

# Supplementary Material

Anonymous Author(s)

Affiliation

Address

email

## 1 A Website

2 Video results are available at <https://play-fusion.github.io>.

## 3 B Experimental Setup

4 We evaluate our method on three simulated environments. Below, we provide their details.

5 **CALVIN [1].** The CALVIN benchmark tests a robotic agent’s ability to follow language instructions.  
6 CALVIN contains four manipulation environments, each of which include a desk with a sliding  
7 door and a drawer that can be opened and closed, as well as a 7-DOF Franka Emika Panda robot  
8 arm with a parallel gripper. The four environments differ from each other in both their spatial  
9 composition (e.g., positions of drawers, doors, and objects) and visual features. The training data for  
10 each environment contains around 200K trajectories, from which we sample a sequence of transitions  
11 for each element of the minibatch. A portion of the dataset contains language annotations; we use  
12 this subset to train our language-conditioned model. Each transition consists of the RGB image  
13 observation, proprioceptive state information, and the 7-dimensional action. The agent is evaluated  
14 on its success rate in completing 34 tasks, which include variations of rotation, sliding, open/close,  
15 and lifting. These are specified by language instructions that are unseen during training in order  
16 to test the generalization ability of the agent. We evaluate on two setups: (1) CALVIN A, where  
17 the model is trained and tested on the same environment (called D→D in the benchmark) and (2)  
18 CALVIN B, where the model is trained on three of the four environments and tested on the fourth  
19 (called ABC→D in the benchmark).

20 **Franka Kitchen [2].** Franka Kitchen is a simulated kitchen environment with a Franka Panda robot.  
21 It contains seven possible tasks: opening a sliding cabinet, opening a hinge cabinet, sliding a kettle,  
22 turning on a switch, turning on the bottom burner, turning on the top burner, and opening a microwave  
23 door. The dataset contains 566 VR demonstrations of humans performing four of the seven tasks in  
24 sequence. Each transition consists of the RGB image observation, proprioceptive state information,  
25 and the 9-dimensional action. We split each of these demonstrations into their four tasks and annotate  
26 them with diverse natural language to create a language-annotated play dataset. In our experiments,  
27 we evaluate agents on two setups within this environment, which we denote as Kitchen A and Kitchen  
28 B. In Kitchen A, we evaluate an agent’s language generalization ability at test-time by prompting  
29 it with unseen instructions asking it to perform one of the seven tasks. This requires the model to  
30 identify the desired task and successfully execute it. Kitchen B is a more challenging evaluation  
31 setting, where the agent must perform two of the desired seven tasks in sequence given an unseen  
32 language instruction. In this setting, the agent must exhibit long-horizon reasoning capabilities and  
33 perform temporally consistent actions, in addition to the language generalization required in Kitchen  
34 A.

35 **Language-Conditioned Ravens [3, 4].** Ravens is a tabletop manipulation environment with a  
36 Franka Panda arm. We evaluate on two tasks in the Ravens benchmark: stacking blocks to form a

pyramid and putting blocks in bowls. The dataset consists of 1000 demonstrations collected by an expert policy. Although the dataset proposed in [4] contains language instructions denoting which color block to move and the desired final location, they are not diverse like human natural language annotations would be. In order to study our model’s performance on a play-like language-annotated dataset, we instead annotate the demonstrations with diverse natural language. At test-time, we prompt the agent with an unseen language instruction, similar to our other setups.

## B.1 Real World Setup

We create multiple play environments in the real world as well. We use a 7-DOF Franka Emika Panda robot arm with a parallel gripper, operating in joint action space. We have three different environments cooking, dining table and sink. All of these tasks are multi-step, i.e., in each the robot has to at least grab one object and put it in another, i.e. grab a carrot and put it inside the oven. In cooking, we test how the robot can handle articulated objects. It has to first open the oven, grill or pot, and then place an object properly inside. All of these objects have different articulations. Each of the placed objects (bread, carrot, knife, steak, spoon, etc.) have unique and different ways of being interacted with. In the sink, we test very precise manipulation skills, where the robot has to place objects in the narrow dish rack or hang objects (like mugs). In all of these settings, we test unseen goals (a combination of objects) that has never been seen before, as well as an instruction that has never been seen before. We provide more details in the Appendix.

## B.2 Implementation Details

Table 1 shows the main hyperparameters of our model in our simulation and real world experiments. We build off of the implementation of MCIL from CALVIN [5]. For Franka Kitchen and Ravens dataset and environment processing, we use implementations from [6] and [7], respectively. For implementations of the baselines, we modify [8] for C-BeT and [5] for Play-LMP and GCBC. Where possible, we use the same hyperparameters for PlayFusion and the baselines.

Table 1: Hyperparameters of PlayFusion in our simulation and real-world experiments.

Hyperparameter	CALVIN	Franka Kitchen	Ravens	Real World
Batch size	32	32	128	12
Codebook size	2048	2048	2048	2048
U-Net discretiz. wgt	0.5	0.5	0.5	0.5
Lang. discretiz. wgt	0.5	0.5	0.5	0.5
Action horizon $T_a$	16	64	2	32
Context length $T_o$	2	1	1	1
Language features	384	384	384	384
Learning rate	1e-4	2.5e-4	2.5e-4	2.5e-4
Diffusion timesteps	50	50	50	50
Beta scheduler	squaredcos_cap_v2	squaredcos_cap_v2	squaredcos_cap_v2	squaredcos_cap_v2
Timestep embed dim	256	256	128	256

## References

- [1] O. Mees, L. Hermann, E. Rosete-Beas, and W. Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022.
- [2] A. Gupta, V. Kumar, C. Lynch, S. Levine, and K. Hausman. Relay policy learning: Solving long-horizon tasks via imitation and reinforcement learning. *arXiv preprint arXiv:1910.11956*, 2019.

- 68 [3] A. Zeng, P. Florence, J. Tompson, S. Welker, J. Chien, M. Attarian, T. Armstrong, I. Krasin,  
69 D. Duong, V. Sindhwani, and J. Lee. Transporter networks: Rearranging the visual world for  
70 robotic manipulation. *CoRL*, 2020.
- 71 [4] M. Shridhar, L. Manuelli, and D. Fox. Cliport: What and where pathways for robotic manipula-  
72 tion. In *Conference on Robot Learning*, pages 894–906. PMLR, 2022.
- 73 [5] Calvin. <https://github.com/mees/calvin/>.
- 74 [6] Relay policy learning environments. [https://github.com/google-research/](https://github.com/google-research/relay-policy-learning/)  
75 [relay-policy-learning/](https://github.com/google-research/relay-policy-learning/).
- 76 [7] Cliport. <https://github.com/cliport/cliport>.
- 77 [8] From play to policy: Conditional behavior generation from uncured robot data. [https:](https://github.com/jeffacce/play-to-policy)  
78 [//github.com/jeffacce/play-to-policy](https://github.com/jeffacce/play-to-policy).