EfficientFlow: Efficient Equivariant Flow Policy Learning for Embodied AI

Jianlei Chang[1*], **Ruofeng Mei**[1*], **Wei Ke**[1], **Xiangyu Xu**[1†]

[1]Xi'an Jiaotong University

changjl@stu.xjtu.edu.cn, mrf2022@stu.xjtu.edu.cn, wei.ke@xjtu.edu.cn, xiangyu.xu@xjtu.edu.cn

Abstract

Generative modeling has recently shown remarkable promise for visuomotor policy learning, enabling flexible and expressive control across diverse embodied AI tasks. However, existing generative policies often struggle with *data inefficiency*, requiring large-scale demonstrations, and *sampling inefficiency*, incurring slow action generation during inference. We introduce EfficientFlow, a unified framework for efficient embodied AI with flow-based policy learning. To enhance data efficiency, we bring equivariance into flow matching. We theoretically prove that when using an isotropic Gaussian prior and an equivariant velocity prediction network, the resulting action distribution remains equivariant, leading to improved generalization and substantially reduced data demands. To accelerate sampling, we propose a novel acceleration regularization strategy. As direct computation of acceleration is intractable for marginal flow trajectories, we derive a novel surrogate loss that enables stable and scalable training using only conditional trajectories. Across a wide range of robotic manipulation benchmarks, the proposed algorithm achieves competitive or superior performance under limited data while offering dramatically faster inference. These results highlight EfficientFlow as a powerful and efficient paradigm for high-performance embodied AI.

Introduction

Learning robotic policies from data using generative models has emerged as a powerful and flexible paradigm in embodied AI, particularly with the recent success of diffusion-based approaches (Chi et al. 2023; Ze et al. 2024). These models have demonstrated strong performance in visuomotor control by learning complex action distributions conditioned on high-dimensional observations. However, two key limitations remain: low data efficiency, requiring large amounts of training data, and low sampling efficiency, incurring high computational cost at inference due to the iterative sampling process.

Recent works have sought to address the data efficiency issue by incorporating equivariance into diffusion models for policy learning (Wang et al. 2024). By leveraging the inherent symmetries of the environment (e.g., 2D rotation),

---

*Equal contribution.

†Corresponding author.

these methods introduce strong inductive biases that enable policies to generalize across symmetric configurations. Nevertheless, as they are still built upon diffusion models, which typically require hundreds of iterative denoising steps to generate a single action (Sohl-Dickstein et al. 2015; Ho, Jain, and Abbeel 2020), they remain impractical for real-time robotic control. To overcome this limitation, we turn to Flow Matching (Lipman et al. 2022), a recent class of generative models that learns a continuous trajectory from a simple prior distribution to the data distribution using an ordinary differential equation (ODE) defined by a velocity field. Compared to diffusion models, flow-based approaches offer better numerical stability and faster inference, making them highly appealing for efficient embodied AI.

We present EfficientFlow, a new policy learning framework that unifies equivariant learning and flow-based generative modeling. We first investigate how to incorporate equivariance into flow-based policy models and theoretically show that, under an isotropic Gaussian prior and an equivariant velocity field network, the conditional action distribution induced by flow matching remains equivariant with respect to input observation transformations (see Figure 1(a)). This property allows policies to generalize across symmetric configurations of the environment without additional supervision or data augmentation.

To further improve the action sampling efficiency, we introduce a regularization technique that penalizes the acceleration of the generation flow trajectory, i.e., the second-order temporal derivative, which encourages a smoother and more stable action sampling process.

However, computing acceleration requires consecutive points along the marginal flow trajectories, which are unavailable in the standard flow matching framework. To address this challenge, we propose a novel surrogate objective called Flow Acceleration Upper Bound (FABO). FABO provides a practical and effective approximation of the acceleration penalty using only conditional flow trajectories available during training, enabling much faster flow policies with lower computational costs.

The proposed EfficientFlow combines the best of both worlds: it achieves fast inference speed thanks to the flow-based architecture and smoothed sampling trajectory, and maintains high performance by leveraging equivariance. As illustrated in Figure 1(b), EfficientFlow compares favorably

against existing methods in both inference speed and task success rates.

Our primary contributions are as follows:

- We formulate a flow-based policy learning framework, EfficientFlow, that achieves equivariance to geometric transformations, allowing the model to generalize across symmetric states and significantly improve data efficiency. We provide a theoretical analysis showing that equivariance is preserved in the flow framework when using an isotropic prior and an equivariant velocity field conditioned on visual observations.

- To promote sampling speed, we propose a second-order regularization objective that penalizes flow acceleration. Since direct acceleration computation requires access to neighboring marginal samples that are unavailable, we introduce a novel surrogate loss called FABO, enabling effective training.

- We provide comprehensive evaluations of EfficientFlow on 12 robotic manipulation tasks in the MimicGen (Mandlekar et al. 2023) benchmark, showing that EfficientFlow achieves favorable success rates with high inference speeds (19.9 to 56.1 times faster than EquiDiff (Wang et al. 2024)).

Related Work

Equivariance in Robot Manipulation. Applying equivariance to robot manipulation is a highly promising research direction, and multiple studies have demonstrated that it can significantly enhance the data efficiency of robot policy learning (Wang, Walters, and Platt 2022; Jia et al. 2023; Wang et al. 2022b; Simeonov et al. 2023; Pan et al. 2023; Huang et al. 2023; Liu et al. 2023; Kim et al. 2023; Nguyen et al. 2023; Yang et al. 2024a). Early work used SE(3) open-loop or SE(2) closed-loop for control, validated the effectiveness of equivariant models in on-robot learning (Zhu and Wang 2022; Wang et al. 2022a; Zhu et al. 2023), and achieved pick-and-place tasks based on few-shot demonstrations (Huang et al. 2022; Simeonov et al. 2022; Ryu et al. 2022; Huang et al. 2024). Building on this foundation, EquiDiff (Wang et al. 2024) extends the research to the SE(3) closed-loop action space, substantially improving the efficiency of imitation learning by integrating symmetry with diffusion policies. However, the DDPM architecture employed by EquiDiff requires a multi-step denoising process, resulting in slow inference speeds. In contrast, the EfficientFlow model marks a significant breakthrough in inference efficiency, attaining higher success rates than EquiDiff with only a minimal number of inference steps.

Flow Policy. Flow Matching (Lipman et al. 2022) represents a novel class of generative models grounded in optimal transport theory. Its objective is to learn a vector field of a probability path, which is more efficient than diffusion paths, offering faster training and sampling, alongside better generalization capabilities. Compared to diffusion models, Flow Matching significantly reduces the number of inference steps, a critical factor for real-world robotic operations, thereby substantially broadening the applicability of such models. The work Flow Policy (Zhang et al. 2025) introduced conditional Consistent Flow Matching (Yang et al. 2024b) to robotic manipulation. Conditioned on observed 3D point clouds, Flow Policy utilizes Consistency Flow Matching to directly define straight-line flows from different temporal states to the same action space, concurrently constraining their velocity values. It approximates trajectories from noise to robot actions by normalizing the self-consistency of the velocity field within the action space, thereby enhancing inference efficiency. MP1 (Sheng et al. 2025) leverages Mean Flow (Geng et al. 2025) to shrink policy learning to a single state-action step, while a lightweight Dispersive Loss repels state embeddings. This combination steadies the flow field and delivers millisecond inference that outpaces DP3 and Flow Policy. Currently, many VLA (Vision-Language-Action) models (Black et al. 2024; Gao et al. 2025; Bjorck et al. 2025; Reuss et al. 2025) are utilizing flow matching policies and have achieved good results.

Method

Preliminaries

Flow Matching. The core idea of Flow Matching (Lipman et al. 2022) is to learn the vector field of an ODE that smoothly transforms samples x_0 from a simple prior distribution p_0 (e.g., Gaussian noise) to samples x_1 from a target data distribution p_1.

Specifically, let $\{p_t\}_{t \in [0,1]}$ be a time-evolving family of probability distributions satisfying the boundary conditions $p_{t=0} = p_0$ and $p_{t=1} = p_1$. This path induces an underlying ground-truth instantaneous velocity field $u^{gt}(t, x)$. Flow Matching aims to learn a vector field $u_\theta(t, x)$ parameterized by θ, such that trajectories x_t defined by the following ODE:

$$\begin{cases} \frac{dx_t}{dt} = u_\theta(t, x_t) \\ x_0 \sim p_0 \end{cases} \quad (1)$$

can effectively transport the prior distribution p_0 to the target distribution p_1. Ideally, the learned vector field $u_\theta(t, x)$ should approximate the true vector field $u^{gt}(t, x)$. Thus, a natural learning objective for Flow Matching is:

$$L_{FM} = \mathbb{E}_{t, x_t} \left[\left\| u_\theta(t, x_t) - u^{gt}(t, x_t) \right\|_2^2 \right], \quad (2)$$
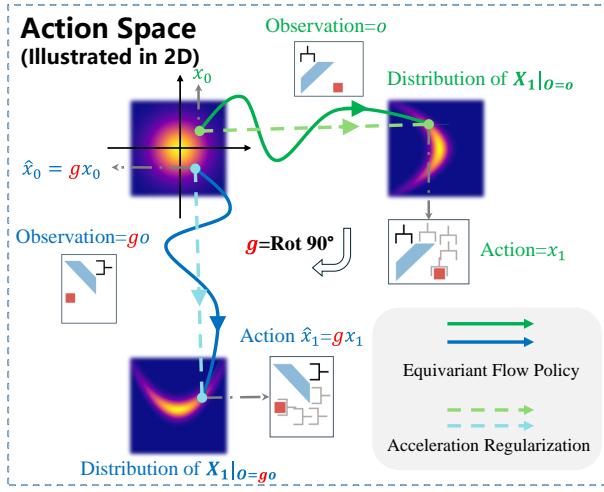
where $x_t \sim p_t$.

As $u^{gt}(t, x_t)$ is generally intractable in practice, Conditional Flow Matching (CFM) (Lipman et al. 2022) proposes to learn $u_\theta(t, x_t)$ by regressing against a conditional vector field $u(t, x_t | x_1)$, using samples from a conditional probability path $p_t(x | x_1)$. The corresponding objective is:

$$L_{CFM} = \mathbb{E}_{t, x_1, x_t} \left[\left\| u_\theta(t, x_t) - u(t, x_t | x_1) \right\|_2^2 \right], \quad (3)$$

where $t \sim U(0, 1)$, $x_1 \sim p_1(x)$, and $x_t \sim p_t(x | x_1)$.

Equivariance. Equivariance is a desirable property in many learning systems, especially when modeling structured data influenced by known symmetries (Cesa, Lang, and Weiler 2022). A function f is said to be equivariant with

(a) Key Concepts of EfficientFlow

(b) Comparison with SOTA
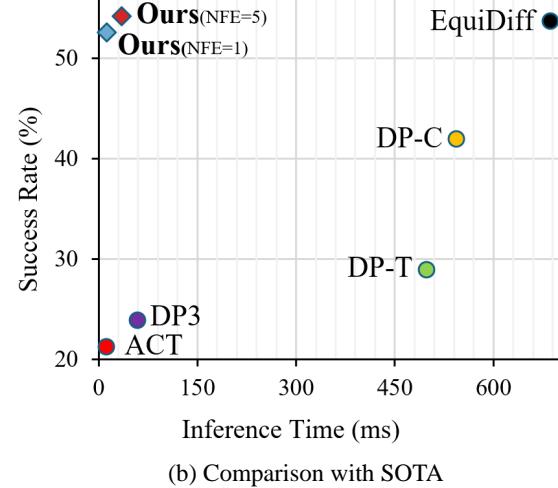
Figure 1: We propose EfficientFlow (a) to effectively combine equivariance with Flow Policy and introduce an acceleration regularization to achieve high-quality, fast action generation. As shown in (b), EfficientFlow compares favorably against baseline policy learning approaches in both success rate and inference speed. Results are from MimicGen with 100 training demonstrations.

respect to a transformation group G if it commutes with the actions of the group. Formally, this is expressed as:

$$f(\rho_x(g)x) = \rho_y(g)f(x), \quad \forall g \in G, \qquad (4)$$

where ρ_x and ρ_y denote group representations that describe how the group acts on the input space and output space, respectively. This equation ensures that applying a group transformation to the input and then evaluating the function (Eq. 4 left) yields the same result as first applying the function and then transforming the output (Eq. 4 right).

In this work, we focus on learning equivariant policies for robot arm control, where the input x represents the robot arm action in the task space. Since robotic manipulation tasks often exhibit rotational symmetry (Wang et al. 2024; Wang, Walters, and Platt 2022), we study the action of the rotation group $\mathrm{SO}(2)$ and its finite cyclic subgroup $C_u \subset \mathrm{SO}(2)$, which models discrete rotational symmetries (e.g., rotations by $\frac{2\pi}{u}$ radians).

We consider the following standard representations: 1) the trivial representation ρ_0, which maps every group element $g \in G$ to the identity transformation. This is typically used when the function output should remain invariant under the group action. 2) the standard irreducible representation ρ_1, which describes the canonical action of $\mathrm{SO}(2)$ or C_u on the 2D plane, defined as $\rho_1(g) = \begin{bmatrix} \cos(g) & -\sin(g) \\ \sin(g) & \cos(g) \end{bmatrix}$, where we slightly abuse the notation to use g denotes both group element and its corresponding rotation angle.

By designing policy networks that are equivariant under these group actions, we aim to incorporate inductive biases that reflect the underlying symmetries of the robot's action space. This not only improves sample efficiency but also enhances generalization across real task configurations.

Equivariant Flow Policy

Generative models for policy learning have received significant attention in recent years. Given an observation o, such models can predict a conditional distribution $p_{X_1|O=o}$, and generate robot actions by sampling $x \sim X_1\big|_{O=o}$, where X_1 represents the random variable for the action to be executed by the robot arm under the condition that $O = o$.

Both o and x can span multiple time steps: $o = [o^{\tau-(m-1)}, \cdots, o^{\tau-1}, o^\tau]$, $x = [x^\tau, x^{\tau+1}, \cdots, x^{\tau+(n-1)}]$, where m is the number of historical observations, and n is the number of future action steps. The observation o^τ includes both image and the robot state at robot time τ.

A desirable property for such models is equivariance: when the input o is transformed by an element $g \in G$ of a symmetry group (e.g., a rotation), the conditional distribution of the output action should transform accordingly. In other words, symmetry in the observation space should induce symmetry in the action space:

$$X_1\big|_{O=go} \overset{d}{=} g\left(X_1\big|_{O=o}\right), \qquad (5)$$

where $\overset{d}{=}$ denotes that the two random variables have the same distribution. We leave the group representation $\rho(g)$ implicit here and directly use g for brevity.

How to Make Flow Policy Equivariant? The main contribution of this work is to demonstrate that the desired property in Eq. 5 can be achieved within the Flow Matching framework by:

1. using an isotropic distribution for p_0 in Eq. 1, e.g., Gaussian noise $X_0 \sim \mathcal{N}(0, I)$;

2. using an equivariant network u_θ for the velocity field
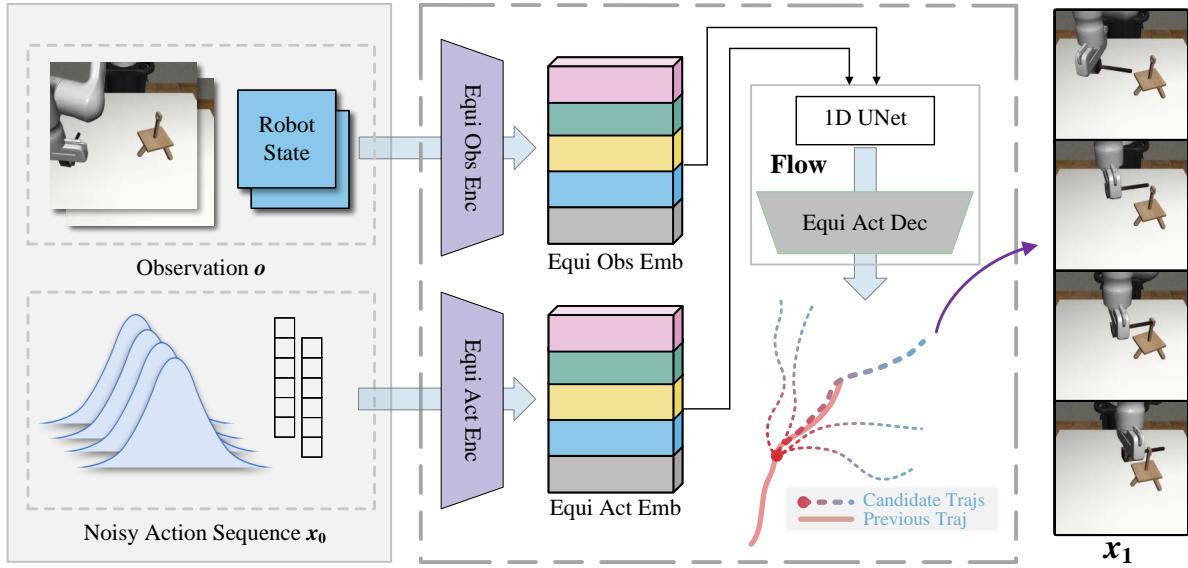
Figure 2: Overview of EfficientFlow. At each decision step, the policy utilizes the most recent two observation steps **o** as input. This information is processed by the equivariant Flow Matching network to generate five candidate action trajectories. The trajectory that exhibits the minimum Euclidean distance to the previously predicted trajectory is then selected for execution, ensuring a smooth and coherent action sequence.

such that:

$$u_\theta(t, gx|go) = g\left(u_\theta(t, x|o)\right), \quad \forall g \in G. \quad (6)$$

Importantly, we do *not* impose the strong assumption that the expert policy in the training data be equivariant, which is in sharp contrast with (Wang et al. 2024).

Theorem 1. *Let G be a transformation group acting on both the observation space and the action space. Suppose the initial distribution p_0 is isotropic, i.e., $p_0(gx) = p_0(x)$ for all $g \in G$, and the velocity network $u_\theta(t, x|o)$ is equivariant as in Eq. 6. Then the induced conditional distribution at time t, given by the flow ODE Eq. 1, satisfies*

$$X_t\big|_{O=go} \stackrel{d}{=} g\left(X_t\big|_{O=o}\right), \quad t \in [0, 1] \quad (7)$$

i.e., the output distribution is equivariant under the group action.

The special case $t = 1$ of Eq. 7 gives us the desired property in Eq. 5. An intuitive visualization of this result is provided in Figure 1(a). From a discrete-time perspective, consider starting from a randomly-sampled initial action $x_0 \sim p_0$. After a small time step Δt, the action evolves under the velocity field to reach $x_{\Delta t} = x_0 + \Delta t \cdot u_\theta(0, x_0|o)$. This corresponds to the green curve in Figure 1(a).

Now, consider a rotated scenario where the initial action is transformed to $\hat{x}_0 = gx_0$, and the observation is rotated accordingly to go. The updated action becomes

$$
\begin{aligned}
\hat{x}_{\Delta t} &= \hat{x}_0 + \Delta t \cdot u_\theta(0, \hat{x}_0|go) && \text{(one-step update)} \\
&= gx_0 + \Delta t \cdot u_\theta(0, gx_0|go) && \text{(since } \hat{x}_0 = gx_0\text{)} \\
&= gx_0 + \Delta t \cdot gu_\theta(0, x_0|o) && \text{(Eq. 6)} \\
&= gx_{\Delta t}.
\end{aligned}
$$

This corresponds to the blue curve in Figure 1(a), showing that the evolution of the rotated action $\hat{x}_t$ under the rotated observation go aligns with the rotated evolution of the original action x_t.

By repeating this process over the entire flow trajectory, we conclude that $\hat{x}_1 = gx_1$ (see Figure 1(a)). Since p_0 is isotropic, x_0 and $\hat{x}_0$ have the same probability density. Given that the flow deterministically transports x_0 to x_1 and $\hat{x}_0$ to $\hat{x}_1$, it follows that x_1 and $\hat{x}_1$ share the same density. This implies that the resulting distribution of x_1 and $\hat{x}_1$ respects the desired equivariance with respect to g. We emphasize that this is only an intuitive explanation; a rigorous proof is provided in the supplementary material.

Since standard Flow Matching uses Gaussian distribution as p_0 by default, the isotropy condition in Theorem 1 is automatically satisfied. As a result, making Flow Policy equivariant reduces to designing an equivariant network u_θ.

Design of the Equivariant u_θ. To implement the equivariant policy network u_θ, we leverage the escnn library (Cesa, Lang, and Weiler 2022), which supports constructing neural networks that are equivariant to symmetry groups (planar rotations modeled by SO(2) in our case). A critical step in using escnn is specifying how each output component transforms under group actions, which requires carefully choosing representations that respect the underlying task symmetries.

In our setting, the policy outputs an absolute 6-DoF end-effector pose with 3D rotation and 3D translation, along with a scalar gripper width to control a robot arm. To represent the 3D rotation, we adopt the 6D continuous representation (Zhou et al. 2019) that encodes the first two rows of a 3×3 rotation matrix, corresponding to the x and y axes.

This 6D representation can be seen as three 2D vectors in the x-y plane, which transform under SO(2) according to the irreducible representation ρ_1. Therefore, the 3D rotation component corresponds to ρ_1^3.

For 3D translation, the x and y components transform as a 2D vector under SO(2), again corresponding to ρ_1, while the z component remains invariant and is modeled as ρ_0. The scalar gripper width is also invariant under planar rotation, corresponding to another ρ_0.

Combininng these components, the action vector at robot time τ, denoted by x^τ, is a 10D vector comprising a 6D rotation representation (first 6 dimensions), a 3D translation vector (next 3 dimensions), and a scalar gripper width (final dimension). The corresponding equivariant representation of the action output is:

$$gx^\tau = (\rho_1^3 \oplus (\rho_1 \oplus \rho_0) \oplus \rho_0)(g)x^\tau.$$

This representation enables u_θ to produce actions that respect the SO(2) symmetry of the task, ensuring consistent behavior under planar rotations of the scene.

Network Architecture. As introduced above, the input of u_θ is flow time t, action sequence x_t, and observation o. We set the equivariant group as finite cyclic subgroup $C_u \in$ SO(2), and u is the order of the group. We first use an equivariant observation encoder to map observation o to embeddings $e_o \in \mathbb{R}^{u \times d_o}$ and use an equivariant action encoder to map action sequence x_t to embeddings $e_x \in \mathbb{R}^{u \times d_x}$, where d_o and d_x are the feature dimensions associated with each group element.

The encoded embeddings e_o, e_x, along with the timestep t, are fed into a core equivariant neural network. This network, together with the observation and action encoders, parameterize the conditional vector field $u_\theta(t, x_t, o)$. As all components are designed to be equivariant, the entire mapping process from raw inputs to the predicted vector field strictly adheres to C_u symmetry.

Temporal Consistency. When generating action sequences, adjacent segments are predicted independently. As a result, the policy may switch between different behavioral modes, leading to inconsistencies in long-term execution.

To address this, we adopt a temporal overlapping strategy similar to (Chi et al. 2023): only the first n_1 steps of each predicted sequence are executed, while the remaining $n - n_1$ steps overlap with the subsequent prediction starting from time $\tau + n_1$. Long-term consistency can be achieved by generating neighboring action sequences with similar overlap.

To this end, we employ a batched trajectory selection and periodic reset strategy, inspired by IMLE Policy (Rana et al. 2025), which balances multi-modal expressivity with temporal coherence. During inference, we sample m initial noise vectors $\{x_{0,i}\}_{i=1}^m$ from a Gaussian distribution and evolve each through our model to generate m candidate action trajectories $\{x_{1,i}\}_{i=1}^m$. We then select the trajectory whose overlapping segment best matches the previous trajectory in the Euclidean sense:

$$\arg \min_{i \in 1, \ldots, m} d\left([x_{\text{pre}}^{\tau+n_1}, \ldots, x_{\text{pre}}^{\tau+n}], [x_{1,i}^{\tau+n_1}, \ldots, x_{1,i}^{\tau+n}]\right),$$

where we assume the current robot time is $\tau + n_1$, and x_{pre} denotes the previous action sequence predicted at time τ, where the steps $x_{\text{pre}}^\tau, \ldots, x_{\text{pre}}^{\tau+n_1-1}$ have already been executed.

To preserve the model's ability to explore diverse behaviors, we introduce periodic resets: every 10 prediction cycles, we randomly select one trajectory from the batch for execution, instead of the one that minimizes the overlap distance. This approach improves temporal consistency while maintaining multi-modality, and the batched design ensures minimal overhead in inference time due to parallelization.

Acceleration Regularization

In our experiments, we observe that flow-based policies trained solely with the conditional flow matching objective (Eq. 3) tend to perform poorly when the number of function evaluations (NFE) is low. This suggests that the learned flow fields are overly curved, requiring more integration steps for accurate trajectory generation.

To address this, we propose an acceleration regularization term that encourages smoother, low-curvature flow trajectories. The underlying intuition is that smoother motion corresponds to smaller second-order derivatives (accelerations) of the trajectory x_t. In the extreme case of zero acceleration, the trajectory becomes a straight line.

We augment the training objective as follows:

$$\underbrace{\mathbb{E}\left[\|u_\theta(t, x_t) - u(t, x_t \mid x_1)\|_2^2\right]}_{\text{Data Term}} + \lambda \underbrace{\mathbb{E}\left[\left\|\frac{\mathrm{d}^2 x_t}{\mathrm{d}t^2}\right\|_2^2\right]}_{\text{Acceleration Penalty}}, \quad (8)$$

where λ controls the trade-off between fidelity to the target velocity field and trajectory smoothness. In practice, we use a time-dependent weighting $\lambda(t) = (1 - t)^2$, which encourages smoother flow at earlier timesteps and prioritizes accuracy as $t \to 1$.

According to Eq. 1, the second derivative term can be rewritten as:

$$\mathbb{E}\left[\left\|\frac{\mathrm{d}^2 x_t}{\mathrm{d}t^2}\right\|_2^2\right] \approx \frac{1}{(\Delta t)^2} \mathbb{E}\|u_\theta(t, x_t) - u_\theta(t + \Delta t, x_{t+\Delta t})\|_2^2,$$

which, however, cannot be directly evaluated, because x_t and $x_{t+\Delta t}$ lie on the same underlying marginal trajectory that is unknown.

To overcome this, we introduce a practical surrogate regularization, which we call the Flow Acceleration Upper Bound (FABO):

$$\mathbb{E}\|u_\theta(t, \tilde{x}_t) - u_\theta(t + \Delta t, \tilde{x}_{t+\Delta t})\|_2^2 \quad (9)$$
$$\geq \mathbb{E}\|u_\theta(t, x_t) - u_\theta(t + \Delta t, x_{t+\Delta t})\|_2^2,$$

when Δt is small. Notably, $\tilde{x}_t$ and $\tilde{x}_{t+\Delta t}$ are sampled from the same conditional trajectory at time t and $t + \Delta t$, which are easy to draw and require no knowledge of the marginal trajectory.

In essence, FABO minimizes an upper bound on the true acceleration penalty, serving as a tractable and effective proxy. A formal proof for Eq. 9 is provided in the supplementary material.

Method	Obs	NFE	Stack D1			Square D2			Threading D2			Stack Three D1			Coffee D2			3 Pc. Asm. D2		
			100	200	1000	100	200	1000	100	200	1000	100	200	1000	100	200	1000	100	200	1000
		1	**94**	**100**	**100**	21	**45**	67	**31**	36	49	48	73	92	65	**81**	79	11	35	60
Ours	RGB	3	88	**100**	**100**	20	**45**	71	**31**	**43**	53	49	76	94	66	80	**84**	11	38	69
		5	87	**100**	**100**	22	43	**71**	**31**	41	58	50	**79**	93	**67**	79	83	11	**42**	**71**
EquiDiff	RGB	100	93	**100**	**100**	**25**	41	60	22	40	**59**	55	77	**96**	60	79	76	**15**	39	69
DP-C	RGB	100	76	97	**100**	8	19	46	17	35	**59**	38	72	94	44	66	79	4	6	30
DP-T	RGB	100	51	83	99	5	11	45	11	18	41	17	41	84	47	61	75	1	4	43
DP3	PCD	10	69	87	99	7	6	19	12	23	40	7	23	65	34	45	69	0	1	3
ACT	RGB	1	35	73	96	6	18	49	10	21	35	6	37	78	19	33	64	0	3	24

Method	Obs	NFE	Hmmer Cln. D1			Mug Cln. D1			Kitchen D1			Pick Place D0			Nut Asmn.D0			Coffee Pre. D1		
			100	200	1000	100	200	1000	100	200	1000	100	200	1000	100	200	1000	100	200	1000
		1	**75**	**75**	77	**50**	65	67	66	78	81	37	50	67	59	83	94	75	74	70
Ours	RGB	3	72	**75**	84	**50**	65	70	**73**	81	81	41	62	86	61	86	96	81	81	**89**
		5	74	**75**	83	**50**	68	70	**73**	81	83	41	66	87	62	**87**	98	**83**	82	87
EquiDiff	RGB	100	65	63	77	**50**	64	67	67	77	81	**42**	**74**	92	**74**	85	94	77	**83**	85
DP-C	RGB	100	52	59	73	43	59	65	67	**85**	87	35	65	83	55	68	83	65	62	58
DP-T	RGB	100	48	60	76	30	43	63	54	75	81	15	37	50	31	32	46	38	51	76
DP3	PCD	10	54	71	87	21	33	53	45	71	**91**	12	15	34	16	24	58	10	22	63
ACT	RGB	1	38	54	71	23	31	56	37	61	87	7	17	50	42	64	84	32	46	65

Table 1: Comparison against SOTA. We report the success rates of 12 MimicGen (Mandlekar et al. 2023) tasks using 100, 200, and 1000 demonstrations, respectively. Results averaged over three seeds. The results of the baseline methods are directly cited from EquiDiff (Wang et al. 2024).

Experiments

Implementation Details

We evaluate EfficientFlow on 12 tasks from the MimicGen (Mandlekar et al. 2023) benchmark. These tasks span a wide range of difficulties, time horizons, and object arrangements, providing a comprehensive testbed for assessing policy performance across diverse robotic manipulation scenarios.

Notably, the agent view camera in MimicGen is not positioned orthogonally to the workspace but rather provides an agent-centric perspective. While the resulting image rotations may not perfectly align with the true object rotations, potentially impacting the performance of equivariant networks, this setup more closely mirrors real-world scenarios where state information is acquired from non-ideal viewpoints and thus offers a more rigorous test of policy generalization.

Our work aims for effective and efficient robot control in embodied AI. Thus, we compare EfficientFlow against strong baselines in this field, including EquiDiff (Wang et al. 2024), ACT (Zhao et al. 2023), and two variants of Diffusion Policy (Chi et al. 2023): the CNN-based DP-C and the Transformer-based DP-T. For fair comparison, all baseline methods use the same input as EfficientFlow: RGB images from both the agent-view and wrist-mounted cameras. In addition to the RGB-based methods, we further compare with DP3 (Ze et al. 2024), which utilizes 3D point cloud information. All policies employ absolute pose estimation for control. We evaluate EfficientFlow with 1, 3, and 5 number of function evaluations (NFE), while for baseline methods, we adhere to their original configurations.

Quantitative Comparison

Sampling Efficiency. EfficientFlow demonstrates notable advantages in inference speed and sampling efficiency. As shown in Table 2, it achieves an average inference time of only 12.22 ms under a 1-NFE setting, offering an approximately $56.1\times$ speedup over EquiDiff. Even with a more computationally demanding 5-NFE inference, EfficientFlow remains approximately $19.9\times$ faster than EquiDiff on average. This significant gain in efficiency is critical for real-time robotic control, where rapid response is essential, and enables inference frequencies of up to 81.8 Hz with single-step EfficientFlow.

Data Efficiency. As shown in Table 1, under the data-limited setting of only 100 demonstrations, EfficientFlow not only enables significantly faster inference but also achieves strong policy success rate. Across the 12 test tasks, EfficientFlow outperforms EquiDiff in 7 of them. For the remaining 5 tasks, with the exception of the Nut Assembly D0 task, the performance gap between EfficientFlow and EquiDiff is within 5 percentage points. These results indicate that EfficientFlow can match or even surpass the performance of the SOTA baseline method while drastically reducing inference latency. When trained with 200 demonstrations, EfficientFlow achieves 98.4% of the success rate of the DP-C method trained with 1000 demonstrations, while surpassing the average success rates of DP-T, DP3, and ACT. This remarkable performance underscores its exceptional data efficiency and strong generalization under limited supervision.

Method	NFE	Runtime (ms)	Average Success Rate (%)		
			100	200	1000
Ours	1	12.22	52.61	66.18	75.25
	3	22.59	53.49	**69.33**	**81.36**
	5	34.45	**54.18**	**70.26**	**81.99**
EquiDiff	100	685.92	53.77	68.59	79.69
DP-C	100	542.96	42.00	57.75	71.42
DP-T	100	497.53	29.00	43.00	64.92
DP3	10	53.83	23.92	35.08	56.75
ACT	1	12.51	21.33	38.17	63.25

Table 2: Average success rates and inference time of EfficientFlow and baselines across 12 MimicGen tasks.

Method	Sk	Sq	Th	S3	Cf	3P	Hm	Mu	Ki	PP	Nu	CP	Avg
Ours	**20**	60	**20**	**40**	**30**	**40**	**10**	30	**30**	**40**	40	**20**	**31.7**
EquiDiff	70	80	40	50	40	70	50	40	**30**	60	50	40	51.7
NoAcc	30	**40**	**20**	60	**30**	60	20	**10**	**30**	50	**30**	40	35.0

Table 3: Minimum training epochs required for EfficientFlow and EquiDiff to reach 50 % of their final maximum success rate (MimicGen, 100 demonstrations, evaluated every 10 epochs with a fixed seed).

More Analysis of the Performance. As shown in Table 2, the average success rates across all tasks further highlights the advantage of EfficientFlow. First, the average performance of EfficientFlow exhibits an upward trend as the NFE increases; with sufficient data, multi-step inference can better capture the conditional action distribution to achieve higher success rates. More importantly, across all dataset sizes, EfficientFlow consistently exceeds the average success rate of EquiDiff (Wang et al. 2024) while requiring dramatically fewer inference steps. We attribute the advantage of EfficientFlow to two key design choices: the strong inductive biases introduced by the equivariant architecture, and the acceleration regularization that stabilizes the action sampling trajectories. These factors enable the model to more efficiently learn key task structures and robust dynamics representations from limited demonstrations.

Learning Efficiency. To quantitatively evaluate the learning efficiency of EfficientFlow, we measure the minimum number of training epochs required to reach 50% of the final peak success rate. As shown in Table 3, both EfficientFlow and its variant without acceleration regularization (NoAcc) require substantially fewer training epochs than EquiDiff to reach 50% of their maximum success rate. Notably, in the Hammer Cleanup D1 task, EfficientFlow requires only one-fifth of the epochs needed by EquiDiff. These results demonstrate the improved learning efficiency and stronger optimization dynamics of our equivariant flow-based framework. The acceleration constraint further improves convergence speed, as evidenced by the faster learning of EfficientFlow compared to NoAcc. These results indicate that EfficientFlow can extract essential policy information from demonstrations more rapidly, which reflects its

Method	Sk	Sq	Th	S3	Cf	3P	Hm	Mu	Ki	PP	Nu	CP	Avg
Ours	94	21	**31**	**48**	**65**	**11**	**75**	**50**	66	**37**	**59**	**75**	**52.6**
NoAcc	88	16	24	44	56	10	56	28	42	16.5	28	62	39.3
NonEqui	88	12	12	14	52	0	66	38	54	17.5	43	56	37.7
EquiCFM	88	8	22	44	60	8	54	36	**66**	19	40	40	40.4
EquiMF	**96**	**22**	26	34	50	8	58	**50**	62	26	51	72	46.3

Table 4: Ablation study on 12 MimicGen tasks using 100 demonstrations.

superior learning dynamics, enabling it to reach target performance levels significantly faster than baseline methods.

Ablation Study

To disentangle the contributions of the equivariant architecture and the acceleration regularization, we conduct comprehensive ablation studies across 12 MimicGen tasks using 100 demonstrations in Table 4. We evaluate four key variants: 1) NoAcc, which removes the acceleration term and is trained solely with the Conditional Flow Matching loss (L_{CFM} in Eq. 3); 2) NoEqui, which discards the equivariant architecture in favor of a non-equivariant backbone similar to Diffusion Policy (Chi et al. 2023) while retaining the acceleration constraint; 3) EquiCFM, which combines the equivariant network with Consistency Flow Matching (Yang et al. 2024b); and 4) EquiMF, which integrates the equivariant network with MeanFlow (Geng et al. 2025).

The results reveal two clear trends. First, both our proposed components, including equivariance and acceleration regularization, substantially and independently improve performance. Removing either one leads to a consistent drop in success rate, demonstrating their complementary roles: the equivariant structure provides strong inductive biases for learning symmetric behaviors, while the acceleration term stabilizes sampling trajectory learning.

Second, to further investigate the benefit of our acceleration regularization, we compare EfficientFlow against other efficient one-step flow matching variants. By replacing our formulation with Consistency Flow Matching (Yang et al. 2024b) and MeanFlow (Geng et al. 2025) while keeping the equivariant architecture fixed, we observe that EfficientFlow achieves higher overall success rates across tasks. This suggests that our acceleration-regularized formulation not only stabilizes training but also leads to more accurate and robust policy generation.

Conclusion

We introduce EfficientFlow, a theory-grounded generative policy learning framework that effectively balances inference speed and data efficiency. By leveraging equivariant flow matching and acceleration regularization, this work provides a principled approach for learning robust visuomotor policies while ensuring strong generalization and efficient learning. The ultra-fast inference and strong data efficiency of EfficientFlow highlights its potential as a practical and high-performance solution for real-world embodied AI systems.

Acknowledgments

References

Bjorck, J.; Castañeda, F.; Cherniadev, N.; Da, X.; Ding, R.; Fan, L.; Fang, Y.; Fox, D.; Hu, F.; Huang, S.; et al. 2025. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*.

Black, K.; Brown, N.; Driess, D.; Esmail, A.; Equi, M.; Finn, C.; Fusai, N.; Groom, L.; Hausman, K.; Ichter, B.; et al. 2024. π_0: A Vision-Language-Action Flow Model for General Robot Control. *arXiv preprint arXiv:2410.24164*.

Cesa, G.; Lang, L.; and Weiler, M. 2022. A program to build E (N)-equivariant steerable CNNs. In *International Conference on Learning Representations*.

Chi, C.; Xu, Z.; Feng, S.; Cousineau, E.; Du, Y.; Burchfiel, B.; Tedrake, R.; and Song, S. 2023. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*.

Gao, D.; Zhao, B.; Lee, A.; Chuang, I.; Zhou, H.; Wang, H.; Zhao, Z.; Zhang, J.; and Soltani, I. 2025. VITA: Vision-to-Action Flow Matching Policy. *arXiv preprint arXiv:2507.13231*.

Geng, Z.; Deng, M.; Bai, X.; Kolter, J. Z.; and He, K. 2025. Mean flows for one-step generative modeling. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33.

Huang, H.; Howell, O.; Zhu, X.; Wang, D.; Walters, R.; and Platt, R. 2024. Fourier Transporter: Bi-Equivariant Robotic Manipulation in 3D. *CoRR*.

Huang, H.; Wang, D.; Walters, R.; and Platt, R. 2022. Equivariant Transporter Network. In *Robotics: Science and Systems*.

Huang, H.; Wang, D.; Zhu, X.; Walters, R.; and Platt, R. 2023. Edge grasp network: A graph-based se (3)-invariant approach to grasp detection. In *2023 IEEE International Conference on Robotics and Automation*.

Jia, M.; Wang, D.; Su, G.; Klee, D.; Zhu, X.; Walters, R.; and Platt, R. 2023. Seil: Simulation-augmented equivariant imitation learning. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*.

Kim, S.; Lim, B.; Lee, Y.; and Park, F. C. 2023. Se (2)-equivariant pushing dynamics models for tabletop object manipulations. In *Conference on Robot Learning*.

Lipman, Y.; Chen, R. T.; Ben-Hamu, H.; Nickel, M.; and Le, M. 2022. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*.

Liu, S.; Xu, M.; Huang, P.; Zhang, X.; Liu, Y.; Oguchi, K.; and Zhao, D. 2023. Continual vision-based reinforcement learning with group symmetries. In *Conference on Robot Learning*.

Mandlekar, A.; Nasiriany, S.; Wen, B.; Akinola, I.; Narang, Y.; Fan, L.; Zhu, Y.; and Fox, D. 2023. Mimicgen: A data generation system for scalable robot learning using human demonstrations. *arXiv preprint arXiv:2310.17596*.

Nguyen, H. H.; Baisero, A.; Klee, D.; Wang, D.; Platt, R.; and Amato, C. 2023. Equivariant reinforcement learning under partial observability. In *Conference on Robot Learning*.

Pan, C.; Okorn, B.; Zhang, H.; Eisner, B.; and Held, D. 2023. Tax-pose: Task-specific cross-pose estimation for robot manipulation. In *Conference on Robot Learning*.

Rana, K.; Lee, R.; Pershouse, D.; and Suenderhauf, N. 2025. IMLE Policy: Fast and Sample Efficient Visuomotor Policy Learning via Implicit Maximum Likelihood Estimation. In *Proceedings of Robotics: Science and Systems (RSS)*.

Reuss, M.; Zhou, H.; Rühle, M.; Yağmurlu, Ö. E.; Otto, F.; and Lioutikov, R. 2025. FLOWER: Democratizing Generalist Robot Policies with Efficient Vision-Language-Action Flow Policies. In *7th Robot Learning Workshop: Towards Robots with Human-Level Abilities*.

Ryu, H.; Lee, H.-i.; Lee, J.-H.; and Choi, J. 2022. Equivariant Descriptor Fields: SE (3)-Equivariant Energy-Based Models for End-to-End Visual Robotic Manipulation Learning. In *The Eleventh International Conference on Learning Representations*.

Sheng, J.; Wang, Z.; Li, P.; and Liu, M. 2025. MP1: Mean Flow Tames Policy Learning in 1-step for Robotic Manipulation. *arXiv preprint arXiv:2507.10543*.

Simeonov, A.; Du, Y.; Lin, Y.-C.; Garcia, A. R.; Kaelbling, L. P.; Lozano-Pérez, T.; and Agrawal, P. 2023. Se (3)-equivariant relational rearrangement with neural descriptor fields. In *Conference on Robot Learning*.

Simeonov, A.; Du, Y.; Tagliasacchi, A.; Tenenbaum, J. B.; Rodriguez, A.; Agrawal, P.; and Sitzmann, V. 2022. Neural Descriptor Fields: SE (3)-Equivariant Object Representations for Manipulation. In *2022 IEEE International Conference on Robotics and Automation (ICRA)*.

Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; and Ganguli, S. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*.

Wang, D.; Hart, S.; Surovik, D.; Kelestemur, T.; Huang, H.; Zhao, H.; Yeatman, M.; Wang, J.; Walters, R.; and Platt, R. 2024. Equivariant Diffusion Policy. In *Conference on Robot Learning (CoRL)*.

Wang, D.; Jia, M.; Zhu, X.; Walters, R.; and Platt, R. 2022a. On-Robot Learning With Equivariant Models. In *6th Annual Conference on Robot Learning*.

Wang, D.; Walters, R.; and Platt, R. 2022. SO(2)-Equivariant Reinforcement Learning. In *International Conference on Learning Representations*.

Wang, D.; Walters, R.; Zhu, X.; and Platt, R. 2022b. Equivariant q learning in spatial action spaces. In *Conference on Robot Learning*.

Yang, J.; Cao, Z.; Deng, C.; Antonova, R.; Song, S.; and Bohg, J. 2024a. EquiBot: SIM (3)-Equivariant Diffusion Policy for Generalizable and Data Efficient Learning. In

CoRL 2024 Workshop on Whole-body Control and Bimanual Manipulation: Applications in Humanoids and Beyond.

Yang, L.; Zhang, Z.; Zhang, Z.; Liu, X.; Xu, M.; Zhang, W.; Meng, C.; Ermon, S.; and Cui, B. 2024b. Consistency flow matching: Defining straight flows with velocity consistency. *arXiv preprint arXiv:2407.02398.*

Ze, Y.; Zhang, G.; Zhang, K.; Hu, C.; Wang, M.; and Xu, H. 2024. 3D Diffusion Policy: Generalizable Visuomotor Policy Learning via Simple 3D Representations. In *Proceedings of Robotics: Science and Systems (RSS).*

Zhang, Q.; Liu, Z.; Fan, H.; Liu, G.; Zeng, B.; and Liu, S. 2025. 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.*

Zhao, T.; Kumar, V.; Levine, S.; and Finn, C. 2023. Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware. *Robotics: Science and Systems XIX.*

Zhou, Y.; Barnes, C.; Lu, J.; Yang, J.; and Li, H. 2019. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.*

Zhu, X.; and Wang, D. 2022. Sample Efficient Grasp Learning Using Equivariant Models. In *Robotics science and systems.*

Zhu, X.; Wang, D.; Su, G.; Biza, O.; Walters, R.; and Platt, R. 2023. On robot grasp learning using equivariant models. *Autonomous Robots.*