasks.

We leverage the official data generation pipelines provided by the RoboMimic and RLbench benchmarks to curate high-

quality expert demonstrations. Specifically, for each of the 16 tasks, we utilize a dataset consisting of 100 expert trajectories to train our learning policy. Detailed visualizations of the simulation tasks are presented in Fig. 4.

#### B. Baselines

The primary objective of this work is to leverage equivariant representations of transformed 3D point clouds as visual guidance for flow-matching policies. Accordingly, we focus our comparisons on **DP3** [9], **iDP3** [14], and two recent state-of-the-art methods, **EquiBot** [31] and **Canonical Policy** [10]. For all baselines, we collect identical observations and adopt consistent training and evaluation protocols to ensure a fair comparison. Detailed hyperparameter configurations for optimization and data processing are provided in Appendix C.

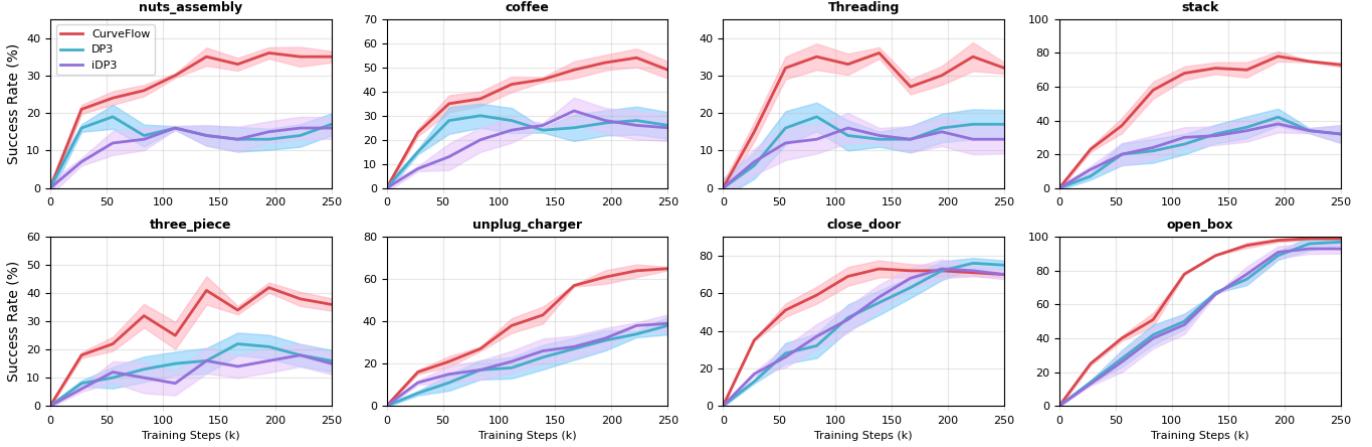


Fig. 5. Training efficiency comparison on representative tasks from RL Bench and RoboMimic. The curves illustrate the mean success rate over training steps, with shaded regions representing the standard deviation across different random seeds.

**DP3** is the first robotic policy that integrates 3D point clouds with diffusion models for manipulation learning. **iDP3** extends DP3 by introducing multi-layer 1D convolutional encoders, enabling the extraction of multi-resolution features from point cloud observations. **EquiBot** adopts a  $SIM(3)$ -equivariant neural network architecture, similar to that used in our approach; however, it combines diffusion-based policies with a reformulation of low-dimensional states and actions into vector scalar representations. **Canonical Policy** is a recent method grounded in canonical representation theory, which exploits the geometric symmetries of canonical forms together with diffusion models to achieve flexible and sample-efficient policy learning from 3D visual inputs.

### C. Analysis

Table I and Table II show that CurveFlow achieves state-of-the-art performance across all subtasks in RL Bench and Robomimic. The mean success rates reach 43.65% and 50.25% on RL Bench and Robomimic, respectively, significantly outperforming the previous best Canonical Policy (38.35% and 44.00%). Compared with existing approaches DP3, iDP3, and EquiBot, CurveFlow consistently yields higher success rates across almost all tasks, indicating that it is more effective for robotic imitation learning.

Across all these tasks in RL Bench and RoboMimic, CurveFlow not only maintains strong performance on the easy tasks but also achieves particularly large gains on the hard tasks, especially when compared with EquiBot and Canonical Policy. This demonstrates that CurveFlow is robust in more challenging scenarios and scales well with task difficulty. For the eight easy tasks, CurveFlow attains the highest average success rate among all methods, outperforming DP3, iDP3, and EquiBot by a clear margin, with an average improvement of nearly 20%. We further observe that Canonical Policy exhibits limited success on the *open\_fridge* task, primarily due to its susceptibility to dynamic self-occlusion, severing critical perceptual feedback and causing execution failure. In contrast, CurveFlow leverages spatiotemporally fused point cloud information and stable geometric representations, enabling it to

effectively overcome occlusion challenges and achieve more reliable performance.

To provide a comprehensive analysis of training dynamics, Fig. 5 visualizes the success rate trajectories for eight representative tasks from the RL Bench and Robomimic suites, comparing CurveFlow against DP3 and iDP3. While all evaluated methods show progressive improvement over time, CurveFlow distinguishes itself by demonstrating significantly faster convergence and superior asymptotic performance. Furthermore, the noticeably narrower error bands associated with CurveFlow underscore its enhanced training stability and robustness relative to the baseline approaches.

### D. Ablation Studies

We conduct ablation studies on multiple representative multi-scene tasks to evaluate the effectiveness of different components of our method. In particular, to rigorously validate robustness, we inject noise into the point clouds of arbitrary frames within the observation window, thereby simulating observation noise.

1) *Module Contribution Analysis*: To investigate the role of temporal modeling in stabilizing canonical representations, we remove the  $SE(3)$ -equivariant temporal transformer and retain only per-frame equivariant encoding. Table III shows that, under viewpoint occlusion, the per-frame equivariant baseline suffers a substantial drop in success rate. In contrast, under clean observation conditions, the performance gap between the two variants is marginal. These findings indicate that relying solely on single-frame geometric equivariance is insufficient to ensure policy stability. Instead, cross-frame temporal consistency plays a critical role in suppressing perceptual noise and mitigating error accumulation over time. We further push the limits of perceptual robustness by evaluating the policy under varying intensities of Gaussian noise, the details of which are provided in Appendix D.1.

To verify the contribution of canonical representations to generalization, we further remove the canonical transformation and train an equivariant policy directly in the original coordinate frame. Specifically, we report results on three RL Bench

TABLE III  
COMPARISON OF CURVEFLOW WITH AND WITHOUT  
SE(3)-TRANSFORMER AND OBSERVATION NOISE ON MULTIPLE TASKS.

SE(3) Transformer	pointcloud Noise	Mug cleanup	Hammer cleanup	Open oven	Shoes box	Avg. success
$\times$	$\times$	$35.7 \pm 3.2$	$68.5 \pm 3.5$	$14.4 \pm 1.6$	$12.1 \pm 3.2$	$32.68 \pm 2.8$
$\times$	✓	$19.1 \pm 3.6$	$9.5 \pm 2.1$	$3.7 \pm 1.2$	$4.2 \pm 1.4$	$9.13 \pm 2.4$
✓	$\times$	$41.7 \pm 2.1$	$71.8 \pm 3.3$	$18.6 \pm 1.8$	$16.3 \pm 2.6$	$37.10 \pm 2.5$
✓	✓	$41.4 \pm 2.3$	$69.5 \pm 3.6$	$19.1 \pm 2.5$	$15.9 \pm 2.9$	$36.48 \pm 2.6$

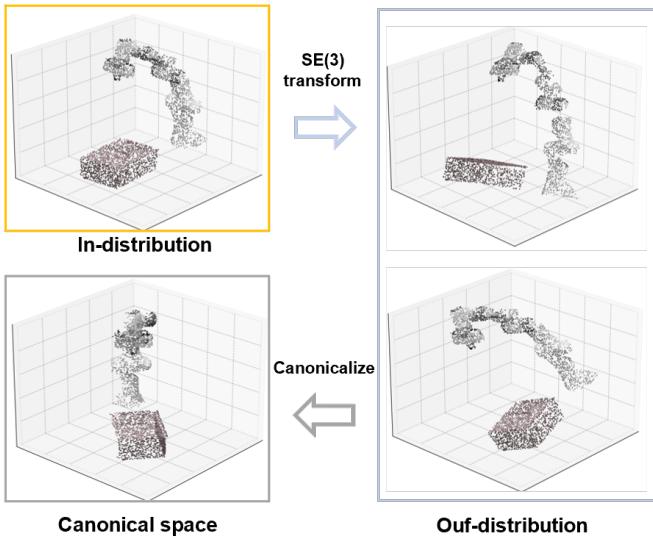


Fig. 6. Schematic illustration of the transformation process from the original coordinate frame to the canonical frame.

tasks with action rotation characteristics and construct the out-of-distribution evaluation set by applying random SE(3) transformations to the initial object poses, creating significant geometric shifts relative to the training distribution (visualized in Fig. 6). Subsequently, the quantitative results in Fig. 7 reveal that red curves correspond to the setting where the policy is trained and evaluated on the same distribution. Under this condition, all policies achieve strong performance. However, when evaluated on unseen object poses, only the canonical policy maintains stable performance, whereas the non-canonical variant degrades significantly. These results confirm that explicit canonical representations play a central role in reducing the difficulty of policy learning and improving robustness under distribution shifts.

**2) Curvature-Guided Flow Analysis:** To assess the effectiveness of Curvature-Guided Flow, we replace it with a standard linear Rectified Flow [17]. The results in Table IV show that the curvature-guided mechanism leads to a significant improvement in success rates, indicating that probability transport paths with non-zero curvature better align with the underlying structure of real action manifolds. Furthermore, when the curvature regularization term is removed, training performance degrades noticeably, and in some tasks even falls below that of the standard linear Rectified Flow. This observation highlights the critical role of curvature regularization in enhancing training stability. Additional analyses and results are provided in Appendix D.3.

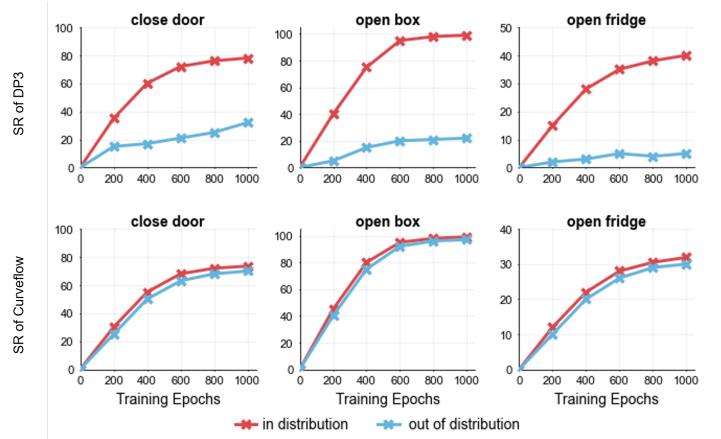


Fig. 7. Ablation study of CurveFlow with curvature guided (CG) and curvature regularization (CR).



Fig. 8. Real-world experimental setup. The platform consists of: (1) the Kinova robotic manipulator, (2) the Robotiq 2F-85 gripper, (3) the manipulation objects, and (4) the Intel RealSense L515 RGB-D camera.

### E. Real-World Experiments

**1) Experimental Setup:** To further validate the practical applicability of the proposed CurveFlow policy beyond simulation, we conduct a systematic evaluation on a real Kinova robotic platform, as shown in Fig. 8. The robot operates in a tabletop workspace of approximately  $0.8 \text{ m} \times 0.6 \text{ m}$ . Visual perception is provided by an Intel RealSense L515 RGB-D camera, mounted above the workspace in an eye-to-hand configuration. Depth images are converted into point clouds in real time and transformed into the robot base frame using a calibrated extrinsic transformation.

Following a setup similar to TidyBot++ [22], the system control frequency is set to 20 Hz. At each control cycle, the policy server predicts a fixed-horizon action chunk and transmits it to an arm server responsible for low-level execution of end-effector motions.

**2) Real-World Tasks:** As Fig. 9 shows, we evaluate the proposed method on four representative real-world manipulation tasks:

**Sorting parts:** Placing parts into a specific container requires medium end-effector posture accuracy.

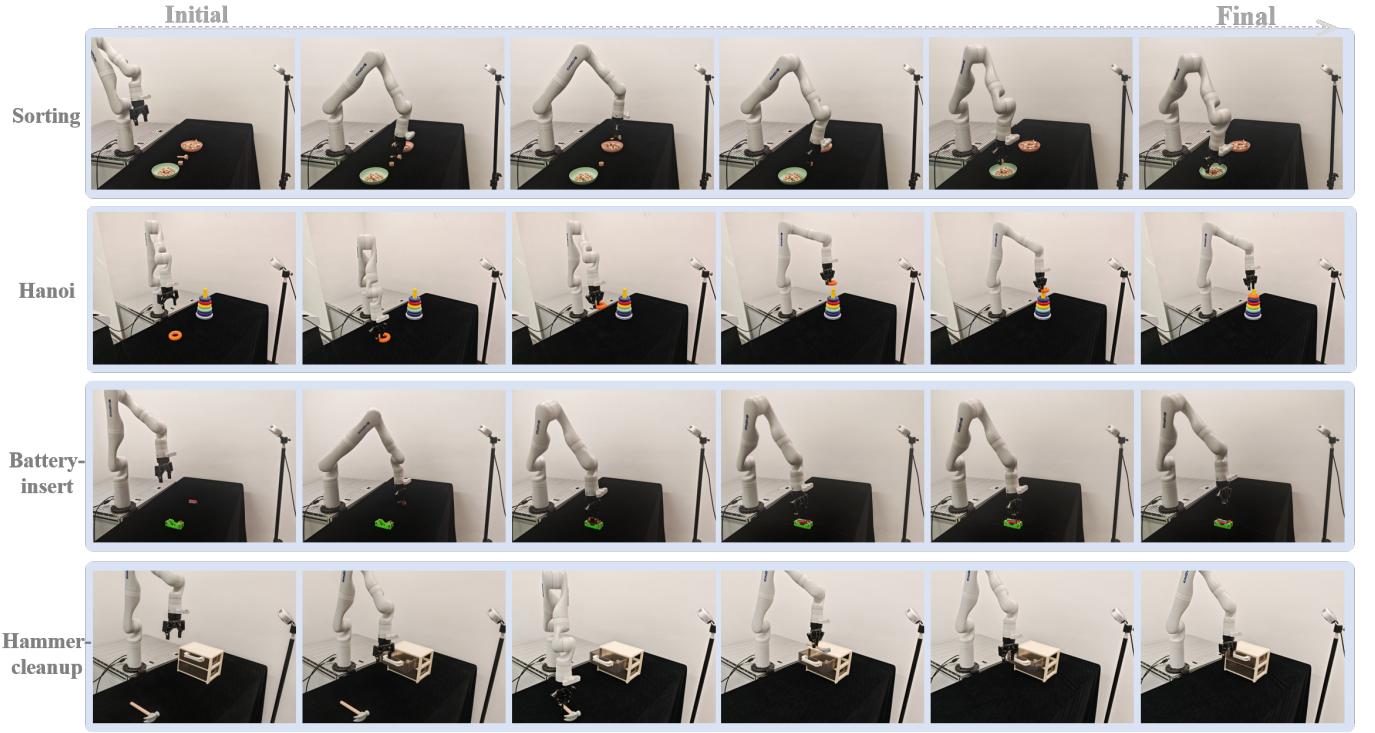


Fig. 9. Execution sequences of real-world manipulation tasks. From top to bottom: Sorting parts, Hanoi, Insert battery and Hammer cleanup. The snapshots demonstrate the robot's ability to handle precise pick-and-place and contact-rich operations.

TABLE IV  
ABLATION STUDY OF CURVEFLOW WITH CURVE-GUIDED (CG) AND CURVATURE REGULARIZATION (CR).

CG CR	RLbench easy (4)	RLbench hard (4)	Robomimic easy (4)	Robomimic hard (4)
✗ ✗	52.57 ± 2.14	16.83 ± 2.28	52.30 ± 4.26	23.38 ± 2.47
✓ ✗	46.73 ± 5.19	14.56 ± 4.28	53.17 ± 6.78	18.49 ± 4.15
✓ ✓	<b>66.73 ± 1.74</b>	<b>20.83 ± 1.28</b>	<b>62.50 ± 3.84</b>	<b>35.50 ± 2.65</b>

**Hanoi:** Place the ring on the top of the tower of Hanoi, which requires precise alignment and end-effector posture accuracy.

**Insert battery:** Inserting the battery into a narrow base requires high posture accuracy and smooth trajectory execution.

**Hammer cleanup:** Cleanup the hammer by placing it in a specific container, which requires precise alignment and continuous contact control.

For each task, 50 expert demonstration trajectories are collected via teleoperation. Demonstrations are recorded as synchronized sequences of point clouds and robot proprioceptive states. Policy training is conducted entirely in an offline imitation learning setting, without any interaction with the real robot during training. Each task is executed for 20 independent trials under randomized object initial poses. During evaluation, the policy is deployed on the real robot in a zero-shot manner, i.e., without any additional fine-tuning in the real environment.

**3) Results:** We compare CurveFlow with the representative baseline DP3, and the quantitative results are summarized in Table V. Across all real-world tasks, CurveFlow consistently

achieves higher success rates than DP3, demonstrating a clear performance advantage.

TABLE V  
REAL-WORLD PERFORMANCE COMPARISON BETWEEN CURVEFLOW AND DP3.

Task	In Distribution		Unseen Objects		Avg. Inference Time	
	DP3	CurveFlow	DP3	CurveFlow	DP3	CurveFlow
Sorting	7/20	<b>14/20</b>	3/20	<b>12/20</b>	92 ms	<b>85 ms</b>
Hanoi	8/20	<b>13/20</b>	2/20	<b>12/20</b>	86 ms	<b>74 ms</b>
Battery	5/20	<b>11/20</b>	1/20	<b>7/20</b>	76 ms	<b>61 ms</b>
Hammer	2/20	<b>10/20</b>	0/20	<b>6/20</b>	98 ms	<b>89 ms</b>
Avg.SR	27.5%	<b>60.0%</b>	15.0%	<b>46.3%</b>	88 ms	<b>77.3 ms</b>

In particular, for the *Insert battery* task, DP3 frequently exhibits abrupt trajectory corrections during the contact phase, which often lead to execution failures. In contrast, CurveFlow generates smoother approach trajectories, significantly reducing late-stage failures. These results indicate that the proposed method is capable of maintaining stable perception-action coupling under real-world uncertainties, thereby demonstrating its effectiveness and robustness in practical robotic manipulation scenarios.

## VI. CONCLUSION

In this work, we proposed CurveFlow, a robust imitation learning framework that addresses the challenges of generalization and dynamic consistency in robotic manipulation.

By integrating a spatio-temporal SE(3)-canonical representation with a curvature-guided flow matching strategy, the proposed method effectively resolves perceptual instability under viewpoint variations and mitigates the limitations of linear trajectory assumptions in complex action manifolds.

Despite these advancements, the current framework presents limitations. Primarily, the policy relies on geometric visual observations, lacking the integration of high-level semantic understanding and fine-grained physical feedback, which restricts its applicability in tasks requiring semantic reasoning or precise force control. Future work will focus on extending the geometry-aware framework to incorporate multi-modal inputs to further enhance dexterous manipulation capabilities in unstructured environments.

## REFERENCES

- [1] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.
- [2] J. Song, C. Meng, and S. Ermon, “Denoising diffusion implicit models,” *arXiv preprint arXiv:2010.02502*, 2020.
- [3] Y. Lipman, R. T. Chen, H. Ben-Hamu, M. Nickel, and M. Le, “Flow matching for generative modeling,” *arXiv preprint arXiv:2210.02747*, 2022.
- [4] 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,” *The International Journal of Robotics Research*, p. 02783649241273668, 2023.
- [5] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, “Learning fine-grained bimanual manipulation with low-cost hardware,” *arXiv preprint arXiv:2304.13705*, 2023.
- [6] M. Shridhar, L. Manuelli, and D. Fox, “Perceiver-actor: A multi-task transformer for robotic manipulation,” in *Conference on Robot Learning*. PMLR, 2023, pp. 785–799.
- [7] T. Gervet, Z. Xian, N. Gkanatsios, and K. Fragkiadaki, “Act3d: 3d feature field transformers for multi-task robotic manipulation,” *arXiv preprint arXiv:2306.17817*, 2023.
- [8] A. Goyal, V. Blukis, J. Xu, Y. Guo, Y.-W. Chao, and D. Fox, “Rvt-2: Learning precise manipulation from few demonstrations,” *arXiv preprint arXiv:2406.08545*, 2024.
- [9] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu, “3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations,” *arXiv preprint arXiv:2403.03954*, 2024.
- [10] Z. Zhang, Z. Xu, J. N. Lakamsani, and Y. She, “Canonical policy: Learning canonical 3d representation for equivariant policy,” *arXiv preprint arXiv:2505.18474*, 2025.
- [11] H. Zhu, Y. Wang, D. Huang, W. Ye, W. Ouyang, and T. He, “Point cloud matters: Rethinking the impact of different observation spaces on robot learning,” *Advances in Neural Information Processing Systems*, vol. 37, pp. 77 799–77 830, 2024.
- [12] D. Wang, S. Hart, D. Surovik, T. Kelestemur, H. Huang, H. Zhao, M. Yeatman, J. Wang, R. Walters, and R. Platt, “Equivariant diffusion policy,” *arXiv preprint arXiv:2407.01812*, 2024.
- [13] X. Ma, S. Patidar, I. Haughton, and S. James, “Hierarchical diffusion policy for kinematics-aware multi-task robotic manipulation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 18 081–18 090.
- [14] 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,” *arXiv e-prints*, pp. arXiv–2410, 2024.
- [15] A. Tong, N. Malkin, G. Huguet, Y. Zhang, J. Rector-Brooks, K. Fatras, G. Wolf, and Y. Bengio, “Conditional flow matching: Simulation-free dynamic optimal transport,” *arXiv preprint arXiv:2302.00482*, vol. 2, no. 3, 2023.
- [16] X. Liu, C. Gong, and Q. Liu, “Flow straight and fast: Learning to generate and transfer data with rectified flow,” *arXiv preprint arXiv:2209.03003*, 2022.
- [17] Q. Liu, “Rectified flow: A marginal preserving approach to optimal transport,” *arXiv preprint arXiv:2209.14577*, 2022.
- [18] P. Esser, S. Kulal, A. Blattmann, R. Entezari, J. Müller, H. Saini, Y. Levi, D. Lorenz, A. Sauer, F. Boesel *et al.*, “Scaling rectified flow transformers for high-resolution image synthesis,” in *Forty-first international conference on machine learning*, 2024.
- [19] Y. Qin, B. Huang, Z.-H. Yin, H. Su, and X. Wang, “Dexpoint: Generalizable point cloud reinforcement learning for sim-to-real dexterous manipulation,” in *Conference on Robot Learning*. PMLR, 2023, pp. 594–605.
- [20] E. Chisari, N. Heppert, M. Argus, T. Welschehold, T. Brox, and A. Valada, “Learning robotic manipulation policies from point clouds with conditional flow matching,” *arXiv preprint arXiv:2409.07343*, 2024.
- [21] C. Wang, H. Fang, H.-S. Fang, and C. Lu, “Rise: 3d perception makes real-world robot imitation simple and effective,” in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 2870–2877.
- [22] J. Wu, W. Chong, R. Holmberg, A. Prasad, Y. Gao, O. Khatib, S. Song, S. Rusinkiewicz, and J. Bohg, “Tidybot++: An open-source holonomic mobile manipulator for robot learning,” *arXiv preprint arXiv:2412.10447*, 2024.
- [23] T. Gervet, Z. Xian, N. Gkanatsios, and K. Fragkiadaki, “Act3d: 3d feature field transformers for multi-task robotic manipulation,” *arXiv preprint arXiv:2306.17817*, 2023.
- [24] I. Liu, C. Arthur, S. He, D. Seita, and G. Sukhatme, “Voxact-b: Voxel-based acting and stabilizing policy for bimanual manipulation,” *arXiv preprint arXiv:2407.04152*, 2024.
- [25] H. Huang, H. Liu, D. Wang, R. Walters, and R. Platt, “Match policy: A simple pipeline from point cloud registration to manipulation policies,” in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 16 907–16 914.
- [26] R. Yang, G. Chen, C. Wen, and Y. Gao, “Fp3: A 3d foundation policy for robotic manipulation,” *arXiv preprint arXiv:2503.08950*, 2025.
- [27] H. Huang, D. Wang, A. Tangri, R. Walters, and R. Platt, “Leveraging symmetries in pick and place,” *The International Journal of Robotics Research*, vol. 43, no. 4, pp. 550–571, 2024.
- [28] A. Simeonov, Y. Du, Y.-C. Lin, A. R. Garcia, L. P. Kaelbling, T. Lozano-Pérez, and P. Agrawal, “Se (3)-equivariant relational rearrangement with neural descriptor fields,” in *Conference on Robot Learning*. PMLR, 2023, pp. 835–846.
- [29] D. Wang, S. Hart, D. Surovik, T. Kelestemur, H. Huang, H. Zhao, M. Yeatman, J. Wang, R. Walters, and R. Platt, “Equivariant diffusion policy,” *arXiv preprint arXiv:2407.01812*, 2024.
- [30] B. Eisner, Y. Yang, T. Davchev, M. Vecerik, J. Scholz, and D. Held, “Deep se (3)-equivariant geometric reasoning for precise placement tasks,” *arXiv preprint arXiv:2404.13478*, 2024.
- [31] J. Yang, Z.-a. Cao, C. Deng, R. Antonova, S. Song, and J. Bohg, “Equibot: Sim (3)-equivariant diffusion policy for generalizable and data efficient learning,” *arXiv preprint arXiv:2407.01479*, 2024.
- [32] C. Deng, O. Litany, Y. Duan, A. Poulenard, A. Tagliasacchi, and L. J. Guibas, “Vector neurons: A general framework for so (3)-equivariant networks,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 12 200–12 209.
- [33] C. Tie, Y. Chen, R. Wu, B. Dong, Z. Li, C. Gao, and H. Dong, “Etsed: Efficient trajectory-level se (3) equivariant diffusion policy,” *arXiv preprint arXiv:2411.03990*, 2024.
- [34] H. Ding, N. Jaquier, J. Peters, and L. Rozo, “Fast and robust visuomotor riemannian flow matching policy,” *IEEE Transactions on robotics*, 2025.
- [35] Q. Zhang, Z. Liu, H. Fan, G. Liu, B. Zeng, and S. Liu, “Flowpolicy: Enabling fast and robust 3d flow-based policy via consistency flow matching for robot manipulation,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 14, 2025, pp. 14 754–14 762.
- [36] X. Hu, Q. Liu, X. Liu, and B. Liu, “Adaflow: Imitation learning with variance-adaptive flow-based policies,” *Advances in Neural Information Processing Systems*, vol. 37, pp. 138 836–138 858, 2024.
- [37] S. Jiang, X. Fang, N. Roy, T. Lozano-Pérez, L. P. Kaelbling, and S. Anchira, “Streaming flow policy: Simplifying diffusion / flow-matching policies by treating action trajectories as flow trajectories,” *arXiv preprint arXiv:2505.21851*, 2025.
- [38] J. Sheng, Z. Wang, P. Li, and M. Liu, “Mp1: Meanflow tames policy learning in 1-step for robotic manipulation,” *arXiv preprint arXiv:2507.10543*, 2025.