STEREOPOLICY: IMPROVING ROBOTIC MANIPULATION POLICIES VIA STEREO PERCEPTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent advances in robot imitation learning have produced powerful visuomotor policies that manipulate diverse objects from visual inputs. However, monocular observations lack depth information, which is critical for precise manipulation in cluttered or geometrically complex scenes. Explicit depth maps and point clouds are often noisy and fragile in real-world manipulation. We introduce **StereoPolicy**, a visuomotor policy learning framework that directly leverages synchronized stereo image pairs to improve spatial perception without constructing explicit 3D representations. StereoPolicy processes each image with pretrained 2D vision encoders and fuses left-right features through a cross-attention-based Stereo Transformer, learning task-conditioned binocular representations. The framework integrates with diffusion-based and pretrained vision-language-action (VLA) policies, delivering consistent improvements over RGB, RGB-D, point cloud, and multi-view baselines across three simulation benchmarks and seven real-robot tabletop and bimanual mobile manipulation tasks. Our results support task-trained binocular fusion as an effective way to adapt 2D visual representations for manipulation. For more videos, please check our anonymous project page: https://stereopolicy-paper.github.io.

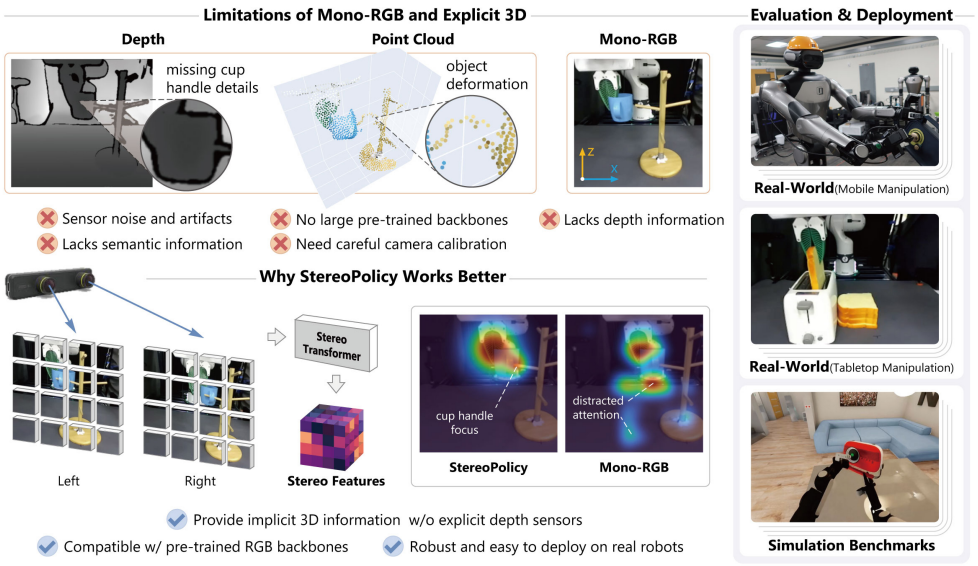


Figure 1: Compared with other visual modalities for robot learning, using stereo image pairs directly as policy input provides implicit 3D information without a depth sensor, works with pretrained 2D encoders, and is easy to deploy on real robots. It avoids the depth ambiguity of monocular RGB, the noise and lost detail of depth maps, and the distortion and calibration demands of point clouds. We introduce StereoPolicy, which offers broader robot-manipulation validation in simulation and real.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Large-scale robot learning has recently produced impressive visuomotor policies and vision-language-action (VLA) models that perform diverse manipulation tasks from visual inputs alone (Levine et al., 2016; Zhu et al., 2018; Shridhar et al., 2021; Reed et al., 2022; Jiang et al., 2023; Brohan et al., 2023b;a; Chi et al., 2024; Zhao et al., 2023; Kim et al., 2024; Black et al., 2025; NVIDIA et al., 2025). These policies typically operate on monocular images processed by pretrained 2D vision encoders to extract semantic and spatial features for downstream action generation (He et al., 2016; Radford et al., 2021; Dosovitskiy et al., 2021). While monocular inputs can be effective in controlled scenes, recovering accurate depth and geometry from a single view remains challenging. These capabilities are crucial for robots to operate in complex everyday environments (Liu et al., 2024; Uppal et al., 2024; Yang et al., 2024; Jiang et al., 2025; Sundaresan et al., 2026) and perform dexterous, fine-grained manipulation (Qin et al., 2022; Wang et al., 2024).

Stereo vision instead naturally encodes 3D spatial structure through synchronized image pairs as shown in Figure 1. Extensive research in computer vision (Marr & Poggio, 1976; Zhang et al., 2020; Poggi et al., 2020; Xu & Zhang, 2020; Lipson et al., 2021; Li et al., 2021; Shen et al., 2021; Li et al., 2022; Weinzaepfel et al., 2023; Xu et al., 2023; Wen et al., 2025) has shown that stereo pairs significantly improve depth estimation and geometric understanding over monocular images, suggesting strong potential for robotic applications where 3D information is necessary.

However, prior work in robotics typically employs stereo images indirectly, by first constructing explicit 3D representations such as depth maps (Shankar et al., 2022; Yang et al., 2023; Uppal et al., 2024; Liu et al., 2024) and point clouds (Qin et al., 2022; Goyal et al., 2023; Ze et al., 2024; Jiang et al., 2024; Yang et al., 2024; Jiang et al., 2025; Sundaresan et al., 2026), and then feeding these into control or planning modules. While effective in controlled settings, such approaches face practical challenges. First, explicit 3D representations are sensitive to sensor noise, calibration errors, and object properties, making high-quality reconstruction difficult to achieve reliably. Even with careful calibration, depth maps often miss fine semantic details, and point clouds suffer from object deformation, as visualized in Figure 1. Second, pretrained 3D vision encoders remain less mature than their 2D counterparts, largely due to the scarcity of large-scale 3D datasets (Geiger et al., 2012; Menze & Geiger, 2015; Scharstein et al., 2014) and scalable 3D backbone architectures (Qi et al., 2017a;b; Thomas et al., 2019; Zhao et al., 2021; Jaegle et al., 2021; Wu et al., 2022; 2024). Third, robotic manipulation requires both high precision and low latency, whereas stereo pipelines that produce explicit disparity maps (Wen et al., 2025) incur substantial inference overhead from cost-volume construction and iterative refinement, posing challenges for real-time manipulation.

In this work, we take a different approach. Instead of relying on explicit 3D reconstruction, we design StereoPolicy, a framework that directly consumes synchronized stereo image pairs for action generation. Our method leverages 2D vision encoders pretrained on internet-scale data to extract features from each image independently, then fuses them into stereo-aware tokens through cross attention in a Stereo Transformer. The fusion module is trained through the action objective, producing task-conditioned binocular features without explicit depth supervision or reconstruction.

Our main results can be summarized as follows:

Performance. We empirically demonstrate that StereoPolicy outperforms policies based on RGB, RGB-D, point cloud, and multi-view across diverse real and simulated manipulation tasks, spanning tabletop and bimanual mobile settings. It can also be integrated to improve pretrained VLA models.

Hardware Insights. We systematically analyze key hardware factors including camera angles and stereo baseline-to-distance ratio, deriving practical guidelines for configuring stereo setups.

Model Design. We comprehensively study stereo encoder design, comparing different backbone choices and identifying an effective architecture for robust stereo-based manipulation.

2 METHOD

We propose StereoPolicy, a stereo perception module to extract task-conditioned binocular features for robot policies (§2.1). The resulting representations can be seamlessly integrated into both diffusion policies and VLA models without modifying their backbone architectures. Below, we describe two models that form the conditions of our experiments (detailed in §3).

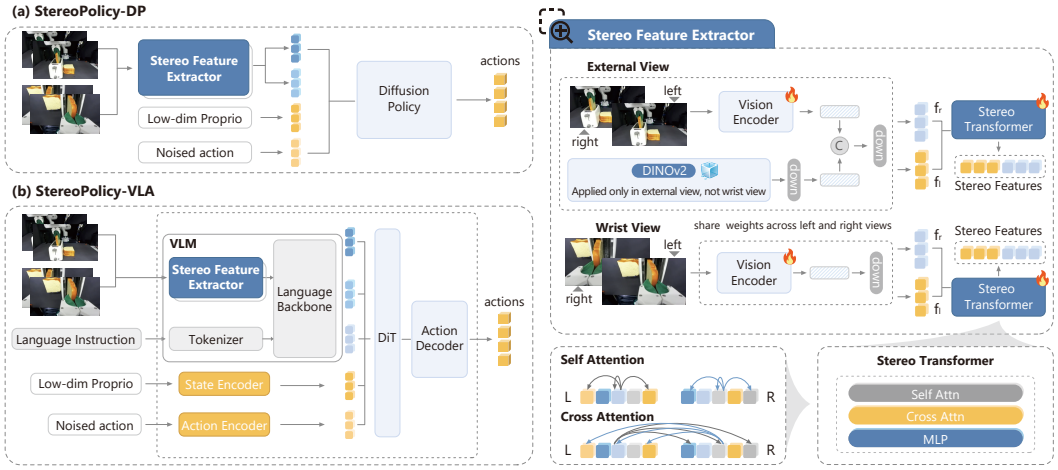


Figure 2: **StereoPolicy Pipeline**. Stereo inputs are encoded by a vision backbone, fused with a Stereo Transformer, and applied to both diffusion policy training and fine-tuning VLA baselines.

- StereoPolicy-DP integrates a stereo encoder into diffusion policy, with the policy backbone trained from scratch on each benchmark task (§2.2).
- StereoPolicy-VLA combines the stereo encoder with a pretrained VLA model and then fine-tunes the whole system end-to-end on each downstream task (§2.3).

2.1 Stereo Feature Extraction

Stereo vision has shown superior performance in 3D understanding, including depth estimation (Wen et al., 2025) and robot grasping (Singh et al., 2026) in simulation environments. Building upon this observation, we design StereoPolicy that leverages stereo image pairs for robot manipulation. Formally, the policy consists of a stereo vision encoder and a policy backbone. The stereo vision encoder takes N pairs of stereo images: $\mathcal{I}_{t,i} = (I^L_{t,i}, I^R_{t,i})$, where $(I^L_{t,i}, I^R_{t,i})$ denote the left and right images captured by a stereo camera setup from the i-th view at timestep t.

Stereo image encoding Each image is processed by a 2D vision encoder $V_i(\cdot)$ to produce single-view feature maps: $f^L_{t,i}, f^R_{t,i}, \quad f^{\cdot} \in \mathbb{R}^{H\times W\times C}$. Within each stereo pair, the left and right encoders share weights to preserve geometric consistency. To enrich visual representations, we augment the image features with DINOv2 (Oquab et al., 2024) embeddings, which provide strong monocular priors to disambiguate regions where epipolar correspondence is unreliable (Wen et al., 2025):

$$\tilde{f}^{\cdot}_{t,i} = \phi\big([f^{\cdot}_{t,i} \| f^{\text{DINO}}_{t,i}]\big), \tag{1}$$

where $\phi(\cdot)$ is a linear projection aligning feature dimensions. Empirically, we observe that DINOv2 improves external views but degrades wrist-view performance, likely due to domain mismatch with egocentric imagery. Therefore, DINOv2 is applied only to external views. We ablate the architecture without DINOv2, and the effect of applying it to wrist views, in §4.2.

Stereo feature fusion To lift 2D image features with implicit depth cues, we adopt an alternating attention mechanism in Stereo Transformer: self-attention computes attention between pixels, while cross attention computes attention between left and right images. We apply 2D RoPE (Heo et al., 2024) to cross-attention queries and keys using each token's image coordinates (x, y) to encode spatial positions (Singh et al., 2026).

$$\forall i \in \{1, \ldots, N\}: \quad z_{t,i} = \text{StereoTrans}_i\big(\tilde{f}^L_{t,i}, \tilde{f}^R_{t,i}\big) \in \mathbb{R}^{d_z}, \tag{2}$$

Multi-view aggregation Stereo fusion is performed independently per camera view. The resulting embeddings are then concatenated (denoted as $\|$) with low-dimensional proprioceptive features s_t as

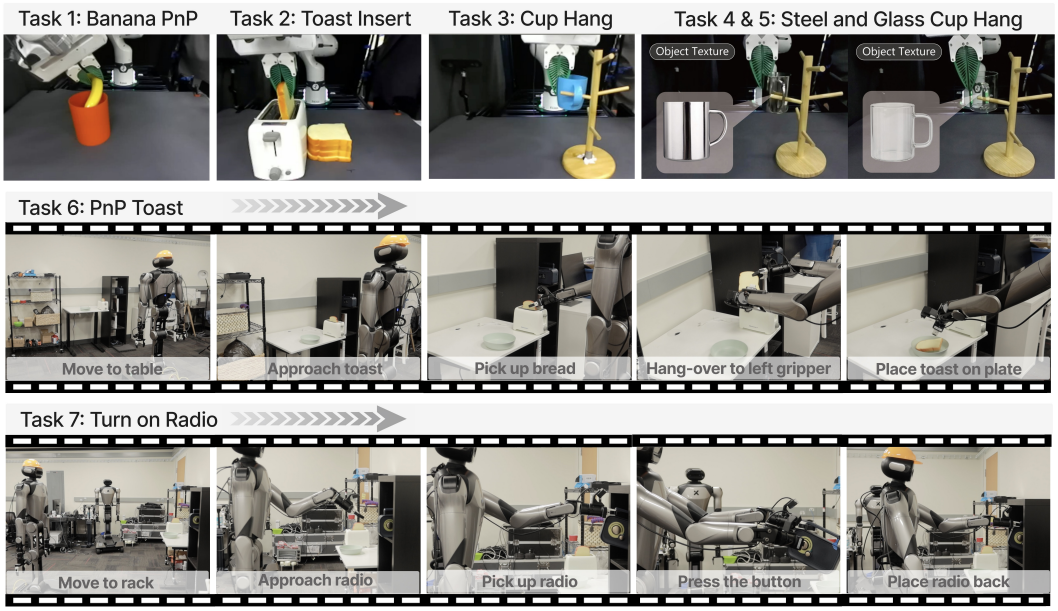


Figure 3: **Real-World Task Visualizations**. *Top*: Five tabletop tasks, with the final state shown for each. Tasks 3–5 vary by cup material. *Bottom*: Two mobile manipulation tasks.

the input to the policy backbone to predict robot actions:

$$z_{\text{obs}} = [\, z_{t,1} \| z_{t,2} \| \cdots \| z_{t,N} \| s_t \,]. \tag{3}$$

2.2 StereoPolicy-DP: Diffusion Policy with StereoPolicy

For imitation learning, we primarily adopt a diffusion policy (Chi et al., 2024) as the policy backbone.

The policy predicts a sequence of continuous actions $\mathbf{a}_{t:t+H-1} = [\mathbf{a}_t, \mathbf{a}_{t+1}, \ldots, \mathbf{a}_{t+H-1}]$ over a planning horizon H based on z_{obs}. In StereoPolicy-DP, we inject stereo perception by conditioning the denoising network on the fused stereo embedding z_{obs}, which combines the fused stereo-view features. Following Diffusion Policy (Chi et al., 2024), we train the noise predictor conditioned on z_{obs} with the denoising objective

$$\mathcal{L}_{\text{DP}} = \mathbb{E}\left[\left\| \epsilon - \epsilon_\theta(x^{(\tau)}, \tau, z_{\text{obs}}) \right\|_2^2 \right], \tag{4}$$

where $\epsilon \sim \mathcal{N}(0, I)$ and $x^{(\tau)}$ is the demonstration action sequence corrupted by noise at diffusion step τ. The expectation is over demonstration samples, diffusion steps, and Gaussian noise. The trainable vision encoder, Stereo Transformer, and policy backbone are optimized jointly, while DINOv2 remains frozen. At inference, the policy generates actions by iteratively denoising an initial Gaussian sample. This design allows StereoPolicy-DP to enhance action generation with implicit stereo cues while preserving the original diffusion policy architecture and training objective.

2.3 StereoPolicy-VLA: Adapting Monocular VLA to Stereo Inputs

Pretrained VLA models exhibit strong semantic understanding through VLM pretraining, but their depth reasoning is limited by monocular inputs. To improve spatial perception, we extend the visual input from monocular to stereo by introducing a lightweight stereo feature extractor on top of the pretrained vision encoder, without modifying the VLA backbone architecture.

Concretely, we replace the monocular visual embedding with a stereo-aware representation. For each stereo image pair, we extract features with the pretrained vision encoder from VLA, then inject the fused stereo representation from the Stereo Transformer into the VLA backbone to predict actions:

$$\mathbf{a} = \pi_\theta(z_{\text{obs}}, x_{\text{lang}}, s), \tag{5}$$

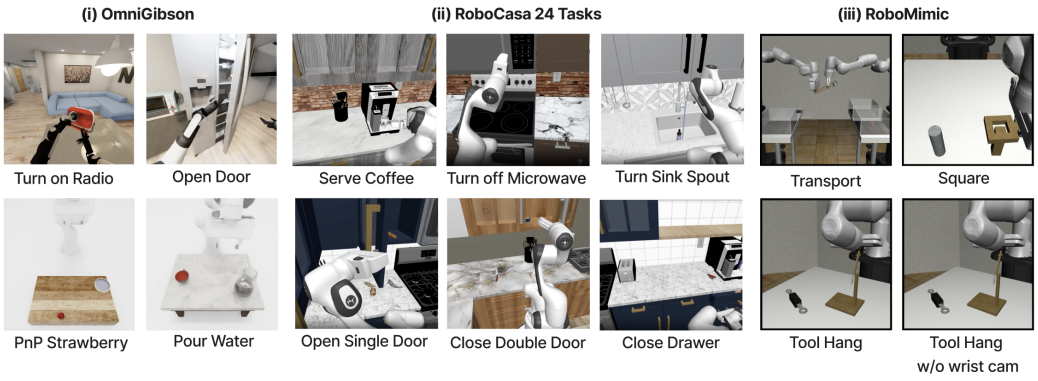


Figure 4: **Simulation Task Visualization**, from three benchmarks: OMNIGIBSON (4 tasks), ROBOCASA (24 tasks), and ROBOMIMIC (3 tasks).

where π_θ denotes the VLA policy (e.g., DiT), z_{obs} is the fused stereo feature from StereoTrans, x_{lang} is the language instruction, and s denotes additional state.

3 EXPERIMENTS

Our experiments were guided by the following research questions:

$\mathcal{Q}\mathbf{1}$. How does StereoPolicy perform compared to monocular RGB, RGB-D, point cloud, and multi-view-based policies under the same policy backbone and training data?

$\mathcal{Q}\mathbf{2}$. Can StereoPolicy be readily combined with large pretrained policy models, such as vision-language-action (VLA) models, although these models are trained on monocular RGB data?

$\mathcal{Q}\mathbf{3}$. How do factors such as stereo baseline (distance between two cameras), distance to the objects, and camera angles affect StereoPolicy's performance?

$\mathcal{Q}\mathbf{4}$. Can large-scale pretrained stereo models replace task-trained stereo fusion from StereoPolicy?

3.1 TASKS AND ENVIRONMENTS

We evaluate StereoPolicy on two robotic embodiments: a 7-DoF tabletop Franka arm and a 24-DoF Galaxea R1 Pro bimanual mobile manipulator with two 7-DoF arms, in both real-world and simulated settings. Task and data collection details are summarized in Appendix A.

Real-World We design seven tasks spanning both tabletop and mobile manipulation (Figure 3).

Simulation We include tasks from three widely used robotics benchmarks, including ROBOMIMIC (Mandlekar et al., 2021), ROBOCASA (Nasiriany et al., 2024), and four customized tasks in OMNIGIBSON (Li et al., 2024). Figure 4 visualizes tasks.

3.2 IMPLEMENTATION DETAILS

Baselines We compare STEREOPOLICY-DP against six baseline architectures: monocular RGB, multi-view RGB, RGB-D, point-clouds with PointNet (Qi et al., 2017a). To compare STEREOPOLICY-DP with policies that more effectively utilize depth and point clouds, we also implement RGBD-3DDA (Ke et al., 2024) and PCD-DP3 (Ze et al., 2024). All baselines use the same policy backbones, image resolution, training budgets, and optimization hyperparameters where applicable.

To effectively evaluate STEREOPOLICY-VLA, we adopt $\pi_{0.5}$ (Physical Intelligence et al., 2025) and GR00T-N1.5 (NVIDIA et al., 2025) as our baseline VLA frameworks and follow the same settings from the original implementation. The StereoPolicy architecture, baseline implementations, training and evaluation details are described in Appendix C.

Method	Banana PnP	Toast Insert	Plastic Cup Hang	Steel Cup Hang	Glass Cup Hang	AVG SR (%)
RGB	12/20	8/20	12/20	10/20	1/20	43.0%
RGB-D	14/20	8/20	11/20	8/20	0/20	41.0%
RGBD-3DDA	13/20	9/20	13/20	10/20	0/20	45.0%
PCD-PointNet	7/20	0/20	5/20	2/20	0/20	14.0%
PCD-DP3	11/20	3/20	8/20	5/20	0/20	27.0%
Multi-View	13/20	9/20	13/20	9/20	1/20	45.0%
StereoPolicy-DP	**16/20**	**12/20**	**15/20**	**13/20**	**3/20**	**59.0%**

Table 1: **Real-World Tabletop Task Performance**. StereoPolicy-DP consistently outperforms other visual modalities. PCD performs worst in all real tasks; both RGB-D and PCD fail on glass cup hang tasks, demonstrating that explicit 3D methods are sensitive to object texture in practice.

Benchmark	OmniGibson								RoboMimic					
Task	Strawberry		PourWater		OpenDoor		TurnOnRadio		ToolHang		Square		Transport	
# Demos	100	200	100	200	200	300	200	300	100	200	100	200	100	200
RGB	$62_{3.3}$	$86_{1.6}$	$15_{4.1}$	$50_{2.8}$	$30_{3.3}$	$80_{1.6}$	$46_{2.8}$	$74_{2.8}$	$53_{3.4}$	$90_{1.6}$	$76_{1.9}$	$98_{0.9}$	$92_{1.9}$	$94_{1.6}$
RGB-D	$63_{3.8}$	$85_{4.7}$	$16_{1.6}$	$52_{0.0}$	$31_{2.5}$	$80_{4.3}$	$47_{0.9}$	$73_{3.4}$	$56_{0.0}$	$88_{2.5}$	$79_{1.9}$	$92_{0.0}$	$\mathbf{94}_{1.6}$	$94_{1.6}$
RGBD-3DDA	$74_{4.3}$	$93_{1.9}$	$26_{0.0}$	$61_{3.4}$	$48_{3.3}$	$\mathbf{100}_{0.0}$	$46_{4.3}$	$75_{4.1}$	$84_{1.6}$	$92_{2.8}$	$83_{1.9}$	$97_{2.5}$	$\mathbf{94}_{1.6}$	$\mathbf{96}_{1.6}$
PCD-DP3	$45_{3.4}$	$63_{2.5}$	$3_{4.7}$	$31_{0.9}$	$30_{1.6}$	$69_{4.1}$	$35_{1.9}$	$64_{4.9}$	$40_{1.6}$	$76_{0.9}$	$69_{2.5}$	$88_{1.6}$	$63_{2.5}$	$72_{2.8}$
Multi-View	$70_{3.3}$	$91_{2.5}$	$24_{3.4}$	$56_{3.3}$	$35_{4.3}$	$82_{2.8}$	$47_{3.4}$	$77_{2.5}$	$64_{2.8}$	$92_{1.9}$	$80_{1.6}$	$96_{0.9}$	$92_{1.9}$	$94_{1.6}$
StereoPolicy-DP	$\mathbf{82}_{2.8}$	$\mathbf{100}_{0.0}$	$\mathbf{34}_{1.6}$	$\mathbf{70}_{2.5}$	$\mathbf{57}_{1.9}$	$\mathbf{100}_{0.0}$	$\mathbf{55}_{3.4}$	$\mathbf{82}_{4.3}$	$\mathbf{94}_{1.9}$	$\mathbf{96}_{2.8}$	$\mathbf{88}_{0.9}$	$\mathbf{100}_{0.0}$	$\mathbf{94}_{0.0}$	$\mathbf{96}_{1.6}$

Table 2: **Simulation Task Performance of Diffusion Policies across Visual Modalities**. Stereo input consistently improves performance, especially under low-data regime. We report average scores and standard deviations across three seed runs. All numbers are reported in percentage (%).

4 Results and Analysis

4.1 Experimental Results

($\mathcal{Q}1$) StereoPolicy-DP consistently outperforms monocular RGB, RGB-D, point cloud, and multi-view-based policies. Table 1 and Table 2 compare StereoPolicy-DP with standard visual modalities in real-robot and simulation respectively. Overall, StereoPolicy-DP presents strong performance in both settings. In particular, StereoPolicy-DP outperforms RGB across all tasks, highlighting the benefit of stereo inputs for manipulation. For less depth-sensitive tasks (e.g., *Transport*), monocular RGB inputs already provide sufficient visual information for task success. For tasks involving occlusion and fine-grained 3D alignment (e.g., *ToolHang* and *PourWater*), StereoPolicy-DP shows clear improvements [1].

However, point cloud–based methods perform poorly overall: depth measurements are often noisy, leading to reconstructed point clouds with missing or distorted task-relevant geometry (see Figure 7). Even with access to clean depth in simulation, DP3 still underperforms on most benchmarks. We also observe point-cloud methods are fragile and heavily rely on careful pre-processing. Without careful cropping and sampling, their performance can easily drop to 0. We additionally discuss the failure reason of point-clouds policies in Appendix D.5. While geometry-aware models such as RGBD-3DDA achieve competitive results in simulation due to perfect depth, their performance degrades notably on reflective and transparent objects (e.g., steel and glass cup hang), where depth sensing becomes unreliable; their reliance on calibrated depth sensors also limits robustness. In contrast, StereoPolicy-DP achieves higher success from image pairs in these evaluated settings, although transparent-object performance re-

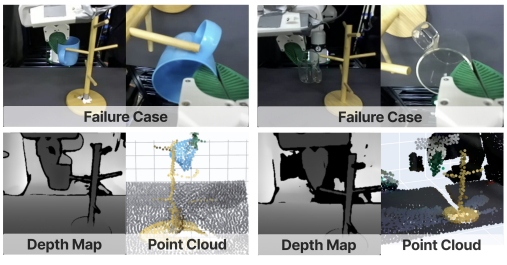


Figure 7: RGB-D and PCD are fragile in real environments. The glass cup is entirely missing.

[1] We present the inference latency of RGB, RGBD-3DDA and StereoPolicy-DP in Appendix D.7.

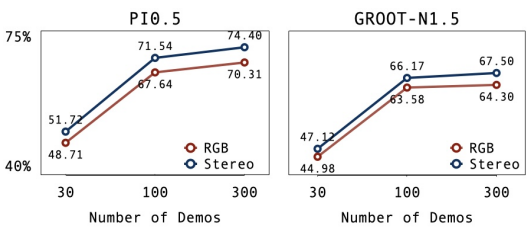


Figure 5: Average STEREOPOLICY-VLA Performance on ROBOCASA-KITCHEN 24 Tasks.

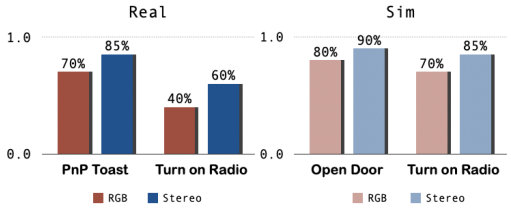


Figure 6: STEREOPOLICY-VLA ($\pi_{0.5}$) Performance on bimanual mobile manipulation tasks in both real-world and simulation.

mains low in absolute terms. STEREOPOLICY-DP also outperforms RGBD-3DDA in most evaluated simulation settings, demonstrating the effectiveness of stereo inputs for manipulation without explicit depth or point-cloud reconstruction.

STEREOPOLICY-DP outperforms multi-view-based policies. Our parameter-matched multi-view RGB baselines also use stereo images but concatenate their features without a Stereo Transformer, supporting the effectiveness of our stereo fusion. Therefore, the performance gain of STEREOPOLICY-DP is not solely from the additional view, but from the stereo fusion design.

STEREOPOLICY-DP is more data-efficient. In OMNIGIBSON tasks, STEREOPOLICY-DP reaches higher SR with fewer demonstrations. Although Peri et al. (2024) claims point-cloud inputs can yield higher sample efficiency compared to RGB/RGB-D, here we observe the opposite.

($\mathcal{Q}2$) StereoPolicy enhances pretrained VLA models, although these models are trained on monocular data. We next examine STEREOPOLICY-VLA, where StereoPolicy is incorporated into two state-of-the-art pretrained VLA models, $\pi_{0.5}$ and GR00T-N1.5, for fine-tuning. As summarized in Figure 5, incorporating stereo input and stereo fusion module yields consistent improvements across both high-data (300 and 100 demos) and low-data (30 demos) settings. These gains, while modest, support our earlier claim: stereo features can provide geometric cues complementary to pretrained VLM representations, and fine-tuning allows these models to combine stereo cues with pretrained semantic features. We also compare STEREOPOLICY-VLA with a VLA that directly takes the two stereo images as its multi-view input in Appendix Table 9.

Figure 6 further evaluates STEREOPOLICY-VLA on mobile manipulation. STEREOPOLICY-VLA outperforms monocular baselines in both simulation and real-world settings. In particular, RGB policies fail on precise actions such as inserting the gripper into a radio handle and pressing buttons accurately (failure case visualization in Appendix), while STEREOPOLICY-VLA succeeds.

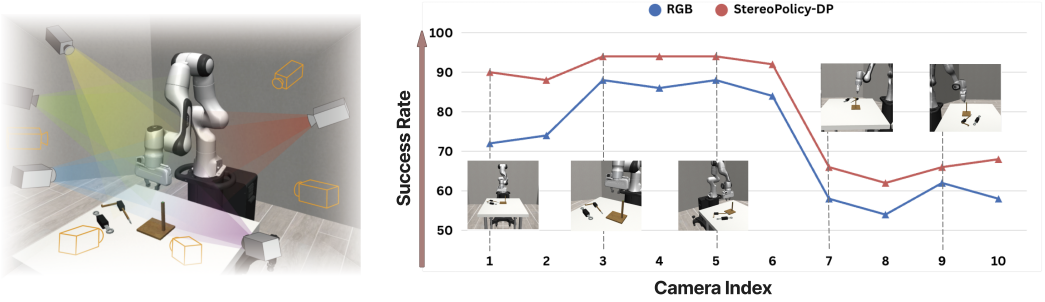
($\mathcal{Q}3$) STEREOPOLICY-DP is most effective when the baseline is approximately 10% of the target object distance. We vary the stereo baseline distance (2 cm, 4 cm, 6 cm, 8 cm, 10 cm) and camera–object distance (0.6 m–1.0 m) while keeping other conditions fixed. Results show that performance is governed not by either factor alone, but by their ratio:

$$r = \frac{\text{stereo baseline}}{\text{camera-object distance}} \tag{6}$$

which determines the effective disparity. STEREOPOLICY-DP achieves the best performance when r lies in the range of $r \in [0.09, 0.13]$, as shown in Figure 9.

Within this range, disparity is strong enough for reliable depth estimation while preserving sufficient overlap between left and right views. For typical tabletop setups (0.6–0.8m camera-object distance), a 6 cm baseline tends to perform better than smaller or larger ones.

When r becomes too small (e.g., 2 cm baseline or > 0.9 m distance), performance drops due to weak disparity and poor depth discrimination. Conversely, excessively large r values (e.g., 10 cm baseline at 0.6 m) reduce view overlap and cause geometric inconsistency across views. In conclusion, stereo policies are most effective when the baseline is approximately 10% of the target distance. This ratio maintains an optimal, balanced trade-off between disparity and overlap. It allows STEREOPOLICY-DP to extract stable geometric features and improves manipulation accuracy across viewpoints.



(a) **Camera Angle Visualization**. From left to right: *sideview left, agentview left, frontview, agentview right, and sideview right*.

(b) **SR across 10 Different Camera Angle Views** over TOOLHANG task for RGB policy and STEREOPOLICY-DP. STEREOPOLICY-DP *is more robust towards changing camera angles than RGB baseline*. Appendix B provides full mapping of camera indices.

Figure 8: Performance of STEREOPOLICY-DP across different camera angles.

In addition, STEREOPOLICY-DP consistently outperforms RGB-based policies across different camera angles. We evaluate 10 camera angles on the TOOLHANG task, visualized in Figure 8a; Figure 8b shows the corresponding views and performance. The largest gains appear in front-facing views (*frontview*, +18%; *frontview-high*, +14%), where stereo disparity provides complementary depth cues. Oblique views such as *agentview-left/right* and their elevated counterparts also benefit, as stereo helps resolve partial occlusions and spatial correspondence. Side views show smaller gains due to limited overlap and stronger self-occlusion. Overall, stereo is most helpful when monocular cues are geometrically ambiguous, and it helps in all ten tested views.

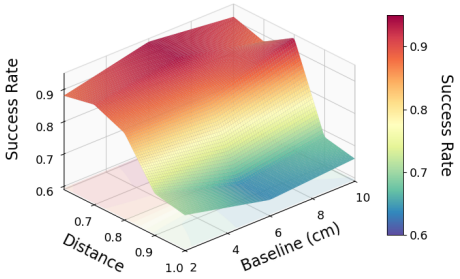


Figure 9: Effect of stereo baseline and camera–object distance on task success rate.

Method	SR (%)	Latency (ms)
STEREOPOLICY-DP	$94_{1.9}$	698
A. Pretrained stereo features		
Frozen FS features	$76_{4.3}$	740
B. Stereo-derived geometry		
FS depth + RGB-D	$44_{4.3}$	995
FastFS depth + RGB-D	$38_{3.3}$	792
FS depth + 3DDA	$28_{2.8}$	1008
FastFS depth + 3DDA	$22_{2.8}$	777

Table 3: Comparison with pre-trained stereo foundation models on ToolHang (100 demo). FS denotes FoundationStereo. Subscripts are std over three seeds.

(*Q*4) Pre-trained Stereo features cannot replace task-adapted stereo fusion. Table 3 compares STEREOPOLICY-DP with two ways of using a pretrained stereo model on ToolHang with 100 demonstrations, keeping the diffusion-policy backbone fixed. First, following StereoVLA (Deng et al., 2025), we replace our trainable encoder and Stereo Transformer with frozen FoundationStereo features. Success decreases from 94% to 76%, indicating that a generic pretrained stereo representation does not directly substitute for task-adapted fusion in this setting. Second, we replace the depth source in RGBD and 3DDA policies by depth inferred with (Fast) FoundationStereo. However, they incur substantially higher end-to-end latency while their success rate degrades greatly, likely because depth errors introduced by FS affect the reconstructed 3D coordinates. On the real robot, the stale observation caused by this latency becomes a more significant practical problem (Appendix D.3).

4.2 ABLATIONS

We examine which part of STEREOPOLICY-DP contributes most to the success rate. Figure 10 removes one component at a time (numbers in Appendix Table 5). The largest drop comes from removing binocular information. When the left image is fed to both branches, success falls from 84.0% to 62.7%. DINOv2 helps, but it is not the source of the stereo gain.

Without it, the policy still reaches 76% on average and stays ahead of both single-image controls on every task. Removing the Stereo Transformer hurts significantly on most tasks. On these tasks, the second image adds little unless the two views are fused through cross-attention. The FPN features are also important for stereo feature extraction. We also explore how to appropriately use DINO features. Applying DINO to the wrist images hurts all tasks' success, likely due to a domain mismatch. Therefore, we apply DINO only on the external view. Together with the additional ablations in Appendix D, these results indicate that the gains come mainly from stereo fusion, suggesting StereoPolicy-DP learns stereo geometry.

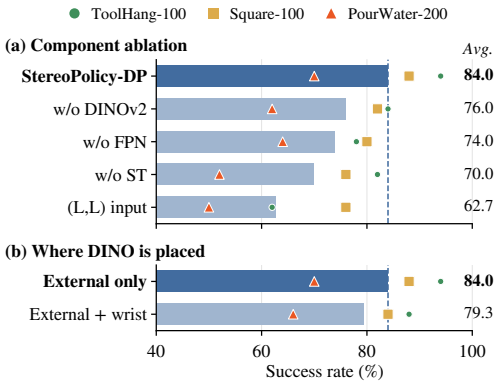


Figure 10: **Component Ablation**. ST denotes StereoTransformer.

5 Related Work

Stereo Vision Stereo matching recovers 3D structure from rectified image pairs. Modern methods estimate disparity or depth with cost volumes, multi-scale features, and recurrent refinement (Wen et al., 2025; Min et al., 2025; Bartolomei et al., 2025; Xu et al., 2025; Lipson et al., 2021; Chang & Chen, 2018), and recent models transfer zero-shot to challenging conditions (Wen et al., 2025; Wang et al., 2025a). In robotics, stereo-based manipulation systems often use such models to reconstruct depth first (Shankar et al., 2022; Bai et al., 2024; Kollar et al., 2021), while stereo grasping policies can learn directly from images (Singh et al., 2026; Lum et al., 2024). Concurrent with our work, StereoVLA (Deng et al., 2025) combines pretrained FoundationStereo features with semantic features and adds auxiliary depth and camera-parameter supervision. StereoPolicy differs from both lines of work: it learns task-conditioned binocular features directly through the action objective, on top of generic pretrained 2D encoders, without explicit disparity estimation, stereo-specific pretraining, or depth supervision.

3D-aware Policies and VLAs Many policies take explicit 3D input. DP3 (Ze et al., 2024) and FP3 (Yang et al., 2025) encode reconstructed point clouds, Act3D (Gervet et al., 2023) and 3D Diffuser Actor (Ke et al., 2024) lift 2D features into 3D with depth and camera calibration, and RVT (Goyal et al., 2023) and RVT-2 (Goyal et al., 2024) re-render point clouds into virtual views. Similar ideas have been applied to VLAs via point-cloud features (Li et al., 2025a), projected point-cloud images (Li et al., 2025b), and 3D position encodings from monocular depth estimates (Qu et al., 2025). These methods depend on depth sensing and calibration or on estimated depth, whereas StereoPolicy learns geometric cues end-to-end from a rectified stereo pair.

Multi-view Policy Learning Recent multi-view policies go beyond concatenating per-view features and model cross-view structure explicitly, e.g., by rendering a fused RGB-D point cloud into virtual views for a video diffusion model (Li et al., 2026), or by reconstructing a shared 3D Gaussian field from several agents' cameras (Wang et al., 2025b). StereoPolicy addresses a different and complementary problem: it exploits dense correspondence between two nearly identical rectified images from a single viewpoint without explict depth or pcd reconstruction. Because stereo fusion occurs within each camera rig, it could be combined with cross-view aggregation as in those works.

6 Conclusion

We presented StereoPolicy, a visuomotor learning framework that directly consumes synchronized stereo image pairs instead of explicit 3D reconstructions. By combining pretrained 2D vision encoders with a cross-attention-based stereo fusion module, StereoPolicy improves over monocular RGB, RGB-D, point-cloud, and multi-view baselines across both simulated and real-world settings, spanning tabletop and mobile manipulation tasks. Our analysis further identifies practical design choices for stereo-based policies, including camera baseline, viewpoint, vision backbone, and fusion architecture. StereoPolicy also improves pretrained VLA models, suggesting that stereo cues can complement large-scale 2D visual representations for robot learning without explicit 3D supervision.

Reproducibility Statement

We provide detailed descriptions of the StereoPolicy architecture, training objectives, camera configurations, task setups, baseline implementations, optimization settings, and evaluation protocols in the main paper and appendices. StereoPolicy is evaluated across three simulation benchmarks and seven real-robot tasks, with simulation results reported across three training seeds and real-robot performance evaluated over repeated trials with randomized initial poses. The appendices further include matched architectural ablations, duplicated-versus-stereo input controls, alternative fusion mechanisms, model size, computational cost, hardware configurations, and inference latency measurements. We will release all code, training configurations, and model checkpoints upon acceptance.

AI Use Statement

We used large language models as writing assistants to polish the language and improve the clarity of the manuscript, and to help write a small number of plotting scripts for figure visualizations. All research ideas, method design, experiments, and analyses were carried out by the authors. The authors reviewed and verified all AI-assisted text and code, checked every reported number against the underlying experimental records, and take full responsibility for the content of this paper.

References

Kaixin Bai, Huajian Zeng, Lei Zhang, Yiwen Liu, Hongli Xu, Zhaopeng Chen, and Jianwei Zhang. ClearDepth: Enhanced stereo perception of transparent objects for robotic manipulation. *arXiv preprint arXiv:2409.08926*, 2024. URL https://arxiv.org/abs/2409.08926.

Luca Bartolomei, Fabio Tosi, Matteo Poggi, and Stefano Mattoccia. Stereo Anywhere: Robust zero-shot deep stereo matching even where either stereo or mono fail. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1013–1027, 2025. doi: 10.1109/CVPR52734.2025.00103. URL https://arxiv.org/abs/2412.04472.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, Laura Smith, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. In *Proceedings of Robotics: Science and Systems (RSS)*, 2025. doi: 10.15607/RSS.2025.XXI.010. URL https://arxiv.org/abs/2410.24164.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Marcin Choromanski, Tianli Ding, Danny Driess, Kumar Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montserrat Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alexander Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil J. Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael S. Ryoo, Grecia Salazar, Pannag R. Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. RT-2: Vision-language-action models transfer web knowledge to robotic control. In *Proceedings of the 7th Conference on Robot Learning (CoRL)*, volume 229 of *Proceedings of Machine Learning Research*, pp. 2165–2183. PMLR, 2023a. URL https://proceedings.mlr.press/v229/zitkovich23a.html.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alexander Herzog, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Tomas Jackson, Sally Jesmonth, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Kuang-Huei Lee, Sergey Levine, Yao Lu, Utsav Malla, Deeksha Manjunath, Igor Mordatch, Ofir Nachum, Carolina Parada, Jodilyn Peralta, Emily Perez, Karl Pertsch, Jornell Quiambao, Kanishka Rao, Michael S. Ryoo, Grecia Salazar, Pannag R. Sanketi, Kevin Sayed, Jaspiar Singh, Sumedh Sontakke, Austin Stone, Clayton Tan, Huong Tran, Vincent

Vanhoucke, Steve Vega, Quan H. Vuong, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. RT-1: Robotics transformer for real-world control at scale. In *Proceedings of Robotics: Science and Systems (RSS)*, 2023b. doi: 10.15607/RSS.2023.XIX.025. URL https://arxiv.org/abs/2212.06817.

Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5410–5418, 2018. doi: 10.1109/CVPR.2018.00567.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion Policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2024. doi: 10.1177/02783649241273668.

Shengliang Deng, Mi Yan, Yixin Zheng, Jiayi Su, Wenhao Zhang, Yitao Zeng, Xiaoguang Zhao, Heming Cui, Zhizheng Zhang, and He Wang. Stereovla: Enhancing vision-language-action models with stereo vision. *arXiv preprint arXiv:2512.21970*, 2025. URL https://arxiv.org/abs/2512.21970.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361, 2012. doi: 10.1109/CVPR.2012.6248074.

Theophile Gervet, Zhou Xian, Nikolaos Gkanatsios, and Katerina Fragkiadaki. Act3D: 3D feature field transformers for multi-task robotic manipulation. In *Proceedings of The 7th Conference on Robot Learning (CoRL)*, volume 229 of *Proceedings of Machine Learning Research*, pp. 3949–3965, 2023.

Ankit Goyal, Jie Xu, Yijie Guo, Valts Blukis, Yu-Wei Chao, and Dieter Fox. RVT: Robotic view transformer for 3D object manipulation. In *Conference on Robot Learning (CoRL)*, pp. 694–710, 2023. URL https://proceedings.mlr.press/v229/goyal23a.html.

Ankit Goyal, Valts Blukis, Jie Xu, Yijie Guo, Yu-Wei Chao, and Dieter Fox. RVT-2: Learning precise manipulation from few demonstrations. In *Robotics: Science and Systems (RSS)*, 2024.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Byeongho Heo, Song Park, Dongyoon Han, and Sangdoo Yun. Rotary position embedding for vision transformer. In *European Conference on Computer Vision (ECCV)*, pp. 289–305, 2024. doi: 10.1007/978-3-031-72684-2_17. URL https://arxiv.org/abs/2403.13298.

Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 4651–4664. PMLR, 2021. URL https://proceedings.mlr.press/v139/jaegle21a.html.

Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. VIMA: Robot manipulation with multimodal prompts. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 14975–15022. PMLR, 2023. URL https://proceedings.mlr.press/v202/jiang23b.html.

Yunfan Jiang, Chen Wang, Ruohan Zhang, Jiajun Wu, and Li Fei-Fei. TRANSIC: Sim-to-real policy transfer by learning from online correction. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 1691–1729. PMLR, 2024. URL https://proceedings.mlr.press/v270/jiang25a.html.

Yunfan Jiang, Ruohan Zhang, Josiah Wong, Chen Wang, Yanjie Ze, Hang Yin, Cem Gokmen, Shuran Song, Jiajun Wu, and Li Fei-Fei. BEHAVIOR robot suite: Streamlining real-world whole-body manipulation for everyday household activities. In *9th Annual Conference on Robot Learning*, 2025. URL https://openreview.net/forum?id=v2KevjWScT.

Tsung-Wei Ke, Nikolaos Gkanatsios, and Katerina Fragkiadaki. 3D Diffuser Actor: Policy diffusion with 3D scene representations. In *Conference on Robot Learning (CoRL)*, pp. 1949–1974, 2024. URL https://proceedings.mlr.press/v270/ke25a.html.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model. In *8th Annual Conference on Robot Learning*, 2024. URL https://openreview.net/forum?id=ZMnD6QZAE6.

Thomas Kollar, Michael Laskey, Kevin Stone, Brijen Thananjeyan, and Mark Tjersland. SimNet: Enabling robust unknown object manipulation from pure synthetic data via stereo. In *Proceedings of the 5th Conference on Robot Learning (CoRL)*, volume 164 of *Proceedings of Machine Learning Research*, pp. 938–948. PMLR, 2021. URL https://proceedings.mlr.press/v164/kollar22a.html.

Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17(39):1–40, 2016. URL https://jmlr.org/papers/v17/15-522.html.

Chengmeng Li, Junjie Wen, Yan Peng, Yaxin Peng, Feifei Feng, and Yichen Zhu. Pointvla: Injecting the 3d world into vision-language-action models. *arXiv preprint arXiv:2503.07511*, 2025a. URL https://arxiv.org/abs/2503.07511.

Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Wensi Ai, Benjamin Martinez, Hang Yin, Michael Lingelbach, Minjune Hwang, Ayano Hiranaka, Sujay Garlanka, Arman Aydin, Sharon Lee, Jiankai Sun, Mona Anvari, Manasi Sharma, Dhruva Bansal, Samuel Hunter, Kyu-Young Kim, Alan Lou, Caleb R Matthews, Ivan Villa-Renteria, Jerry Huayang Tang, Claire Tang, Fei Xia, Yunzhu Li, Silvio Savarese, Hyowon Gweon, C. Karen Liu, Jiajun Wu, and Li Fei-Fei. BEHAVIOR-1K: A human-centered, embodied AI benchmark with 1,000 everyday activities and realistic simulation. *arXiv preprint arXiv:2403.09227*, 2024.

Jiankun Li, Peisen Wang, Pengfei Xiong, Tao Cai, Ziwei Yan, Lei Yang, Jiangyu Liu, Haoqiang Fan, and Shuaicheng Liu. Practical stereo matching via cascaded recurrent network with adaptive correlation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16263–16272, June 2022.

Peiyan Li, Yixiang Chen, Hongtao Wu, Xiao Ma, Xiangnan Wu, Yan Huang, Liang Wang, Tao Kong, and Tieniu Tan. BridgeVLA: Input-output alignment for efficient 3D manipulation learning with vision-language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025b. URL https://openreview.net/forum?id=ffBF6hYuQv.

Peiyan Li, Yixiang Chen, Yuan Xu, Jiabing Yang, Xiangnan Wu, Jun Guo, Nan Sun, Long Qian, Xinghang Li, Xin Xiao, Jing Liu, Nianfeng Liu, Tao Kong, Yan Huang, Liang Wang, and Tieniu Tan. Multi-view video diffusion policy: A 3D spatio-temporal-aware video action model. *arXiv preprint arXiv:2604.03181*, 2026. URL https://arxiv.org/abs/2604.03181.

Zhaoshuo Li, Xingtong Liu, Nathan Drenkow, Andy Ding, Francis X. Creighton, Russell H. Taylor, and Mathias Unberath. Revisiting stereo depth estimation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6197–6206, October 2021.

Lahav Lipson, Zachary Teed, and Jia Deng. RAFT-Stereo: Multilevel recurrent field transforms for stereo matching. In *International Conference on 3D Vision (3DV)*, pp. 218–227, 2021. doi: 10.1109/3DV53792.2021.00032. URL https://arxiv.org/abs/2109.07547.

Minghuan Liu, Zixuan Chen, Xuxin Cheng, Yandong Ji, Ri-Zhao Qiu, Ruihan Yang, and Xiaolong Wang. Visual whole-body control for legged loco-manipulation. In *Proceedings of The 8th Conference on Robot Learning (CoRL)*, volume 270, pp. 234–257. PMLR, 2024. URL https://arxiv.org/abs/2403.16967.

Tyler Ga Wei Lum, Martin Matak, Viktor Makoviychuk, Ankur Handa, Arthur Allshire, Tucker Hermans, Nathan D. Ratliff, and Karl Van Wyk. DextrAH-G: Pixels-to-action dexterous arm-hand grasping with geometric fabrics. In *Proceedings of The 8th Conference on Robot Learning (CoRL)*, volume 270, pp. 3182–3211. PMLR, 2024. URL https://arxiv.org/abs/2407.02274.

Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In *Proceedings of the 5th Conference on Robot Learning (CoRL)*, volume 164 of *Proceedings of Machine Learning Research*, pp. 1678–1690. PMLR, 2021.

D. Marr and T. Poggio. Cooperative computation of stereo disparity. *Science*, 194(4262):283–287, 1976. doi: 10.1126/science.968482. URL https://www.science.org/doi/abs/10.1126/science.968482.

Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

Junhong Min, Youngpil Jeon, Jimin Kim, and Minyong Choi. S^2M^2: Scalable stereo matching model for reliable depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 26729–26739, 2025. URL https://arxiv.org/abs/2507.13229.

Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. RoboCasa: Large-scale simulation of household tasks for generalist robots. In *Robotics: Science and Systems (RSS)*, 2024.

NVIDIA, Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhenjia Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. GR00T N1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025. URL https://arxiv.org/abs/2503.14734.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=a68SUt6zFt.

Skand Peri, Iain Lee, Chanho Kim, Li Fuxin, Tucker Hermans, and Stefan Lee. Point cloud models improve visual robustness in robotic learners. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 17529–17536, 2024. doi: 10.1109/ICRA57147.2024.10610710. URL https://arxiv.org/abs/2404.18926.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z.

Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization, 2025. URL https://arxiv.org/abs/2504.16054.

Matteo Poggi, Filippo Aleotti, Fabio Tosi, and Stefano Mattoccia. On the uncertainty of self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017a. doi: 10.1109/CVPR.2017.16. URL https://arxiv.org/abs/1612.00593.

Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. PointNet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017b.

Yuzhe Qin, Binghao Huang, Zhao-Heng Yin, Hao Su, and Xiaolong Wang. DexPoint: Generalizable point cloud reinforcement learning for sim-to-real dexterous manipulation. In *Conference on Robot Learning (CoRL)*, volume 205 of *Proceedings of Machine Learning Research*, pp. 594–605, 2022. URL https://proceedings.mlr.press/v205/qin23a.html.

Delin Qu, Haoming Song, Qizhi Chen, Yuanqi Yao, Xinyi Ye, Yan Ding, Zhigang Wang, JiaYuan Gu, Bin Zhao, Dong Wang, and Xuelong Li. SpatialVLA: Exploring spatial representations for visual-language-action model. In *Robotics: Science and Systems (RSS)*, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gómez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Giménez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, Tom Eccles, Jake Bruce, Ali Razavi, Ashley Edwards, Nicolas Heess, Yutian Chen, Raia Hadsell, Oriol Vinyals, Mahyar Bordbar, and Nando de Freitas. A generalist agent. *Transactions on Machine Learning Research*, 2022. URL https://openreview.net/forum?id=1ikK0kHjvj.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, volume 9351 of *Lecture Notes in Computer Science*, pp. 234–241. Springer, 2015. doi: 10.1007/978-3-319-24574-4_28.

Daniel Scharstein, Heiko Hirschmüller, York Kitajima, Greg Krathwohl, Nera Nešić, Xi Wang, and Porter Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In Xiaoyi Jiang, Joachim Hornegger, and Reinhard Koch (eds.), *Pattern Recognition*, pp. 31–42, Cham, 2014. Springer International Publishing. ISBN 978-3-319-11752-2.

Krishna Shankar, Mark Tjersland, Jeremy Ma, Kevin Stone, and Max Bajracharya. A learned stereo depth system for robotic manipulation in homes. *IEEE Robotics and Automation Letters*, 7(2): 2305–2312, 2022. doi: 10.1109/LRA.2022.3143895.

Zhelun Shen, Yuchao Dai, and Zhibo Rao. CFNet: Cascade and fused cost volume for robust stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13906–13915, 2021. URL https://openaccess.thecvf.com/content/CVPR2021/html/Shen_CFNet_Cascade_and_Fused_Cost_Volume_for_Robust_Stereo_Matching_CVPR_2021_paper.html.

Mohit Shridhar, Lucas Manuelli, and Dieter Fox. CLIPort: What and where pathways for robotic manipulation. In *Proceedings of the 5th Conference on Robot Learning (CoRL)*, volume 164 of *Proceedings of Machine Learning Research*, pp. 894–906. PMLR, 2021.

Ritvik Singh, Arthur Allshire, Ankur Handa, Nathan Ratliff, and Karl Van Wyk. DextrAH-RGB: Visuomotor policies to grasp anything with dexterous hands. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2026.

Balakumar Sundaralingam, Siva Kumar Sastry Hari, Adam Fishman, Caelan Garrett, Karl Van Wyk, Valts Blukis, Alexander Millane, Helen Oleynikova, Ankur Handa, Fabio Ramos, Nathan Ratliff, and Dieter Fox. cuRobo: Parallelized collision-free minimum-jerk robot motion generation, 2023.

Priya Sundaresan, Rhea Malhotra, Phillip Miao, Jingyun Yang, Jimmy Wu, Hengyuan Hu, Rika Antonova, Francis Engelmann, Dorsa Sadigh, and Jeannette Bohg. HoMeR: Learning in-the-wild mobile manipulation via hybrid imitation and whole-body control. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2026. arXiv:2506.01185.

Hugues Thomas, Charles R. Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, Francois Goulette, and Leonidas J. Guibas. KPConv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Shagun Uppal, Ananye Agarwal, Haoyu Xiong, Kenneth Shaw, and Deepak Pathak. SPIN: Simultaneous perception, interaction and navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18133–18142, 2024.

Chen Wang, Haochen Shi, Weizhuo Wang, Ruohan Zhang, Fei-Fei Li, and Karen Liu. DexCap: Scalable and portable mocap data collection system for dexterous manipulation. In *Proceedings of Robotics: Science and Systems (RSS)*, 2024. doi: 10.15607/RSS.2024.XX.043.

Yuran Wang, Yingping Liang, Yutao Hu, and Ying Fu. RobuSTereo: Robust zero-shot stereo matching under adverse weather. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 25134–25144, 2025a.

Ziye Wang, Li Kang, Yiran Qin, Jiahua Ma, Zhanglin Peng, Lei Bai, and Ruimao Zhang. GauDP: Reinventing multi-agent collaboration through gaussian-image synergy in diffusion policies. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025b.

Philippe Weinzaepfel, Thomas Lucas, Vincent Leroy, Yohann Cabon, Vaibhav Arora, Romain Brégier, Gabriela Csurka, Leonid Antsfeld, Boris Chidlovskii, and Jérôme Revaud. CroCo v2: Improved cross-view completion pre-training for stereo matching and optical flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17969–17980, 2023.

Bowen Wen, Matthew Trepte, Joseph Aribido, Jan Kautz, Orazio Gallo, and Stan Birchfield. FoundationStereo: Zero-shot stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5249–5260, 2025. doi: 10.1109/CVPR52734.2025.00495.

Bowen Wen, Shaurya Dewan, and Stan Birchfield. Fast-FoundationStereo: Real-time zero-shot stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7513–7524, 2026.

Xiaoyang Wu, Yixing Lao, Li Jiang, Xihui Liu, and Hengshuang Zhao. Point Transformer V2: Grouped vector attention and partition-based pooling. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 33330–33342. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/d78ece6613953f46501b958b7bb4582f-Paper-Conference.pdf.

Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point Transformer V3: Simpler, faster, stronger. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4840–4851, 2024. doi: 10.1109/CVPR52733.2024.00463.

Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 3–19, 2018. doi: 10.1007/978-3-030-01261-8_1.

Gangwei Xu, Xianqi Wang, Xiaohuan Ding, and Xin Yang. Iterative geometry encoding volume for stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21919–21928, June 2023.

Gangwei Xu, Xianqi Wang, Zhaoxing Zhang, Junda Cheng, Chunyuan Liao, and Xin Yang. IGEV++: Iterative multi-range geometry encoding volumes for stereo matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(8):7108–7122, 2025. doi: 10.1109/TPAMI.2025.3569218.

Haofei Xu and Juyong Zhang. AANet: Adaptive aggregation network for efficient stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1956–1965, 2020. doi: 10.1109/CVPR42600.2020.00203.

Jingyun Yang, Zi ang Cao, Congyue Deng, Rika Antonova, Shuran Song, and Jeannette Bohg. EquiBot: SIM(3)-equivariant diffusion policy for generalizable and data efficient learning. In *Proceedings of the 8th Conference on Robot Learning (CoRL)*, volume 270, pp. 1048–1068. PMLR, 2024. URL https://proceedings.mlr.press/v270/yang25a.html.

Ruihan Yang, Ge Yang, and Xiaolong Wang. Neural volumetric memory for visual locomotion control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1430–1440, 2023. doi: 10.1109/CVPR52729.2023.00144.

Rujia Yang, Geng Chen, Chuan Wen, and Yang Gao. FP3: A 3D foundation policy for robotic manipulation, 2025. URL https://arxiv.org/abs/2503.08950.

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 *Proceedings of Robotics: Science and Systems (RSS)*, 2024. doi: 10.15607/RSS.2024.XX.067. URL https://www.roboticsproceedings.org/rss20/p067.html.

Feihu Zhang, Xiaojuan Qi, Ruigang Yang, Victor Prisacariu, Benjamin Wah, and Philip Torr. Domain-invariant stereo matching networks. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (eds.), *Computer Vision – ECCV 2020*, pp. 420–439, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58536-5.

Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip H. S. Torr, and Vladlen Koltun. Point Transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16239–16248, 2021. doi: 10.1109/ICCV48922.2021.01595.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In Kostas E. Bekris, Kris Hauser, Sylvia L. Herbert, and Jingjin Yu (eds.), *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, 2023. doi: 10.15607/RSS.2023.XIX.016. URL https://doi.org/10.15607/RSS.2023.XIX.016.

Yuke Zhu, Ziyu Wang, Josh Merel, Andrei Rusu, Tom Erez, Serkan Cabi, Saran Tunyasuvunakool, János Kramár, Raia Hadsell, Nando de Freitas, and Nicolas Heess. Reinforcement and imitation learning for diverse visuomotor skills. In *Proceedings of Robotics: Science and Systems (RSS)*, 2018. doi: 10.15607/RSS.2018.XIV.009. URL https://www.roboticsproceedings.org/rss14/p09.html.

Appendix

A Task and Data Collection Details

Simulation Tasks

- RoboMimic: We exclude *Lift* and *Can* since the success rate has already saturated in the original study, and include three more challenging tasks *ToolHang*, *Square*, and *Transport*.
- RoboCasa-Kitchen: We use 24 kitchen tasks from this benchmark, and the main purpose is to evaluate the pretrained VLA models ($\pi_{0.5}$ and GR00T-N1.5).
- OmniGibson is a high-fidelity physics simulation environment that models a large variety of object interactions in realistic home-scale scenes, enabling long-horizon mobile manipulation tasks grounded in everyday human activities. We design four tasks and collect demonstrations using cuRobo (Sundaralingam et al., 2023).

Real Robot Tasks. We design five real-world tabletop manipulation tasks and collect 200 demonstrations per task. Example demonstration trajectories are shown in Figure 11.

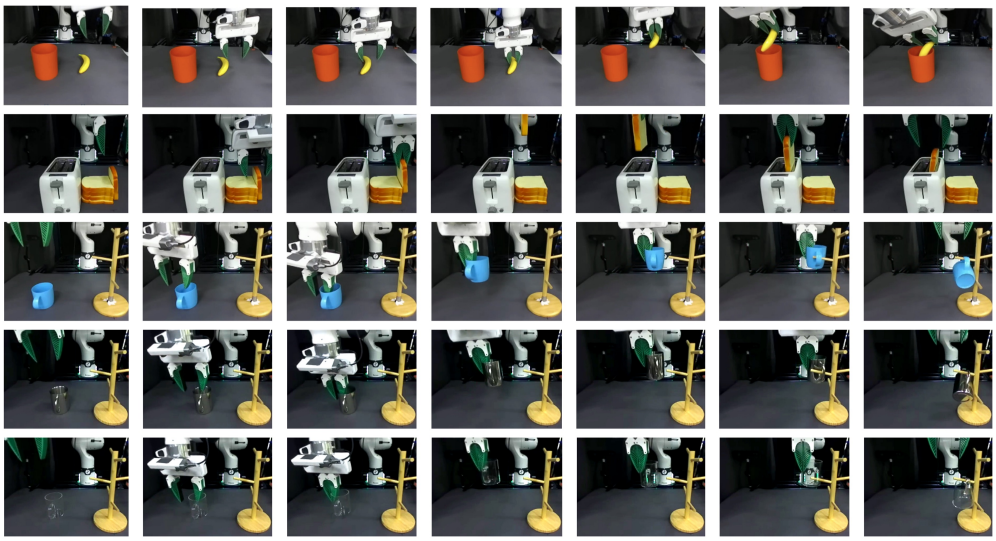


Figure 11: Example trajectories of real-world tabletop tasks.

- Banana PnP: the robot picks up a banana and inserts it into a container.
- Toast Insert: the robot picks up a slice of toast and inserts it into the toaster.
- Plastic Cup Hang: the robot picks up a plastic mug and hangs it on the holder.
- Steel Cup Hang: the robot picks up a steel mug and hangs it on the holder.
- Glass Cup Hang: the robot picks up a glass mug and hangs it on the holder.

For the following bimanual mobile manipulation tasks, we collect 75 demonstrations per task.

- Turn on Radio: The R1Pro robot approaches the shelf and picks up the radio, then uses its left gripper to press the radio's button, and puts the radio back on the shelf.
- Toast PnP: The R1Pro robot approaches the table, picks up the toast from the plate, hands it over to the left gripper and places it on the shelf.

A.1 Real-Robot Hardware Setup

To further validate the real-world applicability, we deploy the stereo policy on a 7-DoF Franka Emika robot arm. For visual perception, we utilize the dual camera setup: a ZED Mini provides an external view, and a wrist-mounted ZED Mini captures a close-range view. For the bimanual R1 Pro setup, each wrist has its own stereo camera.

B Stereo Hardware Configurations

B.1 Stereo Baselines

In simulation, the default stereo baselines are 6 cm for external cameras and 2 cm for wrist cameras. The real-world ZED Mini cameras have a 6.3 cm baseline.

B.2 Test-Time Baseline Transfer

The baseline–distance sweep in the main paper compares camera configurations. Here, we hold the trained policy fixed: training uses a 6 cm baseline, and evaluation uses 4, 6, or 8 cm at a fixed camera–object distance. Success decreases from 94% at the training baseline to 86% at 4 cm and 88% at 8 cm (Table 4). The 8- and 6-point decreases show tolerance to moderate baseline changes without retraining, with the best performance at the training configuration.

Test baseline	4 cm	6 cm	8 cm
Success rate (%)	$86_{1.6}$	$94_{1.9}$	$88_{1.6}$

Table 4: ToolHang SR after training at a 6 cm baseline and changing only the baseline at test time. Subscripts are standard deviations over three seeds.

B.3 Stereo Views across Different Baselines and Distances

In §4 ($\mathcal{Q}\mathbf{3}$), we examine how stereo baseline and camera-object distance affect StereoPolicy. In Figure 12, we visualize all stereo camera views across baselines ranging from 0.02 m to 0.10 m and object distances ranging from 0.59 m to 0.99 m that are used in the experiments.

B.4 Camera Angle Mappings

As reported in §4 ($\mathcal{Q}\mathbf{3}$), we conduct experiments on the ToolHang task across 10 camera views. Figure 13 shows the 10 views, which are *frontview, frontview-high, agentview-left, agentview-right, agentview-righthigh, agentview-lefthigh, sideview-left, sideview-right, sideview-lefthigh, sideview-righthigh* from left to right.

C Implementation Details

C.1 StereoPolicy-DP Training

Each image is encoded by a ResNet18 with an FPN, with weights shared between the left and right images. For external cameras, frozen DINOv2 features are downsampled by a 4×4 convolution with stride 4, concatenated with the CNN features, and projected to the token dimension by an MLP. The Stereo Transformer has 2 layers and 8 attention heads. As described in §2.1, each layer applies self-attention within each image, cross-attention from two images, and an MLP block, with 2D RoPE applied to queries and keys. The output tokens are projected by an MLP to a 128-dimensional view feature. Wrist cameras use the same encoder without DINOv2.

For all diffusion policies, we use a UNet (Ronneberger et al., 2015) as the policy backbone and replace the BatchNorm layers in the UNet with GroupNorm (Wu & He, 2018) for stable training (Chi et al., 2024). We train the diffusion policy backbones from scratch. For RGB policies, we use images

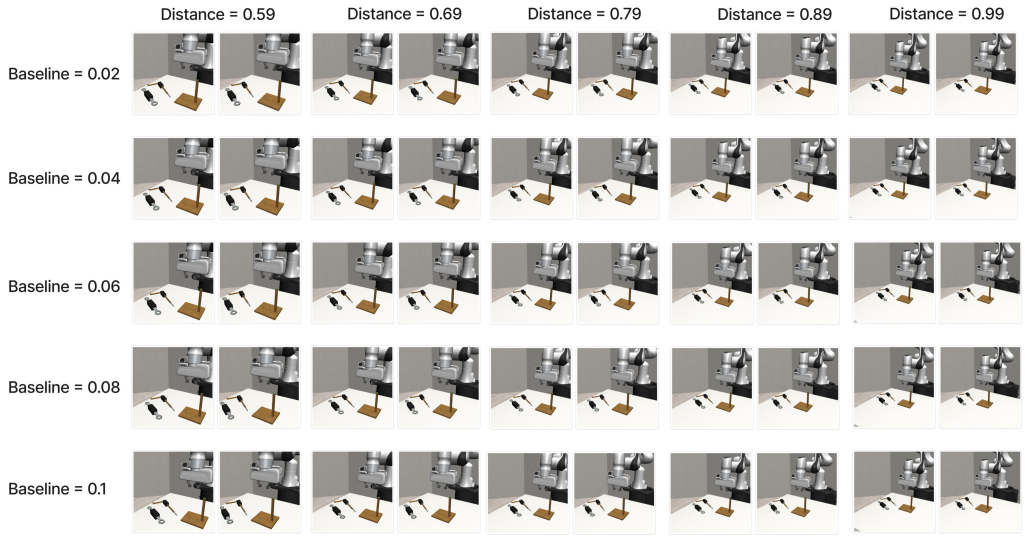


Figure 12: Stereo camera views across different baselines and distances. Baseline indicates the distance between stereo cameras; distance indicates the camera-object distance.

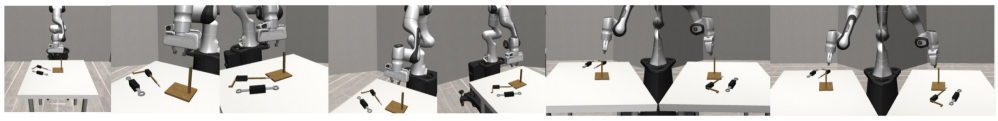


Figure 13: Visualization of the camera views. Experimental results are shown in Figure 8b.

from the left cameras as input. All models are trained with a batch size of 64 and a learning rate of 1e-4. The observation horizon is set to 2 timesteps, while the action prediction horizon is set to 16 steps. For OmniGibson tasks, we train for 1,000 epochs, while Robomimic tasks are trained for 500 epochs following common practice in prior work.

C.2 StereoPolicy-VLA Training

We fine-tune the pretrained GR00T-N1.5 checkpoint on RoboCasa tasks for 60K gradient steps, maintaining the original training protocol as closely as possible to ensure comparability. Learning rate is set to 1e-4. Across all settings, optimization hyperparameters follow the corresponding baseline configurations to isolate the effect of our stereo architecture.

For $\pi 0.5$, we use the pretrained checkpoints to reproduce fine-tuned performance on RoboCasa-Kitchen. We train it for 80K steps, both with a global batch size of 128. We set the learning rate to 2.5e-5 with cosine decay to 2.5e-6 and 1K warmup steps. At inference, we use an action horizon $H = 16$ and execute all actions without replanning. Both training uses 8 H100 GPUs.

C.3 Baseline Implementations

- **Monocular RGB**. Following Robomimic (Mandlekar et al., 2021) and Diffusion Policy (Chi et al., 2024), at each timestep t, the sensor inputs include monocular RGB images from each view i, $I_{t,i} \in \mathbb{R}^{H \times W \times 3}$. For a fair comparison, we augment the RGB policy with the same vision encoder and frozen external-view DINOv2 features as StereoPolicy-DP plus a self-attention Transformer, whose capacity is adjusted to approximately match our trainable parameter count. This comparison tests whether increased model capacity accounts for the improvement.

Benchmark	ROBOMIMIC		OMNIGIBSON	
Task	TOOLHANG-100	SQUARE-100	POURWATER-200	Avg. SR
STEREOPOLICY-DP	$94_{1.9}$	$88_{0.9}$	$70_{2.5}$	**84.0**
with (L,L) input	$62_{1.6}$	$76_{1.6}$	$50_{3.3}$	62.7
w/o StereoTransformer	$82_{2.9}$	$76_{2.8}$	$52_{1.6}$	70.0
w/o FPN	$78_{1.6}$	$80_{1.6}$	$64_{1.6}$	74.0
w/o DINO	$84_{1.6}$	$82_{2.8}$	$62_{2.8}$	76.0
with DINO on both view	$88_{1.6}$	$84_{0.0}$	$66_{3.3}$	79.3

Table 5: **Component Ablation of STEREOPOLICY-DP**. ToolHang and Square use 100 demonstrations; PourWater uses 200. Subscripts are standard deviations over three seeds.

Encoder	SR (%)
ResNet18	$64_{2.8}$
OpenCLIP-B/16	$78_{2.8}$
SigLIP-SO400M/14	$80_{4.3}$
EdgeNeXt-S	$82_{1.6}$
ResNet18+FPN	$84_{1.6}$

Table 6: Vision-encoder comparison (w/o DINO) on ToolHang with 100 demonstrations.

- **Multi-view RGB** Stereo images from each view are passed to the same vision encoders as STEREOPOLICY-DP respectively, and the resulting image features are concatenated before the diffusion policy. The architecture and parameters are matching STEREOPOLICY-DP. To match the architecture and parameters of STEREOPOLICY-DP, we retain the simple multi-view fusion strategy: features from the two images are concatenated without token-level left–right attention, followed by an enlarged projection/MLP whose capacity is adjusted to match StereoPolicy.
- **RGB-D**. Each view provides an RGB image and an aligned depth map, enabling explicit spatial structure for downstream processing. Depth maps $\in \mathbb{R}^{H\times W\times 1}$ are repeated to 3 channels and processed by a separate ResNet18 encoder. The resulting RGB and depth features are concatenated.
- **RGBD-3DDA**. We follow the same setup as in the original paper (Ke et al., 2024), use ResNet18 and FPN to extract multi-scale RGB features and reconstruct point-clouds from depth.
- **PCD-PointNet**. We reconstruct and fuse multi-view point clouds from RGB-D observations, followed by downsampling them to 4096 points using farthest point sampling (FPS) (Qi et al., 2017a).
- **PCD-DP3**. For point-cloud inputs, we reconstruct and fuse multi-view point clouds from RGB-D observations, downsample them to 4096 points using farthest point sampling (FPS) (Qi et al., 2017a), and encode them using the DP3 encoder (Ze et al., 2024).
- **Frozen FoundationStereo features (StereoVLA-style).** Following StereoVLA (Deng et al., 2025), we replace our trainable vision encoder and StereoTransformer with StereoVLA's encoder using frozen FoundationStereo (FS) features under the same backbone
- **Stereo-estimated depth.** FoundationStereo or Fast-FoundationStereo (Wen et al., 2026) estimates depth from the same rectified stereo pair, and this depth replaces the simulator depth in the RGB-D and RGBD-3DDA baselines above. Everything else in those baselines is unchanged.

C.4 EVALUATION

For simulation tasks, we perform 100 rollouts at every 50 training epochs for Robomimic tasks, and every 250 training epochs for OmniGibson tasks. The highest success rate achieved across training is reported with three different seeds. For real-robot evaluation, we report the average success rate over 20 trials, with randomized initial poses.

D ADDITIONAL ABLATIONS

D.1 VISION ENCODER

To select vision encoders that mostly utilize stereo images, table 6 ablates image encoder variants without DINOv2 on ToolHang with 100 demonstrations. ResNet18+FPN reaches the highest success rate (84%). These results support the benefit of extracting multi-scale feature from stereo image pairs, and ResNet18+FPN acts as a competitive encoder in this setting.

D.2 ATTENTION DESIGN

We hold the rest of the model architecture, wrist observations, training data, and optimization budget fixed, and change only how the stereo images are fused. Both branches receive identical augmentation

for duplicated (L, L) input. Joint attention concatenates the left and right tokens into a single sequence and applies standard self-attention. The Stereo Transformer alternates within-image and cross-image attention. The attention variants share tokenization, feature width, and positional conventions. Results are mean success rates across three training seeds.

Fusion	Views	ToolHang-100	Square-100	PourWater-200
Joint attention	L, L	$64_{1.6}$	$78_{1.6}$	$42_{2.8}$
Joint attention	L, R	$82_{1.6}$	$82_{1.6}$	$60_{0.0}$
Stereo Transformer	L, L	$62_{1.6}$	$76_{1.6}$	$50_{3.3}$
Stereo Transformer	L, R	$94_{1.9}$	$88_{0.9}$	$70_{2.5}$

Table 7: Matched fusion success rates (%). Task suffixes give demonstration counts. The Stereo Transformer rows correspond to StereoPolicy-DP and its (L, L) variant in Table 5. Subscripts are standard deviations over three seeds.

When both branches see the same image, the fusion designs behave almost identically on ToolHang and Square. PourWater is the exception. Even with duplicated input, the Stereo Transformer reaches 50%, compared with 42% for joint attention , so part of its advantage on this task comes from the architecture itself rather than from stereo. The picture changes once the true right image is provided. The Stereo Transformer rises from 62% to 94% on ToolHang and from 76% to 88% on Square, whereas joint attention reaches only 82% on both tasks . On PourWater, joint attention benefits from the right image about as much as the Stereo Transformer does, and the Stereo Transformer keeps its lead at 70%. Overall, the Stereo Transformer makes better use of the second view than the other fusion designs, and the difference is clearest on ToolHang and Square.

D.3 Pre-trained Stereo Foundation Models on Real Robot

Real robot with FoundationStereo On real-world BananaPnP, FoundationStereo+RGBD succeeds in 1/20 trials, compared with 16/20 for StereoPolicy. The depth-based deployment exhibited significant stale observations and delayed actions. This comparison highlights the importance of the complete perception–control pipeline: its low success reflects the combined effects of depth estimation and execution timing, rather than isolating either factor.

D.4 Geometric Information

We freeze the visual representation before proprioceptive input and fit equal-capacity linear probes to camera-frame gripper–target relative depth. Feature dimensionality and probe training data are fixed. Evaluation holds out complete trajectories and spatial layouts.

Representation	ToolHang MAE (cm)	PourWater MAE (cm)
Joint attention, L, L	2.8	3.6
Joint attention, L, R	1.9	2.5
Stereo Transformer, L, L	2.7	3.5
Stereo Transformer, L, R, before training	3.1	4.0
Stereo Transformer, L, R, after training	1.1	1.5

Table 8: Relative-depth mean absolute error from frozen features.

After action training, the stereo representation reduces depth error from 3.1 to 1.1 cm on ToolHang and from 4.0 to 1.5 cm on PourWater. At fixed stereo input, its errors are 42% and 40% lower than joint attention. The duplicated-input control does not reproduce these gains. These probes identify accessible geometric information.

D.5 Discussion on Point Clouds Policy Performance

Our point-clouds preprocessing follows most standard preprocessing steps. Point-cloud baselines use the same camera views as the other modalities. Calibrated RGB-D observations are fused in a

Backbone	Input	# Demos 30	100	300
$\pi_{0.5}$	Monocular	$48.71_{0.7}$	$67.64_{1.0}$	$70.31_{0.6}$
$\pi_{0.5}$	Two-image	$49.01_{0.1}$	$68.24_{0.7}$	$70.81_{1.8}$
$\pi_{0.5}$	StereoPolicy-VLA	$\mathbf{51.72}_{0.5}$	$\mathbf{71.54}_{1.0}$	$\mathbf{74.40}_{0.5}$
GR00T-N1.5	Monocular	$44.98_{0.4}$	$63.58_{0.6}$	$64.30_{1.1}$
GR00T-N1.5	Two-image	$45.67_{1.6}$	$65.24_{0.7}$	$65.31_{0.7}$
GR00T-N1.5	StereoPolicy-VLA	$\mathbf{47.12}_{0.9}$	$\mathbf{66.17}_{1.2}$	$\mathbf{67.50}_{1.0}$

Table 9: Matched VLA macro-average success (%) on 24 RoboCasa tasks. Two-image: the backbone's native multi-view input with the left and right images. Subscripts are standard deviations over three seeds.

Method	Latency (ms) p50 / p95
RGB-DP	632 / 674
Joint attention	695 / 731
StereoPolicy	698 / 736
FS + 3DDA	1008 / 1060
FastFS + 3DDA	777 / 819

Table 10: End-to-end inference latency (median / 95th percentile) on one H100 GPU.

common frame, cropped to remove the background and retain the workspace, and downsampled by farthest-point sampling to 4096 points. Increasing the count to 10K did not improve performance in the evaluated setting and increased inference cost. This observation does not support point density as the main cause of the performance gap.

Especially, one bottleneck of low-performance of PCD is instead real-world PCD quality: Figure 1 and Figure 7 show missing or deformed task-relevant geometry, which is often happening in point clouds reconstruction. Second, Our real task ToastInsert is especially challenging as successful insertion requires precise 3D alignment with a narrow slot, which is depth-sensitive, the bad-quality point clouds will significantly affect the policy performance.

D.6 StereoPolicy-VLA Performance

We compare monocular input, native two-image input, and StereoPolicy fusion using the same pretrained checkpoints and fine-tuning budgets. Evaluation averages tasks equally across the 24 RoboCasa tasks, using three seeds and 100 test episodes per task and seed.

Native two-image input improves both backbones over monocular input. StereoPolicy further provides improvements over native two-image. These gains identify a benefit from the fusion design beyond supplying a second image.

D.7 Overhead, computational cost

StereoPolicy uses 112M parameters, of which 10M belong to the Stereo Transformer, and the reported compute is 128 GFLOPs, peak memory 1.4G. Inference is profiled on one H100 with batch size one, observation horizon two, prediction horizon sixteen, action dimension seven, and 100 denoising steps. Timing excludes 100 warm-up calls and summarizes 1,000 synchronized calls. Stage medians and end-to-end median/95th-percentile latency use the same profiling scope.

Stereo fusion adds 3 ms to the joint-attention control. The full pipeline is 310 ms faster than FoundationStereo with 3DDA and 79 ms faster than FastFoundationStereo with 3DDA.

D.8 Removing Wrist Observations

Removing the wrist camera reduces StereoPolicy success from 96% to 94%, compared with 90% to 86% for RGB, 88% to 84% for RGB-D, and 76% to 62% for DP3 (Table 11). StereoPolicy has a smaller decrease than these baselines and retains the highest absolute success rate, supporting the value of external stereo observations when wrist sensing is unavailable.

Setting	**RGB**	**RGB-D**	**PCD** PointNet	PointNet++	DP3	**RGB-D** 3DDA	**Multi-View**	**StereoPolicy-DP**
Default (w/ wrist camera)	$90_{1.6}$	$88_{2.5}$	$32_{2.8}$	$28_{4.9}$	$76_{0.9}$	$92_{2.8}$	$92_{1.9}$	$\mathbf{96}_{2.8}$
w/o wrist camera	$86_{1.6}$	$84_{1.6}$	$30_{4.9}$	$28_{3.3}$	$62_{2.8}$	$88_{1.6}$	$88_{1.6}$	$\mathbf{94}_{3.3}$

Table 11: Effect of removing wrist-camera observations on ToolHang with 200 demos. Success rate (%); subscripts are standard deviations over three seeds.

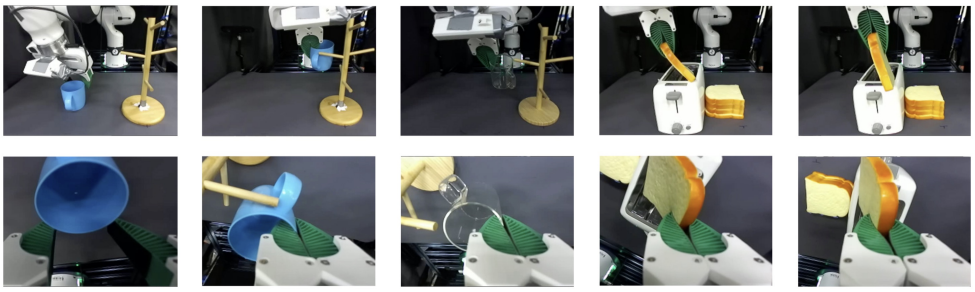


Figure 14: Failure cases of baseline visual modalities on real-world tabletop tasks. RGB, RGB-D, and point-cloud inputs can be inaccurate for fine-grained manipulation: gripper-target alignment can drift, reflective objects can obscure object boundaries, and transparent objects often produce missing or noisy depth and point-cloud geometry.

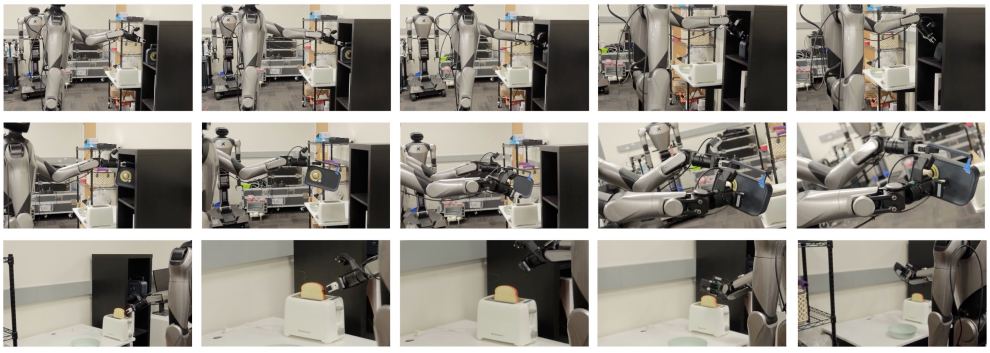


Figure 15: Monocular RGB failures in bimanual mobile manipulation. In the TURN ON RADIO task, RGB policies often fail at precise handle insertion and button pressing due to insufficient depth cues. In the TOAST PNP task, failures commonly occur during grasp alignment and placement due to ambiguous monocular depth cues.

E QUALITATIVE FAILURE CASES

Baseline modality failures in tabletop manipulation. Figure 14 shows representative tabletop failures caused by unreliable object geometry and spatial alignment. In Cup Hang tasks, weak object-boundary perception can lead to inaccurate cup localization and failed grasping. Noisy depth observations further cause the cup handle to misalign with the rack peg during hanging. For Glass Cup Hang tasks, specular reflection and transparency can make the cup partially or completely disappear in depth videos and reconstructed point clouds while it is lifted in free space. In Toast Insert tasks, insufficient 3D spatial information or inaccurate depth estimates can cause vertical or lateral misalignment between the toast and the toaster slot.

Remaining difficulty on transparent objects. StereoPolicy succeeds in only 3/20 Glass Cup Hang trials (Table 1), so transparent-object manipulation remains difficult despite its improvement over the tested baselines. Refraction and weak visual boundaries, especially around the thin glass rim and handle, can make localization and left/right correspondence ambiguous even in RGB images. Avoiding unreliable depth reconstruction therefore does not eliminate the visual ambiguity that can hinder grasping and handle–peg alignment.

Monocular RGB failures in mobile manipulation. Figure 15 visualizes representative failure cases of RGB-based VLA policies on mobile manipulation tasks. In the TURN ON RADIO task, the RGB policy often approaches the radio but fails to insert the gripper into the handle or press the button accurately. In the TOAST PNP task, RGB policies can localize the toast but often fail during grasp alignment or final placement, where precise spatial alignment is required. In contrast, StereoPolicy uses stereo cues to improve gripper–target alignment.