Efficient Hybrid SE(3)-Equivariant Visuomotor Flow Policy via Spherical Harmonics for Robot Manipulation

Supplementary Material

A. Real-World Robot Experiments Details

Real-World Robot Experiments Details. We evaluate the effectiveness of E3Flow on 4 real-world physical manipulation tasks: Storing Toy, Bottle Place, Stack Blocks, and Assembly. The details of the tasks are described in Sec. B. Fig. A.1 shows our real-world robotic experimental setup, which consists of two RealSense D435 RGB-D cameras (a head camera and a hand camera), a 6-DOF PIPER robotic arm equipped with a two-finger gripper, and a white tabletop workspace. For each task, we collected 50 expert demonstrations using a teleoperation (puppet-master) setup, recording 3D information from the head camera and RGB image observations from the hand camera. The image observations from the hand camera have a resolution of 480×640, and the head camera point cloud contains 1024 points after cropping. We trained SDP [46], EquiDiff [32], and DP [2] as baseline methods for comparison with E3Flow, and evaluated each task over 20 rollouts. Table B.1 reports the success rates for all tasks. The results show that E3Flow achieves the highest average success rate.

Results Analysis. We further analyzed the failure modes of each method across tasks. DP lacks orientation-awareness, resulting in poor performance on tasks that are highly sensitive to object orientation. EquiDiff achieves relatively high success on the Assembly task, but still exhibits limitations when handling tasks with significant SE(3) variation. The primary failure mode of SDP is inaccurate object localization, which leads to failed grasps or incorrect placements, an issue largely attributed to occlusions present in single-view point clouds. In contrast, E3Flow effectively mitigates point cloud occlusions by leveraging heterogeneous visual modalities, outperforming the strongest baseline, SDP. Moreover, compared with EquiDiff and DP, the continuous SE(3) representation in E3Flow demonstrates stronger robustness in real-world environments. Overall, the real-world experiments further validate the applicability and effectiveness of E3Flow.

B. Task Details

The visualization of the real-world robot task execution process are illustrated in Fig. B.2.

Storing Toys. A toy is placed on an inclined plane, and the robot must pick it up from the slope and place it into a bowl on the flat tabletop. The incline angle varies between 15° and 60°, and the in-plane positions of both the toy and the slope are randomly initialized within a 10 cm range. In addition, the toy is randomly rotated around the slope surface to simulate SE(3) variations. Successfully completing the Storing Toys task requires the robot to extract rich orientation information.

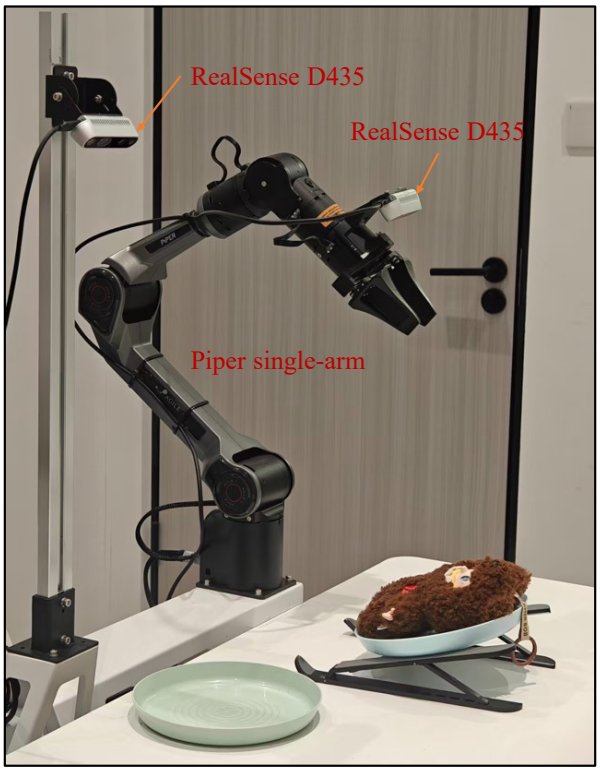
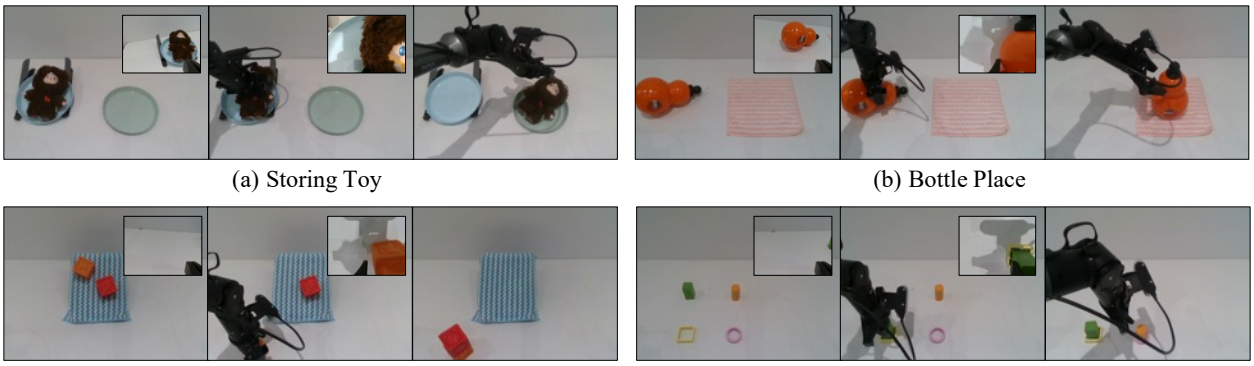


Figure A.1. The overall setup of the real-world robotic experiment platform includes two RealSense D435 RGB-D cameras, a 6-DOF PIPER robotic arm equipped with a two-finger gripper, and a tabletop workspace.

Bottle Place. An irregular-shaped bottle is placed on the tabletop with a random SE(3) orientation. The robot must pick up the bottle and place it upright on a mat, whose position is randomly initialized within a 10 cm range on the tabletop. Placing the bottle upside down on the mat is not counted as a success.

Stack Blocks. Two blocks of different colors are randomly placed on a blue inclined plane, whose tilt angle varies between 15° and 60°. The robot must first pick



(a) Storing Toy (b) Bottle Place

(c) Stack Blocks (d) Assembly

Figure B.2. Visualization of the execution process for the real-world robotic tasks.

Table B.1. Success rates of different methods on various tasks.

Method	Storing Toys	Bottle Place	Stack Blocks	Assembly	Average SR (%)
E3Flow (ours)	70	80	95	60	76
SDP [46]	60	65	85	40	62
EquiDiff [32]	5	25	45	45	30
DP [2]	0	10	35	20	16

up one block and place it on the tabletop, then pick up the second block and stack it on top of the first block to complete the task. The initial positions of the blocks are randomly sampled within the inclined plane, and the stacking location can be anywhere on the tabletop.

Assembly. A cylinder and a cuboid are placed on the tabletop, along with a circular slot and a square slot that correspond to them, respectively. The robot must insert the cylinder into the circular slot and place the cuboid into the square slot, simulating a two-step assembly task. The cylinder and cuboid are randomly positioned on the left and right sides of the tabletop, and the task is considered successful only when both assembly actions are completed.

C. More Implementation Details

Hyperparameters To ensure the reproducibility of our experiments, we provide additional experimental details in Table C.2, including the hyperparameter settings for each component. Most of the configurations follow those used in SDP [46]. In addition, we run three random seeds for each task. The reported success rate is the average of the highest success rates obtained across the three seeds. It is worth noting that, for each method, we perform an evaluation every 20 epochs. Each evaluation consists of 50 episodes, and each episode share identical initial scenarios to guarantee fairness and to justify the selection of the maximum success rate. This evaluation protocol is identical to that of SDP [46].

Network Details Table C.3 presents more details of the network. For the visual encoders, we follow the configurations used in SDP [46] and EquiDiff [32]. Specifically, we use ResNet-18 to extract features from the eye-in-hand RGB images, and EquiformerV2 to extract equivariant point-cloud features. The feature enhancement module (FEM) is applied only to Type-0 (scalar) features, leaving Type-1 (vector) features and Type-2 (higher-order equivariant) features unchanged.

For the Equi-Flow U-Net, we adopt the Spherical Denoising Temporal U-Net (SDTU) proposed by Zhu [46]. SDTU is a 1D U-Net constructed in the spherical Fourier domain with spatio-temporal equivariance. Temporal equivariance is achieved via 1D convolutions along the time dimension t, while spatial (SO(3)) equivariance is ensured by performing channel mixing temporal convolutions independently for each spherical harmonic degree Type-ℓ (i.e. independently for each irreducible representation), so that the SO(3) action does not mix different ℓ subspaces. Concretely, the spherical Fourier temporal convolution can be written as

$$\tilde{h}^{o}_{\ell,m,t} = \sum_{j=0}^{R} \sum_{i \in \mathcal{I}} \tilde{h}^{i}_{\ell,m,\,t-j}\, \omega^{i \to o}_{\ell,\,j}, \tag{12}$$

where i and o index input and output feature channels,

Table C.2. Hyperparameter settings for different methods.

	E3Flow	SDP(DDPM)	SDP(DDIM)	EquiDiff	EquiBot	DP	DP3
Batch Size	64	64	64	128	128	128	128
Image Size	84	-	-	-	84	-	-
Point Clouds	1024	1024	1024	1024	1024	1024	1024
Prediction Horizon	16	16	16	16	16	16	16
Action Horizon	8	8	8	8	8	8	8
Learning Rate	1e-4	1e-4	1e-4	1e-4	1e-4	1e-4	1e-4
Epochs	500	500	500	500	500	500	500
Learning Rate Scheduler	cosine	cosine	cosine	cosine	cosine	cosine	cosine
Noise Scheduler	-	DDPM	DDIM	DDPM	DDPM	DDPM	DDIM
Inference Step	10	100	10	100	100	100	100
Visual Encoded Dimension	128	128	128	128	128	128	64

Table C.3. Network implementation Details.

Module	Details
Image Encoder	ResNet-18
Point Cloud Encoder	5-layer ResNet + EquiformerV2
Point Cloud Features	128-dim
FEM Output	128-dim
Equi-Flow U-Net	SDTU w/ spherical Fourier conv
Equivariant Layers	Equivariant Linear + FiLM

and the pair (ℓ, m) denotes the degree and order of the spherical Fourier component $\tilde{h}$. The set $\mathcal{I}$ contains all input channels, and j indexes the temporal lag up to R. Importantly, the learnable weights $\omega_{\ell,\,j}^{i\to o}$ are independent of m; this independence is required for SO(3)-equivariance. By Schur's lemma, any linear operator that commutes with the SO(3) group action must act as a scalar multiple of the identity on each irreducible subspace. Therefore, the above convolution acting only within each ℓ subspace with weights independent of m is SO(3)-equivariant.

Accordingly, we extend the feature modulation layer into an equivariant FiLM layer using equivariant linear layers, and apply conditional modulation to each feature type separately, ensuring equivariance throughout the entire flow matching process.

$$\mathrm{EFiLM}(h_\ell \mid \gamma_\ell, \beta_\ell) = \left(\gamma_\ell^\top h_\ell \, \frac{h_\ell}{\|h_\ell\|} + \beta_\ell\right) \tag{13}$$

The key property of the equivariant FiLM layer is that applying the rotation before EFiLM produces the same result as applying EFiLM before the rotation. This can also be established using Schur's lemma. For a group element g and an ℓ-type feature h_ℓ, we examine the EFiLM operation under the action of the Wigner D-matrix:

$$\mathrm{EFiLM}(D_\ell(g)h_\ell \,|\, D_\ell(g)\gamma_\ell,\ D_\ell(g)\beta_\ell) \tag{14}$$

$$= (D_\ell(g)\gamma_\ell)^\top D_\ell(g)h_\ell \, \frac{D_\ell(g)h_\ell}{\|D_\ell(g)h_\ell\|} + D_\ell(g)\beta_\ell \tag{15}$$

$$= \gamma_\ell^\top D_\ell(g)^\top D_\ell(g)h_\ell \, \frac{D_\ell(g)h_\ell}{\|D_\ell(g)h_\ell\|} + D_\ell(g)\beta_\ell. \tag{16}$$

Using the orthogonality of Wigner D-matrices, $\|D_\ell(g)h_\ell\| = \|h_\ell\|$, we have

$$\mathrm{EFiLM}(D_\ell(g)h_\ell \,|\, D_\ell(g)\gamma_\ell,\ D_\ell(g)\beta_\ell) \tag{17}$$

$$= \gamma_\ell^\top h_\ell \, \frac{D_\ell(g)h_\ell}{\|h_\ell\|} + D_\ell(g)\beta_\ell \tag{18}$$

$$= D_\ell(g)\left(\gamma_\ell^\top h_\ell \, \frac{h_\ell}{\|h_\ell\|} + \beta_\ell\right) \tag{19}$$

$$= D_\ell(g) \cdot \mathrm{EFiLM}(h_\ell \mid \gamma_\ell, \beta_\ell). \tag{20}$$

This final line follows directly from Schur's lemma. Overall, the above derivation demonstrates that the Equi-Flow U-Net is equivariant with respect to both its inputs and outputs, making it applicable to flow matching processes rather than being limited solely to diffusion-based approaches.

D. End-to-End Symmetry Analysis

We analyze the equivariant properties of E3Flow. First, the point cloud encoder is SO(3)-equivariant, extracting spherical features including Type-0 scalar features, Type-1 vector features, and Type-2 higher-order tensor features, with Type-1 and Type-2 features being equivariant. The image encoder is not equivariant and extracts visual detail features that indicate potential contact regions. Although the feature enhancement module (FEM) is not equivariant, it only injects visual detail

features into type-0 features, thus preserving the equivariant part of the spherical harmonic features. Consequently, the output of the visual encoder remains equivariant. Furthermore, Sec. C provides a detailed analysis showing that the Equi-Flow U-Net is SO(3)-equivariant. Therefore, the vector field predicted by the network is equivariant. In summary, E3Flow is an end-to-end SE(3)-equivariant model, with translational equivariance realized through coordinate normalization.