

GraspLDP: Towards Generalizable Grasping Policy via Latent Diffusion

Supplementary Material

A1. Data Collection for Train and Evaluation

Data efficiency. For simulation experiments, we utilized 12k demonstrations across 20 objects (about 840k frames), while for real-world experiments, we used 500 demonstrations across 10 objects (about 32k frames). Notably, this is substantially less data compared to GraspVLA, which trains on billion-level frames, and is also far smaller than typical VLA models such as π_0 . We use 12K simulated demonstrations for 20 objects and 500 real-world demonstrations over 10 objects, which is substantially less data than GraspVLA (billion-level frames) and typical VLA models such as π_0 . Besides, we also evaluated GraspLDP with 1.2K and 120 demonstrations (6 per object), shown in the table below. As training data decreases, our method exhibits significantly better few-shot capabilities than original DP.

Table 1. Results of In-domain evaluation with few demonstrations.

Method	12K	1.2K	120	Avg
Diffusion Policy	62.8	41.5	13.8	39.4
GraspLDP	80.3 (+17.5)	64.6 (+23.1)	43.1 (+29.3)	62.7 (+23.3)

Using different grasp detectors. We evaluated several grasp detectors with varying performance levels as priors, as shown in the table below. Our method consistently outperforms the grasp detection baseline with traditional motion planning. Notably, the performance gain is even more significant when the initial grasp poses are **suboptimal**, demonstrating the robustness and corrective capability of our proposed closed-loop diffusion-based policy.

Table 2. Results of In-domain setting with different grasp detector.

Method	GSNet	SBG	GraspNet
Grasp Detection	78.5	73.8	70.8
GraspLDP	80.3 (+1.8)	76.9 (+3.1)	75.4 (+4.6)

A2. Data Collection for Train and Evaluation

There exist many open-source datasets of expert demonstrations, collected on real robots or in simulator, that are widely used to train imitation learning policies. However, these datasets are often heterogeneous and contain relatively few demonstrations specifically targeted at grasping tasks; moreover, interaction patterns with objects tend to be narrow (for instance, grasp poses are commonly concentrated around simple top-down planar grasps). Such distri-

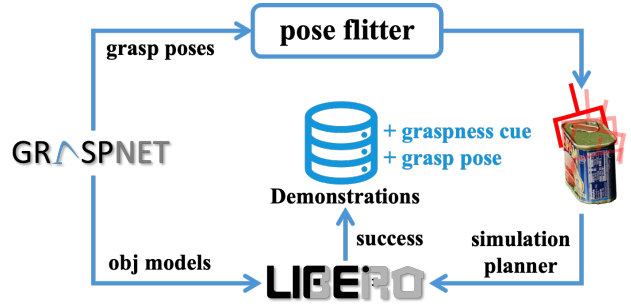


Figure 1. Illustration of the data collection pipeline in simulator.

butional biases strongly limit policy performance on diverse or precise grasping tasks.

To address these limitation, we redesigned our demonstration collection pipeline starting from a grasp detection dataset. GraspNet-1Billion [3] provides dense grasp annotations for 78 objects (each object appearing in multiple poses and scenes), which nicely matches our needs. As illustrated in Figure 1, we import GraspNet-1Billion’s 3D object assets into the LIBERO [6] benchmark (built on the robosuite [13] simulator, which in turn uses MuJoCo [10] as physics engine). After correct transformation, we can place any object in any training scene in LIBERO and directly sample candidate grasp poses from the dataset labels. For each object we sample a large set of high-quality grasp poses and apply NMS to avoid clustering poses in a small region and to ensure diversity. We generate end-effector motion trajectories by Spherical Linear Interpolation (SLERP) for rotations and Linear Interpolation for translations. We then simulate these trajectories in LIBERO and, at each timestep, record the wrist-view RGB images augmented with the graspness visual cue. We log the target grasp pose per episode and include only demonstrations that succeed in simulation into our final dataset. We employ a heuristic speed de-biasing procedure during trajectory synthesis so that the resulting trajectories share as uniform velocities as possible cause prior study [8] have shown that distributional shifts in trajectory speeds across datasets increase the difficulty of policy learning.

A3. Details of Real World Evaluation

A3.1. Language Label for Objects

Figure 2 shows the language labels of objects used in our real world evaluation. These labels are primarily used for evaluating the GraspVLA[1] and are also employed within the cluttered scenarios evaluation.



Figure 2. Language label for each object.

A3.2. Cluttered Scenarios Evaluation

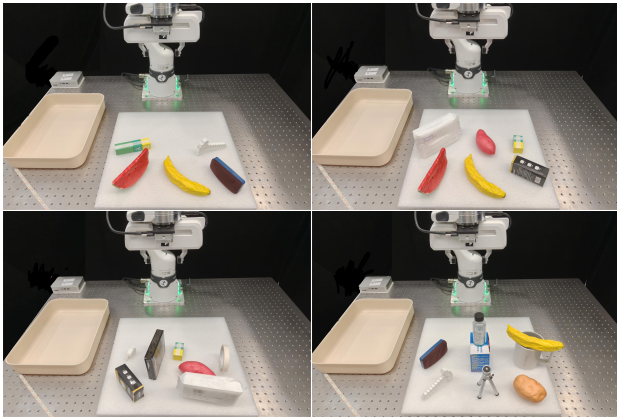


Figure 3. Initial object placements in cluttered scenerios evaluation from scene 1 to scene 4.

As shown in Figure 3, we set up four cluttered scenes, each containing both seen and unseen objects. The number of objects increases from 5 to 8 across scenes 1–4, and scene 4 includes a particularly challenging stacked-object configuration. The goal of task is to grasp and clear all objects from the tabletop; each scene allows up to 10 grasp attempts. For GraspVLA we form language instructions using each object’s label (e.g. “pick up the banana”). For Anygrasp[2] and our method, we follow an approach that uses Grounding DINO[7], a robust open-vocabulary object detector, to filter candidate grasp poses for the target object. During trials, when the timestep reaches a predefined threshold we invoke motion planning to move the gripper to a fixed waypoint and open it to complete the placement step, thereby ensuring continuity of the table-clearing process.

Table 3. Training and inference hyperparameters of Action Latent Learning.

Parameter	Value
horizon	16
n_obs_steps	2
n_latent_dims	16
use_conv_encoder	True
conv_latent_dims	64
conv_layer_num	1
use_rnn_decoder	True
rnn_latent_dims	64
rnn_layer_num	1
kl_multiplier	1e-6
n_embed	16
use_vq	False
dataloader.batch_size	128
dataloader.num_workers	4
optimizer.target_	<code>torch.optim.AdamW</code>
optimizer.lr	1.0×10^{-3}
optimizer.weight_decay	1.0×10^{-4}
training.lr_scheduler	cosine
training.lr_warmup_steps	100
training.num_epochs	1000

A3.3. Description of Dynamic Grasp Tasks

For moving target objects, the policy must continuously update its visual observations and the guidance of grasp pose to realize closed-loop dynamic grasps. In the *banana moving* and *watermelon moving* tasks, the object starts from a corner of the workspace and traverses the entire area; in the *mug handover* task, a human holds the mug by the handle and moves it horizontally and vertically with the goal of having the gripper catch the wall of mug and complete the handover. Because we moved the objects manually, it is difficult to guarantee that the movements are exactly the same across different methods. Therefore, to demonstrate advantage of GraspLDP, we always ensured that the object movement distance was the longest in the evaluation of our method.

A4. Model Implementation

In Action Latent Learning, we use a 1D-CNN encoder \mathcal{E} as action chunk encoder, and adopt a GRU as action chunk decoder \mathcal{D} . During training of the VAE, we set $\lambda = 1 \times 10^{-6}$. Keeping it as such a small value reduces the strength of KL regularization and improves reconstruction quality, since we do not need to sample directly from this latent space for generation. More details of the VAE model and training settings are show in Table 3.

In the Diffusion on Latent Action Space, we freeze the

Table 4. Training and inference hyperparameters of Diffusion on Latent Action Space.

Parameter	Value
observation_horizon	2
prediction_horizon	16
action_horizon	8
unet.diffusion_step_embed_dim	256
unet.down_dims	[256,512,1024]
unet.kernel_size	5
unet.n_groups	8
enable_ddim	True
num_training_timesteps	100
num_inference_timesteps	10
prediction_type	epsilon
use_recon	True
recon_loss_weight	0.2
dataloader.batch_size	64
dataloader.num_workers	8
optimizer.target_	<code>torch.optim.AdamW</code>
optimizer.lr	1.0×10^{-4}
optimizer.weight_decay	1.0×10^{-6}
training.lr_scheduler	cosine
training.epoch_every_n_steps	100
training.num_epochs	1500
training.use_ema	True
training.ema_power	0.75
training.ema_power	0.75

parameters of VAE and use the action latent produced by the action chunk encoder as the supervision of denoising target. We use ResNet18 as our obs encoder to process agent-view and wrist-view image with size of $H = W = 256$. We use DDIM [9] as the noise scheduler, and the number of denoising steps is 100 during training and 10 at inference. For the graspness cue, we set $\tau = 0.2$ to strike a balance between sufficiently covering graspable regions and introducing less noise. During auxiliary reconstruction of the graspness cue, we follow prior work [5] that introduces self-supervised reconstruction into diffusion policy and set the reconstruction weight $\lambda_{\text{Recon.}} = 0.2$ after several experiments. All rotations are represented with the 6D rotation representation [12], which provides better continuity in the numerical and 3D rotation spaces for neural network learning. More details of the latent diffusion model and training settings are show in Table 4.

In Heuristic Pose Selector (HPS) of inference pipeline, a pre-trained GSNet [11], the core grasp detection module of AnyGrasp [2], are used as grasp detector. We set $k = 30$ to filter low scored grasp candidates. And for matrix W , we set $w_t = 100$ and $w_r = 20$ to achieve a trade-off between measurement of translation and rotation following [4].

In our real world experiments with both Diffusion Policy and GraspLDP, inference latency commonly induces jitter when switching action chunks, which in turn perturbs grasp trajectories. To mitigate this effect, we simply discard the first three actions predicted by the policy at each inference process. This pragmatic process substantially reduces transition-induced oscillations and yields noticeably smoother and more reliable grasp executions.

References

- [1] Shengliang Deng, Mi Yan, Songlin Wei, Haixin Ma, Yuxin Yang, Jiayi Chen, Zhiqi Zhang, Taoyu Yang, Xuheng Zhang, Heming Cui, Zhizheng Zhang, and He Wang. Graspvla: a grasping foundation model pre-trained on billion-scale synthetic action data. In *Conference on Robot Learning*, 2025. 1
- [2] Haoshu Fang, Chenxi Wang, Hongjie Fang, Minghao Gou, Jirong Liu, Hengxu Yan, Wenhai Liu, Yichen Xie, and Cewu Lu. Anygrasp: Robust and efficient grasp perception in spatial and temporal domains. *IEEE Transactions on Robotics*, 39(5):3929–3945, 2023. 2, 3
- [3] Hao-Shu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. Graspnet-1billion: A large-scale benchmark for general object grasping. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 1
- [4] Ankit Goyal, Arsalan Mousavian, Chris Paxton, Yu-Wei Chao, Brian Okorn, Jia Deng, and Dieter Fox. Ifor: Iterative flow minimization for robotic object rearrangement. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 3
- [5] Xiang Li, Varun Belagali, Jinghuan Shang, and Michael S Ryoo. Crossway diffusion: Improving diffusion-based visuomotor policy via self-supervised learning. In *IEEE International Conference on Robotics and Automation*, 2024. 3
- [6] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems*, 2023. 1
- [7] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, 2024. 2
- [8] Modi Shi, Li Chen, Jin Chen, Yuxiang Lu, Chiming Liu, Guanghui Ren, Ping Luo, Di Huang, Maoqing Yao, and Hongyang Li. Is diversity all you need for scalable robotic manipulation? *arXiv preprint arXiv:2507.06219*, 2025. 1
- [9] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 3
- [10] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012. 1

- [11] Chenxi Wang, Haoshu Fang, Minghao Gou, Hongjie Fang, Jin Gao, and Cewu Lu. Graspness discovery in clutters for fast and accurate grasp detection. In *IEEE/CVF International Conference on Computer Vision*, 2021. [3](#)
- [12] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. [3](#)
- [13] Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Soroush Nasiriany, and Yifeng Zhu. robosuite: A modular simulation framework and benchmark for robot learning. *arXiv preprint arXiv:2009.12293*, 2020.