Appendix for "Rethinking Bimanual Robotic Manipulation: Learning with Decoupled Interaction Framework"

1. Video Demo

A video demo is provided for both simulation and real-world manipulation experiments using our Decoupled Interaction Framework, showing the **effectiveness**, **extensibility**, and **practical applicability** of our framework. In simulation experiments, we first collect 50 demonstrations for various tasks in the Sapien simulation environment. We then train our framework separately for each task. For real-world experiments, we utilize the Cobot Magic robotic arm and the RealSense L515 camera, employing single-view, third-person point clouds for model training and inference. Watch the video for more details. Enjoy and have Fun!

2. Analysis of Selective Interaction Module

We further explore various interaction design approaches. Specifically, we evaluate concatenation and MLP, which are widely adopted in the Computer Vision community for interaction modeling. The results are presented in Tab. 1. As shown in Tab. 1, compared to the MLP and concatenation-based interaction modeling, our model achieves a performance improvement of at least 5.3%, demonstrating the effectiveness of our selective interaction module.

3. Visualization of the Generated Manipulation Trajectories in RoboTwin

In this section, we visualize some manipulation trajectories in RoboTwin [4] generated by our Decoupled Interaction Framework. As illustrated in Fig. 1, it can be concluded that: (1) We visualize two manipulation trajectories with different target objects in the "Diverse Bottles Pick" task to demonstrate the capability of our framework for **intra-class generalization**. (2) Additionally, we visualize trajectories for the "Block Hammer Beat" and "Empty Cup Place" tasks with varying initial positions of the target objects. As shown in Fig. 1, our model autonomously decides which arm to use based on the position of the target object, demonstrating its **decision-making** ability.

4. Visualization of the Generated Manipulation Trajectories in Real World

In this section, we visualize some manipulation trajectories in real world generated by our Decoupled Interaction Framework. As illustrated in 2, our framework effectively handles both coordinated and uncoordinated tasks. This is because our decoupled design effectively reduces the high-dimensional action space, enabling the network to learn actions more efficiently. Additionally, our selective interaction module explicitly models the interaction between the arms, allowing our framework to better meet the cooperation requirements of various bimanual manipulation tasks.

5. Implementation Details

In this section, we provide a detailed introduction to the implementation details of all baselines and our Decoupled Interaction Framework.

Training Setup. The key training setup for our Decoupled Interaction Framework based on the DDIM [5] and Flow Matching [2] is detailed in Tab. 2.

Baseline Setups. We also outline the training settings for the baseline in Tab. 3. Because of the differences in hyper-parameters between ACT and other baselines, we provide a description of its hyper-parameters here. For the ACT method, we use the AdamW optimizer with an initial learning rate of 1.0e-5 and a weight decay of 1.0e-4. The training process employs a batch size of 8, runs for 2000 epochs and uses an action chunking size of 100.

6. Simulation Tasks

We also visualize the distinct manipulation tasks from the RoboTwin benchmark [4], as illustrated in Fig. 3, and provide detailed descriptions of all simulation tasks in Tab. 4, totaling seven tasks.

References

[1] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Robotics: Science and Systems*, 2023. 2

[2] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. 1, 2

[3] I-Chun Arthur Liu, Sicheng He, Daniel Seita, and Gaurav S. Sukhatme. Voxact-b: Voxel-based acting and stabilizing policy for bimanual manipulation. In *Conference on Robot Learning*, 2024. 2

[4] Yao Mu, Tianxing Chen, Shijia Peng, Zanxin Chen, Zeyu Gao, Yude Zou, Lunkai Lin, Zhiqiang Xie, and Ping Luo. Robotwin: Dual-arm robot benchmark with generative digital twins (early version). *arXiv preprint arXiv:2409.02920*, 2024. 1

[5] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 1, 2

[6] Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. In *Robotics: Science and Systems*, 2024. 2

	Coordinated		Uncoordinated					Average
	Block Handover	Blocks Stack Easy	Dual Bottles Pick Easy	Dual Bottles Pick Hard	Diverse Bottles Pick	Empty Cup Place	Block Hammer Beat	
MLP	1.000	0.330	0.960	0.550	0.590	0.860	0.860	0.736 (↓ 0.053)
Concat	1.000	0.370	0.960	0.510	0.590	0.810	0.890	0.733 (↓ 0.056)
Ours	**1.000**	**0.400**	**0.990**	**0.630**	**0.700**	**0.900**	**0.900**	**0.789**

Table 1. **Illustration of the performance with different interaction designs on seven tasks in the RoboTwin dataset.** Best results are highlighted in bold. Important comparison metrics are marked with gray cells. The red arrows indicate the performance difference between each baseline and our method.

Parameter	Ours (DDIM[5])	Ours (Flow Matching[2])
horizon	8	8
n_obs_steps	3	3
n_action_steps	6	6
num_inference_steps	10	10
dataloader.batch_size	120	120
dataloader.num_workers	8	8
dataloader.shuffle	True	True
dataloader.pin_memory	True	True
dataloader.persistent_workers	False	False
optimizer._target_	torch.optim.AdamW	torch.optim.AdamW
optimizer.lr	1.0e-4	3.0e-5
optimizer.betas	[0.95, 0.999]	[0.95, 0.999]
optimizer.eps	1.0e-8	1.0e-8
optimizer.weight_decay	1.0e-6	1.0e-6
training.lr_scheduler	cosine	cosine
training.lr_warmup_steps	500	10
training.num_epochs	3000	3000
training.gradient_accumulate_every	1	1
training.use_ema	True	True

Table 2. **Model training settings.** Hyper-parameter Settings for Training and Deployment of our Decoupled Interaction Framework.

Parameter	DP [1]	DP3 [6]	Voxact-b [3]
horizon	8	8	8
n_obs_steps	3	3	3
n_action_steps	6	6	6
num_inference_steps	100	10	10
dataloader.batch_size	128	256	256
dataloader.num_workers	0	8	8
dataloader.shuffle	True	True	True
dataloader.pin_memory	True	True	True
dataloader.persistent_workers	False	False	False
optimizer._target_	torch.optim.AdamW	torch.optim.AdamW	torch.optim.AdamW
optimizer.lr	1.0e-4	1.0e-4	1.0e-4
optimizer.betas	[0.95, 0.999]	[0.95, 0.999]	[0.95, 0.999]
optimizer.eps	1.0e-8	1.0e-8	1.0e-8
optimizer.weight_decay	1.0e-6	1.0e-6	1.0e-6
training.lr_scheduler	cosine	cosine	cosine
training.lr_warmup_steps	500	500	500
training.num_epochs	300	3000	3000
training.gradient_accumulate_every	1	1	1
training.use_ema	True	True	True

Table 3. **Baselines settings.** Hyper-parameter Settings for Training and Deployment of DP, DP3 and Voxact-b Algorithms.

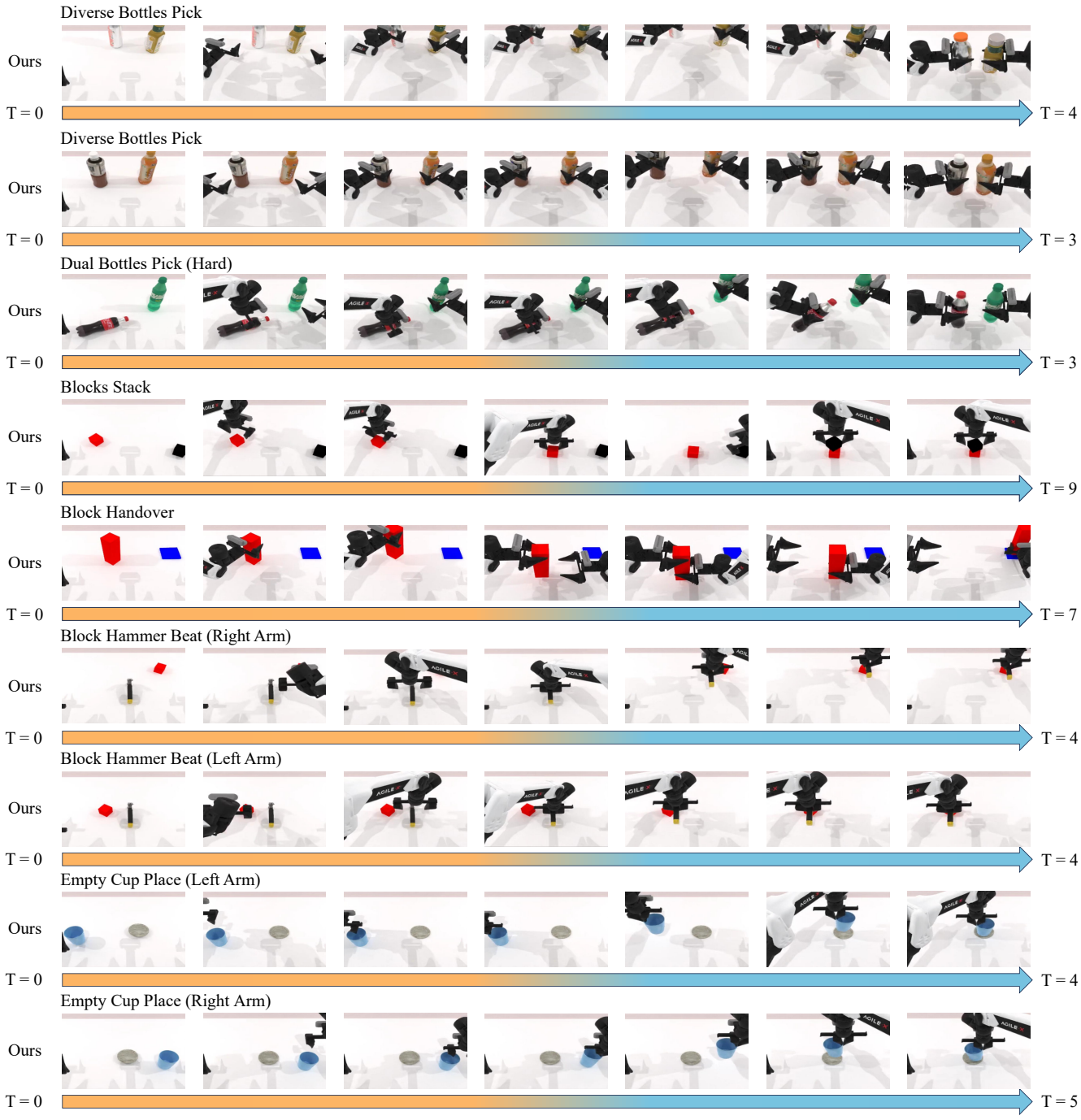

Figure 1. **Visualization of the generated manipulation trajectories of our framework in RoboTwin.** We visualize different coordinated and uncoordinated tasks within various scenes. Zoom in for best view.

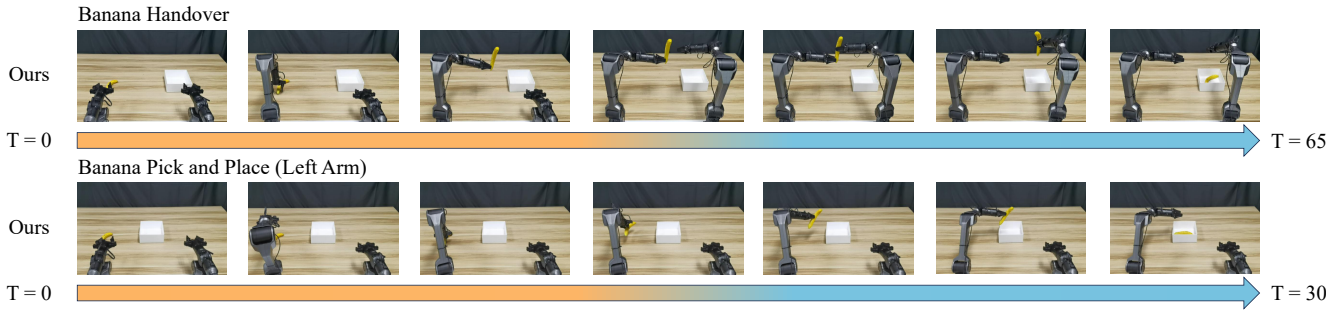

Figure 2. **Visualization of the generated manipulation trajectories of our framework in real-world experiments.** We visualize coordinated and uncoordinated tasks under different configurations. Zoom in for the best view.

Task	**Description**
Block Handover	A long block is placed on the left side of the table. The left arm grasps the upper side of the block and then hands it over to the right arm, which places the block on the blue mat on the right side of the table.
Blocks Stack Easy	Red and black cubes are placed randomly on the table. The robotic arm stacks the cubes in order, placing the red cubes first, followed by the black cubes, in the designated target location.
Dual Bottles Pick Easy	A red bottle is placed randomly on the left side, and a green bottle is placed randomly on the right side of the table. Both bottles are standing upright. The left and right arms are used simultaneously to lift the two bottles to a designated location.
Dual Bottles Pick Hard	A red bottle is placed randomly on the left side, and a green bottle is placed randomly on the right side of the table. The bottles' postures are random. Both left and right arms are used simultaneously to lift the two bottles to a designated location.
Diverse Bottles Pick	A random bottle is placed on the left and right sides of the table. The bottles' designs are random and do not repeat in the training and testing sets. Both left and right arms are used to lift the two bottles to a designated location.
Empty Cup Place	An empty cup and a cup mat are placed randomly on the left or right side of the table. The robotic arm places the empty cup on the cup mat.
Block Hammer Beat	There is a hammer and a block in the middle of the table. If the block is closer to the left robotic arm, it uses the left arm to pick up the hammer and strike the block; otherwise, it does the opposite.

Table 4. **Task descriptions for RoboTwin platform.**

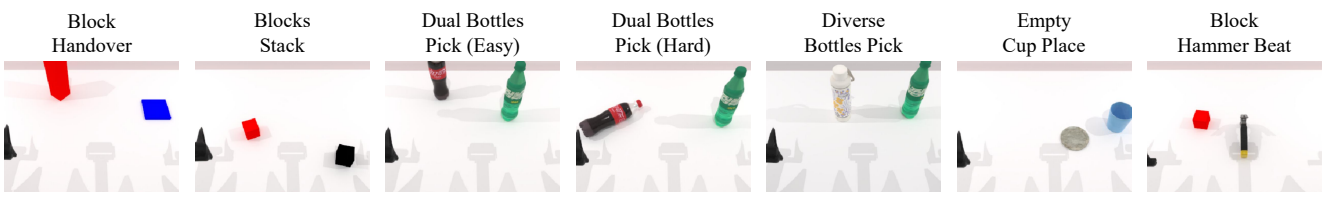

Figure 3. **Seven testing benchmark tasks.** We visualize manipulation tasks used in the RoboTwin benchmark. Zoom in for the best view.