DemoFunGrasp: Universal Dexterous Functional Grasping via Demonstration-Editing Reinforcement Learning

Supplementary Material

A. Hardware Setup

We use a Franka 3 arm paired with an Inspire robotic right hand. Objects are randomly placed within a $0.3 \times 0.3\,\mathrm{m}$ workspace region. To obtain visual observations, we deploy two Intel RealSense D435i cameras positioned to provide complementary viewpoints of the workspace. We perform standard hand–eye calibration to accurately estimate each cameras' extrinsic parameters, and these calibrated parameters are imported into the simulation environment to ensure consistency between real and simulated camera poses. The RGB images captured by the cameras are resized to 256×256 and used as inputs to our vision-based policy. This resolution is selected because it offers a favorable trade-off between detail preservation and system efficiency: compared with higher-resolution inputs, 256×256 images significantly reduce data-transfer latency and computational load while retaining the necessary geometric and semantic cues for reliable perception and control.

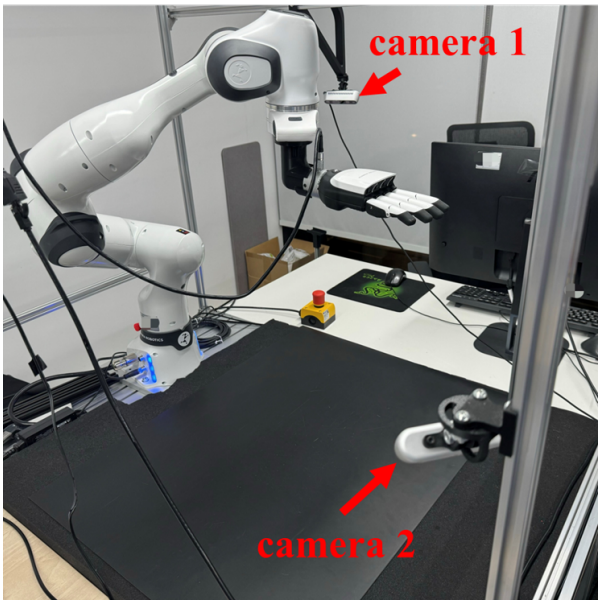

Figure 7. The real-world hardware setup.

B. Objects Used in Experiments

B.1. State-based Training Dataset

In the **state-based configuration**, we utilize three datasets to comprehensively evaluate our method across multiple dimensions:

Figure 8. Objects used to train state-based policy in the simulator.

- **DexGraspNet [31]**: a large-scale dexterous grasping dataset containing over 3,200 diverse objects, serving as the primary benchmark for evaluating general grasp performance.
- **YCB [6]**: a standard benchmark consisting of 75 everyday objects and tools, used to assess the ability of DemoFunGrasp to perform functionally diverse grasps.
- **AffordObj**: a dataset constructed for functional grasping tasks that require attending to specific object regions, derived from the YCB object set.

For our training and evaluation split, we curate a mixed dataset of 175 objects sourced from DexGraspNet and YCB. This combined dataset provides a broader and more varied object distribution than either dataset alone. A small subset of the mixed dataset is shown in Fig. 8.

B.2. Dataset Processing

To enable more accurate sampling of affordance points on objects, we adopt a manual annotation pipeline (Fig. 9). We additionally leverage the AffordPose [16] dataset to sample affordance points for training the state-based model. Our experiments indicate that human-preferred functional regions do not necessarily correspond to regions that are easy for the robot to grasp. Consequently, conditioning solely on human-labeled functional regions does not improve either the training success rate or the affordance accuracy of the state-based model.

Our main framework requires providing the model with high-level semantic information. Experiments show that, after training on the annotated dataset and distilling the model into a vision-based policy *without* explicit affordance conditioning, the student policy can still generalize to seen category objects by grasping their functional regions autonomously—without human or VLM guidance. This demonstrates that our method is capable of acquiring

semantic understanding when scaled to a sufficiently large annotated dataset.

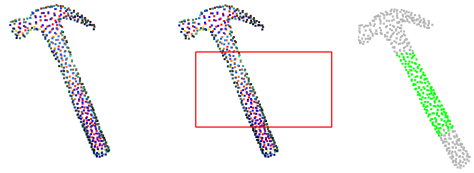

Figure 9. Pipeline for annotating affordance points on point clouds.

B.3. Simulation Object Categorization

In the simulator, after training on a large and diverse set of objects, we evaluate the vision-based policy under human guidance by specifying the desired affordance and grasping style. To systematically assess generalization, we categorize objects into three shape-based classes: *food items*, *kitchen items*, and *tools*. For each category, we select five representative objects for evaluation (Fig. 10).

Figure 10. Objects used to evaluate the vision-based policy in simulation.

B.4. Real-World Object Categorization

Objects used in the real-world experiments are organized into two sets. The primary benchmark contains 9 representative daily objects, shown in Fig. 11. These objects are categorized according to functionality and geometry as follows:

- **Daily Items**: everyday objects with no clearly defined affordance region, such as balls and bananas.
- **Small Tools**: compact tools including spray bottles and small teapots.
- **Large Tools**: larger household tools such as bowls and kettles.

Figure 11. Objects used to evaluate the vision-based policy in real-world experiments.

To further evaluate generalization, we additionally construct an extended real-world test set containing 33 new objects with diverse geometries, including irregular and partially deformable items. These objects are shown in Fig. 12. The expanded set significantly increases appearance and shape diversity compared to the primary benchmark.

Figure 12. Extended real-world object dataset used for generalization evaluation. The set contains 33 additional objects beyond the primary 9-object benchmark (Fig. 11), covering a broader range of shapes, sizes, and functional categories.

C. Cross-Embodiment Evaluation

We evaluate DemoFunGrasp across three robotic hand embodiments. For the Shadow Hand, grasping style priors are derived from the Dexonomy [7] dataset, whereas for the Inspire Hand and Wuji Hand, styles are manually defined via joint tuning. Both sources of style initialization require minimal manual effort while effectively achieving high Grasp Success Rate (GSR) and low Success Affordance Distance (SAD).

Comprehensive results for the state-based evaluation are presented in Table 5. The Inspire Hand achieves the highest success rate and the lowest affordance distance, likely due to its lower degrees of freedom (DoF), which simplify optimization.

Our pipeline can be easily applied to an entirely new type of robotic hand. We first obtain at least 33 grasp styles (based on Grasp Taxonomy) via retargeting. In training period, styles unsuitable for tabletop grasping tasks can be filtered out by evaluating the success rates of different styles.

Table 5. **Cross-Embodiment Evaluation of DemoFunGrasp.**

Hand	DoF	Styles	GSR↑	SAD↓
Inspire Hand	6	4	**87.85**	**2.66**
Shadow Hand	22	9	77.04	3.02
Wuji Hand	20	9	77.09	2.74

Table 6. Sensitivity to demonstration quality. We compare replaying demonstrations with the trained policy. Policy results report success rate (%) / affordance distance (cm).

Demo	small-top	small-side	big-top	big-side
Replay	49.6%	33.4%	5.5%	1.1%
Policy	83.9%/2.7	84.1%/2.8	85.1%/2.6	84.8%/2.7

D. Demonstration Sensitivity to Demonstration Quality

Since the RL policy learns to edit demonstrations, demonstrations that are reasonably close to a feasible grasp are sufficient. To verify this, we collect four different teleoperated demonstrations. We evaluate the success rate (%) and success affordance distance (cm) when replaying the demonstrations and after policy training. As shown in Table 6, demonstration quality has little effect on the final policy performance.

E. The VLM Planner

Prompt for **ChatGPT** and **Gemini 2.5 pro**:

”template”: (”Please provide the 2D point coordinate of the region this sentence describes: {instruction}.” ”The input image size is 256×256 pixels.” ”Generate 4 candidate points and select the best one for grasp affordance.” ”The results are presented in a format⟨point⟩[x,y]⟨/point⟩.” ”You FIRST think about the reasoning process as an internal monologue and then provide the final answer.” ”The reasoning process and answer are enclosed within ⟨think⟩⟨/think⟩ and ⟨answer⟩⟨/answer⟩ tags.” ”The answer consists of only one coordinate point, with the overall format being:⟨think⟩reasoning process here⟨/think⟩⟨answer⟩⟨point⟩[x,y]⟨/point⟩⟨/answer⟩.” ”Important: the point must lie on the object, not on the background or table surface.”),
”description”: ”Object Affordance Grounding - Locating the 2D coordinates of specified object regions based on descriptions.”

Prompt for **Embodied-R1**:

”template”: (”Please provide the 2D points coordinate of the region this sentence describes: {instruction}.” ”The results are presented in a format ⟨point⟩[[x1,y1], [x2,y2], ...]⟨/point⟩.” ”You FIRST think about the reasoning process as an internal monologue and then provide the final answer.” ”The reasoning process and answer are enclosed within⟨think⟩⟨/think⟩and⟨answer⟩⟨/answer⟩ tags.” ”The answer consists only of several coordinate points, with the overall format being:⟨think⟩ reasoning process here ⟨/think⟩⟨answer⟩⟨point⟩[[x1,y1], [x2,y2],...]⟨/point⟩⟨/answer⟩”),
”description”: ”Object Affordance Grounding - Locating the 2D coordinates of specified object regions based on descriptions.”

Instruction:

Grasp the object on the table by identifying the optimal affordance region, and return the coordinates of the reasoning points.

The VLM-generated outputs are shown in Fig. 13. While Gemini 2.5 Pro and GPT-5 demonstrate strong reasoning capabilities and can produce logically coherent interpretations of tabletop scenes, they consistently fail to generate precise point coordinates. We hypothesize that this limitation stems from their insufficient modeling of pixel-level spatial information.

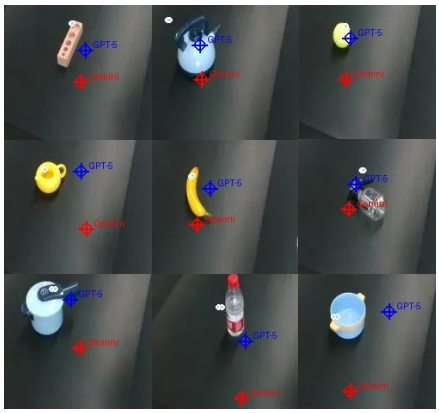

Figure 13. Comparative evaluation of Embodied-R1 (white points), Gemini 2.5 Pro (red points), and GPT-5 (blue points).

Our insight is that, to achieve a universal robotic manipulation policy, it is necessary to train a high-level “cognitive” model capable of long-horizon reasoning and task planning, while the low-level policy focuses primarily on ensuring execution robustness and stability.

F. Additional Qualitative Results

F.1. Results in Simulation

Fig. 14 presents recordings of the vision-based policy in simulation, demonstrating its versatility across a range of object types and grasping scenarios. In addition to executing grasps with a specified hand style, our method effectively handles small, thin, or fragile objects, as well as objects prone to rolling or instability. These demonstrations highlight the policy's ability to adapt to challenging object geometries and physical dynamics, showcasing its generalization capability within the simulator.

F.2. Real-World Results

Fig. 15 presents real-world demonstrations, illustrating the effectiveness of sim-to-real transfer and the robustness of our approach. The policy successfully grasps a wide range of challenging objects, including extremely large objects (e.g., a watering can), delicate items (e.g., a bunch of flowers), and heavy tools (e.g., a long metal instrument). These results emphasize the capability of the vision-based policy to generalize from simulation to real-world tasks while maintaining both precision and functional awareness.

Beyond stable grasping, our method can be extended to enable functional manipulation tasks. For example, it can pour water using a teapot or water plants with a spray bottle, demonstrating that the policy not only handles grasping challenges but also executes downstream functional behaviors.

G. Additional Ablation Studies

In addition to the experiments presented above, we also explore several alternative approaches. Although these methods are ultimately suboptimal, they provide useful insights into the challenges of functional grasping.

Sampling-based method. We first attempt to collect data using a sampling-based strategy. However, its data efficiency is extremely low (around 5%), requiring substantial computation to gather a sufficiently large dataset. Even after more than 10k samples, the policy does not learn meaningful affordance cues. Many object segments are *intrinsically difficult* to grasp without refined end-effector rotations and hand joint positions. As a result, the dataset becomes highly imbalanced: trajectories concentrate on only a few graspable segments per object, and a large portion of samples come from objects that are naturally easier to grasp.

Planning-based method. We also train a policy to predict a pre-grasp rotation and translation, and then plan toward the target affordance via linear interpolation. However, this interpolation strategy severely restricts the feasible action space and makes it difficult to reach certain geometric affordance regions. Experiments on the YCB dataset show that the initial success affordance distance is 3.8 cm. After optimization with a binary success reward, the success rate increases to over 60%, but the mean success affordance distance increases to over 6 cm. These results indicate that although training improves binary success, linear interpolation fundamentally limits grasp accuracy, especially for objects with diverse geometries.

Sampling styles from successful trajectories. We further analyze successful DemoGrasp trajectories and sample hand-style distributions as priors for style adaptation. However, the resulting hand poses are neither human-like nor stable for policy learning. The "diverse" grasp styles lack correlation with object geometry and often lead to loose or suboptimal grasps. This suggests that human-like grasping styles are inherently object-dependent and that geometry-aware style generation is essential for achieving tight, functional grasps.

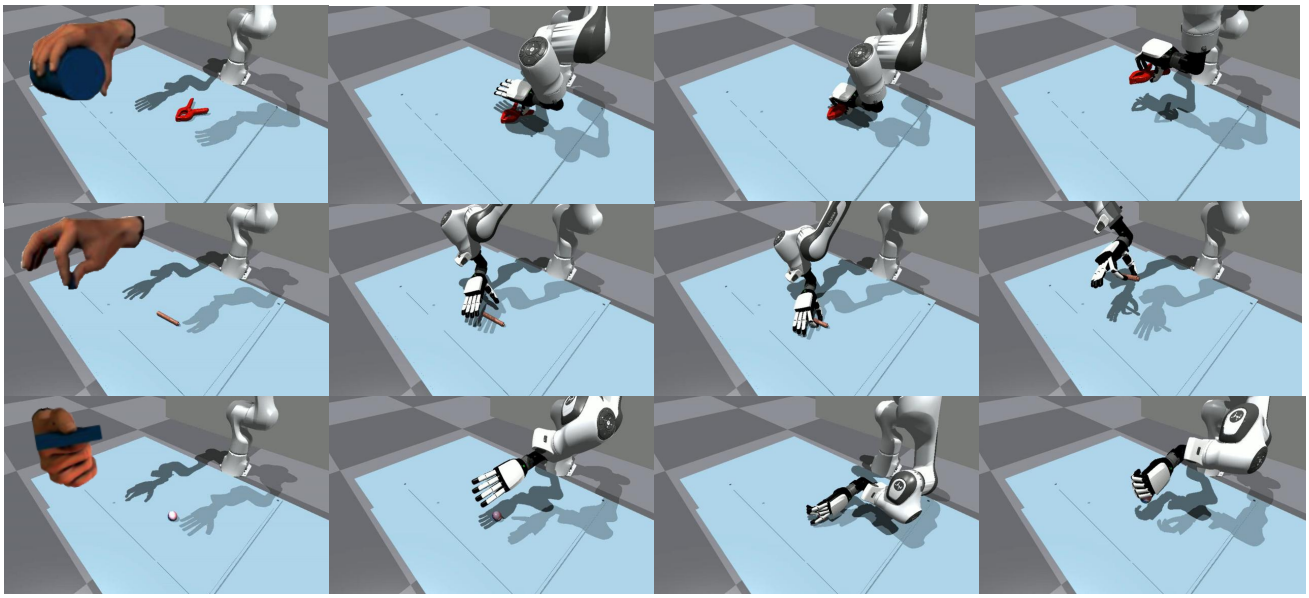

Figure 14. Simulator video recordings of the vision-based policy across diverse objects and grasping styles.

Figure 15. Real-world video recordings of the vision-based policy performing functional grasps on a variety of objects.