Modality Dropout in Privileged-Pose Robotic Manipulation Policies: Investigating Robustness to Pose Corruption

Tobias Yin-Ching Chan
The Chinese University of Hong Kong
1155232245@link.cuhk.edu.hk

Abstract

Policies for robotic manipulation that are trained with perfect object pose information tend to depend too heavily on it, which makes them fragile in real deployment when pose estimates are unreliable or missing. We investigate this issue in a simulated pick-and-place task using NVIDIA Isaac Lab, where we intentionally corrupt the privileged pose signal through modality dropout.

We compare four PPO policies that use identical task setup and rewards but differ in their access to proprioceptive force sensing and whether they were trained with pose dropout: a baseline (M1), one augmented with force feedback (M2), one trained with pose dropout (M3), and one combining both force and dropout training (M4).

Corruptions are injected at various stages of the task and for varying lengths of time, using four realistic failure modes—hard missing estimates, frozen (stale) poses, noise plus gradual drift, and delayed poses—to simulate actual perception outages.

The results reveal a strong over-reliance on privileged pose: success rates drop close to zero when the pose signal disappears early in the episode, forcing the policy to fall back on other cues. Robustness depends heavily on the task phase—training with mixed pose dropout (M3) consistently gives the best recovery once the robot makes contact and the highest overall AUC across corruption types. Force feedback provides some benefits, but mostly under stale pose conditions.

Interestingly, adding more modalities isn't always better; combining force sensing with dropout training can sometimes perform worse than dropout alone during severe outages. Overall, these findings indicate that deliberately training under pose outages is a more effective way to build robustness than simply adding extra sensors.

Code and supplementary materials are available at `https://github.com/Toby0614/IsaacLab_manipulation`.

1 Introduction

Reinforcement learning (RL) has enabled robotic manipulation policies to tackle diverse contact-rich tasks, from surgical assistance to industrial assembly [9, 10]. In simulation-based training, a widespread practice is to provide policies with privileged information—such as ground-truth object poses—to improve sample efficiency and convergence [6, 7]. However, this often leads to over-reliance on these unrealistic signals, causing policies to underutilize other modalities (e.g., raw vision or proprioception) and exhibit brittleness when privileged inputs are degraded or unavailable in deployment [13].

Real-world perception pipelines frequently suffer from pose estimation failures due to occlusions, tracker loss, or environmental variations—issues that are generally under-studied in manipulation RL [16, 15, 5]. In this work, we deliberately incorporate privileged object pose into the observation space of policies trained on a simulated pick-and-place task. We then apply systematic modality dropout to this pose signal—simulating realistic outages—to diagnose over-reliance on privileged information. Furthermore, we analyze the impact of corruption timing and duration across policies. To explore multimodal redundancy, we added gripper force sensing as part of the input data and evaluate its ability to compensate for pose degradation, particularly in post-contact phases [2, 8].

2 Related Work

Several studies leverage privileged information during training to improve policies in partially observable or sensor-limited settings. Privileged Information Dropout [7] uses privileged state to guide multiplicative noise on standard observations, improving robustness in partially observable navigation. Scaffolder [6] similarly employs privileged sensing to scaffold world modeling, yielding strong performance in robotic manipulation with limited deployable sensors.

Force and tactile feedback are well-studied in multimodal manipulation: they often improve grasping and contact-rich control when vision is noisy or incomplete [3, 4], but can be redundant or even degrade performance depending on task and integration [12, 1]. Most prior work, however, focuses on vision–tactile complementarity under image corruption or occlusion. Fewer studies examine robustness when privileged state signals (e.g., object pose) are corrupted or missing, which is the realistic failure mode when such pose estimates are produced by perception pipelines. Our work targets this gap by evaluating robustness to pose-estimate outages and quantifying how force sensing and dropout-trained policies compensate across task phases.

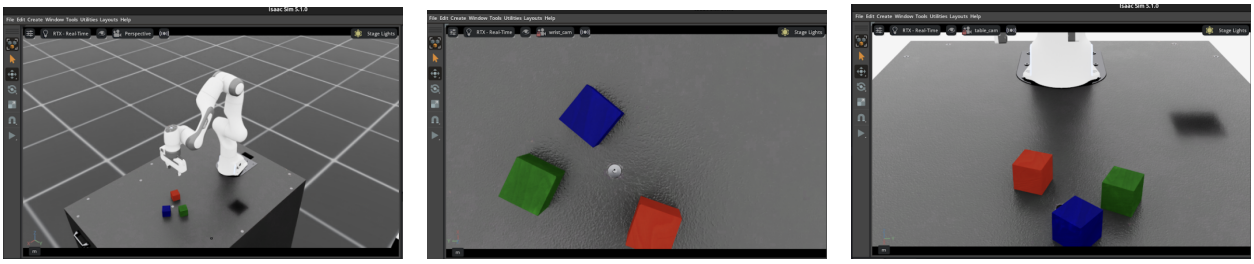
3 Methods

We conduct experiments in the NVIDIA Isaac Lab simulation environment, using the official stack environment set up. We trained 4 policies with the same task: the manipulator must pick up the correct cube and then lift it to the goal area. The policies are trained using Proximal Policy Optimization (PPO) implemented by the built-in RL library in Isaac Lab, rsl_rl [14].

All policies share the same dense reward/penalty function that encourages individual progress in the task; for instance, rewards for reaching the object and successful grasping, and penalty for dropping the objects and penalties to promote smooth movement for the manipulator such as jerky or oscillatory actions by implementing a second-order temporal consistency term, which explicitly suppresses rapid oscillations for even more stable trajectories.

3.1 Observation Space

All policies receive multi-modal observations combining visual, proprioceptive, and privileged inputs. Visual data consists of RGB-D images from a wrist-mounted camera and RGB image from a fixed overhead table camera. Each camera stream is processed independently through the convolutional neural network (CNN) encoder provided by the rsl_rl library, producing feature vectors; these features are then concatenated with a low-dimensional proprioceptive vector, which includes the privileged input: object pose.



(a) Single environment configuration (b) View from the mounted camera (c) View from a fixed table camera

Figure 1: (a) single environment configuration of the task. (b) View from the mounted camera. (c) View from a fixed table camera.

Randomization includes:

- Robot joint state noise at reset
- Cube positions and yaw randomized at reset (x/y ranges, yaw range), for cube_1,2,3 with a minimum separation.

3.2 Data Stream

The policy receives two observation streams: a multicamera image tensor and a proprioceptive vector. The vision stream is 7-channel CHW: wrist RGBD (4 channels) concatenated with table RGB (3 channels), which is encoded by the CNN. The proprio vector (joint states, EE pose, gripper state, goal position, and oracle cube position) is encoded by an MLP. The CNN features are then concatenated with the proprio MLP features and fed to the final policy and value heads.

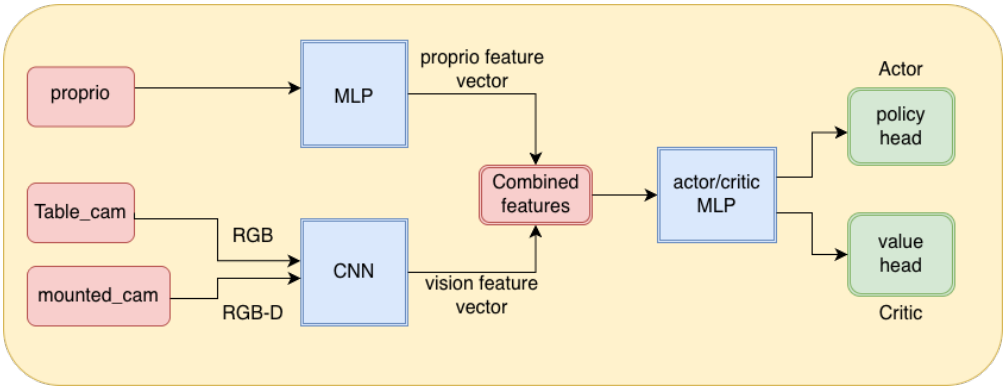


Figure 2: Policy's data stream.

For the policies that enable the force feedback, the force data will be appended to the proprio data stream.

3.3 Modality Dropout

To systematically evaluate robustness to real-world perception failures, we define modality dropout as controlled corruption applied to the privileged object pose estimate.

We implement four realistic corruption modes:

- Hard (missing estimates): set object position to [0, 0, 0]
- Freeze (stale estimate): returns the last observed pose; stale state
- Noise + drift (jitter + drift): adds zero-mean Gaussian noise and a slowly varying drift bias
- Delay: uses a ring buffer of past cube positions and, while active, replaces the current pose with the value from delay_steps earlier (i.e., pose(t) $\leftarrow$ pose($t - k$)), pulled from the buffer.

Those modes represent real perception problems and are well documented in recent studies [17].

3.4 Modality Dropout in Training

To provide a robust comparison, we also implement the modality dropout on the privileged object_pose during training for two of the policies. We implement pose-dropout as event-based, temporally correlated corruption applied independently in each parallel environment. When no outage is active, a new corruption event starts with probability per timestep; once triggered, it persists for a randomly sampled duration steps before returning to normal. This models realistic bursty perception failures (e.g., occlusions/tracker loss/latency episodes) and avoids synchronized dropouts across all environments during training; the randomization is motivated by domain randomization in visual sim-to-real gaps in manipulation tasks [11]. We only choose to use hard, freeze, and noise as the training dropout modes, and preserve the delay for evaluation to measure the policy's general response against new types of dropouts.

3.5 Policies

The four policies trained use identical PPO hyperparameters, reward function, task setup, and base observation structure.

Table 1: Comparison of the four policies trained

Policy	Force Feedback	Pose Dropout Training	Description
M1	No	No	Baseline
M2	Yes	No	Force-augmented
M3	No	Yes	Pose-dropout trained
M4	Yes	Yes	Force + pose-dropout

4 Results and Discussion

Each policy is trained until reaching sufficient (¿85%) defined by the full completion of the pick-and-place task; specifically, all policies achieve high clean-task success rates of: M1 = 0.8717, M2 = 0.8933, M3 = 0.9446, and M4 = 0.961. These baseline rates confirm that all policies solve the task under their respective inputs and conditions; the robustness comparisons below therefore reflect sensitivity to pose outages rather than failure to learn the base task.

We evaluated all four trained policies under systematic oracle pose corruption by sweeping corruption mode the definition of these phrases is as follow:

reach: default phase when none of the later conditions are met (not near object with closed gripper, not lifted, not at goal).

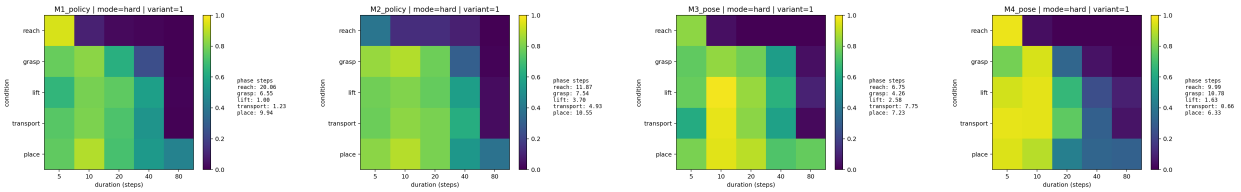
grasp: end-effector is near the object (distance ¡grasp_dist_threshold) and gripper is closed, but object is not lifted.

lift: object height above lift_threshold, but not high enough for transport (transport_height_min).

transport: object is lifted and height above transport_height_min.

place: object XY is within goal radius (goal_xy_radius) regardless of lift state.

Each sweep lasts for duration for both (i) phase-triggered onset across {reach, grasp, lift, transport, place} and (ii) time-triggered onset across specified step indices. (ii) does not include the specific data we need for investigating the timing and duration, but it's included for a complete evaluation and can be found in appendix. For each condition we ran the evaluation until 500 episodes are recorded and within those 500 episodes, we excluded the episodes where the policy was terminated before a corruption is triggered and recorded task success rate and corruption exposure statistics.



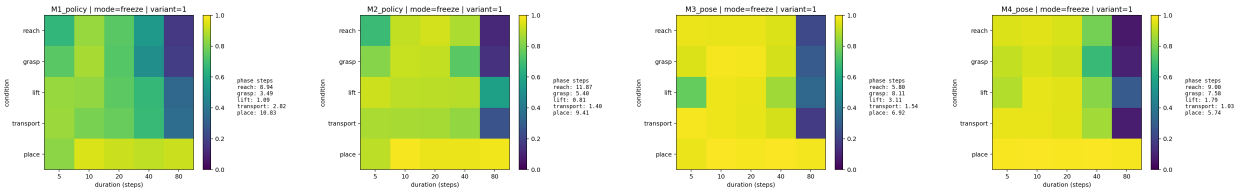
(a) M1 (b) M2 (c) M3 (d) M4

Figure 3: Hard-mode heatmaps (phase × duration) for M1–M4. Each panel shows success rate when corruption is applied to privileged cube position at different phases and durations along with the phases' time steps for different policies.

Under hard pose dropout in variant-1, it is already clear that the policies strongly, or even solely rely on the cube's position to perform the task, reaching 0% success rate if the dropout begins at reach and lasts the entire episode; with the data of 500 episodes, this proves very well that all policies practically cannot finish

the task without the privileged data that they were trained on, even for M3/M4 policies, where a mixed of dropout modes were applied randomly during training.

For the investigation of the redundancy of the force sensor, we must look at the data after contact: while all models show severe reach-phase sensitivity, the post-grasp phases differentiate robustness. M3 exhibits the strongest sustained performance, particularly in place (0.828 average success across durations), suggesting that mixed-dropout training recover the best when the privileged pose signal is lost. M2 improves over M1 in transport and place, indicating that force sensing provides some redundancy after contact, but it still does not match M3's placement robustness. In contrast, M4 is the weakest overall after grasp despite combining force with mixed-dropout; this implies that the added modality does not generalize to the hardest outage regime and may even exacerbate early-phase failures that depress overall performance. These hard-mode results thus highlight that robustness is not monotonic with added modalities: dropout-aware training (M3) is the most effective in late-phase recovery, while force sensing alone is insufficient under fully missing pose.

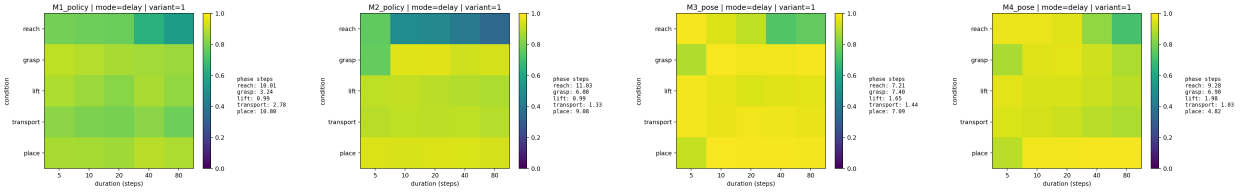


(a) M1 (b) M2 (c) M3 (d) M4

Figure 4: Freeze-mode heatmaps (phase × duration) for M1–M4.

Under freeze pose corruption, there is a stronger contrast between lower duration (5, 10 steps) between M1/M2 and M3/4, hinting that mixed-dropout training persist much better with stale data rather than empty data (hard dropout).

Across all policies, place remains high ($\geq$0.903 average), with M4 and M3 nearly perfect (0.992 and 0.986), implying that once the object is near the goal, stale pose is less damaging. The dominant failure regime is still early in the episode at long freeze durations (80 steps), but M3/M4 maintain higher performance than M1/M2 in most phases, demonstrating that freeze is a dropout regime where robustness training and force sensing do transfer effectively.



(a) M1 (b) M2 (c) M3 (d) M4

Figure 5: Delay-mode heatmaps (phase × duration) for M1–M4.

Under delay corruption—which was not included in mixed-dropout training—M3 and M4 still achieve the strongest performance across all phases, indicating that robustness learned from freeze/noise/hard transfers to delayed pose estimates, and potentially other dropout modes. M3 is especially strong in grasp, lift, transport, and place ($\geq$0.960 average across durations), while M4 leads in reach and remains high across later phases. In contrast, M2 shows a clear weakness in reach (0.492), suggesting force feedback alone does not help when the pose estimate is merely lagged rather than missing. The fact that M3/M4 generalize best to this unseen corruption mode strengthens the claim that mixed-dropout training induces broader robustness, not just mode-specific resilience.

The grouped AUC plot compares robustness across all corruption modes. M3 achieves the highest AUC in every mode, indicating that mixed-dropout training yields the most consistent and broad robustness. M2 generally outperforms M1, showing that force feedback provides useful redundancy, but those gains remain smaller than those from mixed-dropout training. M4 performs well under delay and noise but degrades

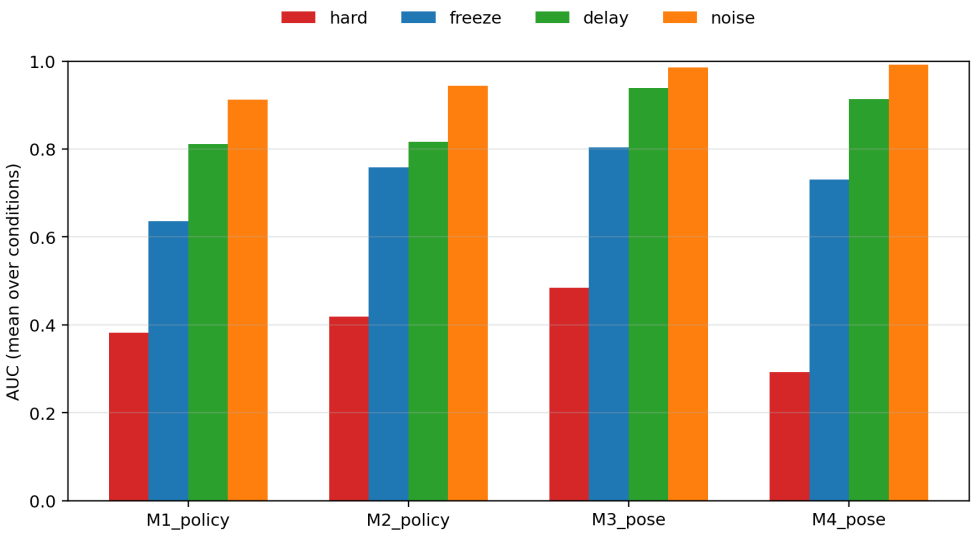


Figure 6: AUC comparison for M1–M4 under pose corruption for different dropout modes.

under hard dropout to the extent it overshadows its mixed-dropout training and backfired—leading to a worse result than M1/2. These results emphasize that robustness is mode-specific: training with pose corruptions produces the most reliable gains, whereas adding modalities alone does not guarantee consistent improvement.

Time-to-success (mean steps across successful episodes) provides a complementary view of efficiency. Averaged across corruption modes, M3 and M4 achieve success faster (23.63 and 22.44 steps, respectively) than M1 and M2 (29.10 and 29.80 steps), with the largest slowdowns occurring under hard dropouts. The policies also differ in phase-time allocation: averaged across all dropout modes, M1 spends the largest proportion in the place phase (42.6%), while M2 and M4 allocate more time to grasp (19.3% and 29.6%). In contrast, M3 shows a more balanced reach–grasp split (26.3% vs. 25.5%) with shorter place time (32.2%), suggesting steadier progression through post-contact phases. This phase-time profile indicates that robustness-trained policies not only tolerate perception corruptions more effectively but also advance through post-contact phases with greater efficiency.

We note that some absolute performance differences can be partially attributed to clean-task success rates (87-97%), but the relative drops under corruption still isolate robustness.

5 Conclusion

These results show that pose outage robustness is the key failure mode for privileged-pose policies. Robust training with mixed pose corruptions improves resilience across multiple modes, and force sensing can provide useful redundancy in certain phases—most notably under stale (freeze) conditions. However, robustness is not monotonic with additional modalities: adding force does not guarantee better performance and can even degrade results under severe outages. The strongest performance comes from training directly on the pose outage regimes that dominate failure, rather than relying on extra sensors to compensate. This suggests that the effect of force feedback is context-dependent and may interact with dropout training in ways that reduce robustness for certain corruption types.

References

[1] Anonymous. When would vision-proprioception policy fail in robotic manipulation?, 2025. Submitted to NeurIPS 2025 (under review).

[2] Roberto Calandra, Andrew Owens, Dinesh Jayaraman, Justin Lin, Wenzhen Yuan, Jitendra Malik, Edward H. Adelson, and Sergey Levine. More than a feeling: Learning to grasp and regrasp using vision and touch. *IEEE Robotics and Automation Letters*, 3(4):3940–3947, 2018. Published version of arXiv:1805.11085.

[3] Roberto Calandra, Andrew Owens, Dinesh Jayaraman, Justin Lin, Wenzhen Yuan, Jitendra Malik, Edward H. Adelson, and Sergey Levine. More than a feeling: Learning to grasp and regrasp using vision and touch, 2018.

[4] Zihao Ding et al. Adaptive visual-tactile fusion recognition for robotic operation of multi-material system. *Frontiers in Neurorobotics*, 17:1181383, 2023.

[5] Juan Del Aguila Ferrandis, João Moura, and Sethu Vijayakumar. Learning visuotactile estimation and control for non-prehensile manipulation under occlusions, 2024.

[6] Edward S. Hu, James Springer, Oleh Rybkin, and Dinesh Jayaraman. Privileged sensing scaffolds reinforcement learning, 2024.

[7] Pierre-Alexandre Kamienny, Kai Arulkumaran, Feryal Behbahani, Wendelin Boehmer, and Shimon Whiteson. Privileged information dropout in reinforcement learning, 2020.

[8] Michelle A. Lee, Yuke Zhu, Krishnan Srinivasan, Parth Shah, Silvio Savarese, Li Fei-Fei, Animesh Garg, and Jeannette Bohg. Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks, 2019.

[9] Sergey Levine, Peter Pastor, Alex Krizhevsky, and Deirdre Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection, 2016.

[10] Puhao Li, Yingying Wu, Ziheng Xi, Wanlin Li, Yuzhe Huang, Zhiyuan Zhang, Yinghan Chen, Jianan Wang, Song-Chun Zhu, Tengyu Liu, and Siyuan Huang. Controlvla: Few-shot object-centric adaptation for pre-trained vision-language-action models, 2025.

[11] Xuanlin Li, Kyle Hsu, Jiayuan Gu, Karl Pertsch, Oier Mees, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, Sergey Levine, Jiajun Wu, Chelsea Finn, Hao Su, Quan Vuong, and Ted Xiao. Evaluating real-world robot manipulation policies in simulation, 2024.

[12] Quan Khanh Luu, Pokuang Zhou, Zhengtong Xu, Zhiyuan Zhang, Qiang Qiu, and Yu She. Manifeel: Benchmarking and understanding visuotactile manipulation policy learning, 2026.

[13] Sasha Salter, Dushyant Rao, Markus Wulfmeier, Raia Hadsell, and Ingmar Posner. Attention-privileged reinforcement learning. In Jens Kober, Fabio Ramos, and Claire Tomlin, editors, *Proceedings of the 2020 Conference on Robot Learning*, volume 155 of *Proceedings of Machine Learning Research*, pages 394–408. PMLR, 2021.

[14] Clemens Schwarke, Mayank Mittal, Nikita Rudin, David Hoeller, and Marco Hutter. Rsl-rl: A learning library for robotics research, 2025.

[15] Nitesh Subedi, Hsin-Jung Yang, Devesh K. Jha, and Soumik Sarkar. Find the fruit: Zero-shot sim2real rl for occlusion-aware plant manipulation, 2025.

[16] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world, 2017.

[17] Hongyin Zhang, Shuo Zhang, Junxi Jin, Qixin Zeng, Runze Li, and Donglin Wang. Robustvla: Robustness-aware reinforcement post-training for vision-language-action models, 2025.