Enhancing 2D VLA Models with 3D Spatial Awareness

Kailash Jagadeesh
Carnegie Mellon University
kailashj@andrew.cmu.edu

Keerthi Gaddobanahalli Vijayakumar
Carnegie Mellon University
kgaddoba@andrew.cmu.edu

***Abstract*— Current Vision-Language-Action (VLA) models primarily rely on 2D visual inputs, which fundamentally limits their ability to interpret depth, object geometry, and occlusions essential for precise robotic manipulation. While native 3D-aware models exist, they often demand prohibitive computational resources and massive 3D datasets. This work explores a lightweight methodology to enhance existing 2D VLA architectures with explicit 3D spatial awareness without requiring specialized hardware. We investigate two "software-only" pipelines: (1) generating pseudo-3D point clouds from single-view RGB images using Depth Anything v3 to drive a 3D Diffusion Policy, and (2) injecting semantic 2D features from a DINOv2 backbone. Our experiments demonstrate that inferred 3D cues can successfully guide manipulation policies, achieving a 100% success rate on planar tasks (Slide Plate) and 85% on dynamic tasks (Basketball) at lower resolutions. However, we observe a significant quality gap between real and predicted depth, where artifacts in high-resolution inputs (512×512) cause performance to degrade in geometrically complex scenarios. These results highlight the potential of inferred 3D representations for scaling robot learning while identifying critical dependencies on the fidelity of monocular depth estimation.**

I. INTRODUCTION

The landscape of robot learning has been fundamentally reshaped by the emergence of Vision-Language-Action (VLA) models. By leveraging massive internet-scale datasets and large language models, recent architectures such as OpenVLA [1], GR00T [2], and π_0 [3] have demonstrated a remarkable ability to unify perception, reasoning, and control within a single framework. These models excel at interpreting natural language instructions and generalizing to unseen environments, largely due to the abundance of 2D image and video data available for pretraining. However, this reliance on 2D visual inputs introduces a critical bottleneck: the world is inherently three-dimensional.

Current VLA models predominantly operate on flat 2D projections, forcing them to implicitly infer spatial relationships. This results in significant limitations when interpreting depth, resolving object geometry, or handling occlusions—capabilities that are fundamental for precise robotic manipulation. Conversely, "3D-native" policies, such as PerAct [4] and 3D Diffusion Policy (DP3) [8], explicitly process point clouds or voxel grids to achieve superior precision in geometric tasks. Yet, these approaches face a scalability barrier: high-quality 3D data is scarce, expensive to collect, and computationally intensive to process, rendering the training of general-purpose "3D foundation models" currently impractical.

This work aims to bridge the gap between the scalability of 2D VLAs and the precision of 3D policies. We propose a methodology to enhance 2D models with "software-defined" spatial awareness, eliminating the need for specialized depth hardware or massive 3D datasets. We investigate two primary mechanisms for this integration: (1) generating pseudo-3D point clouds from single-view RGB images using monocular depth estimators like *Depth Anything v3* [9], and (2) injecting spatially-aware semantic features from *DINOv2* [10] backbones directly into the policy. By systematically evaluating these "inferred 3D" representations against ground-truth 3D policies, we seek to answer a pivotal question: Can lightweight, software-predicted 3D cues effectively replace physical depth sensors to enable scalable, geometrically-aware robot learning?

II. MOTIVATION

Vision-Language-Action (VLA) models have recently gained significant attention for their ability to unify perception, reasoning, and control within a single learning framework. Traditional

robotic systems relied on modular pipelines that separated perception, planning, and actuation, each demanding extensive manual engineering and task-specific tuning. In contrast, VLAs leverage large-scale pretraining on multimodal data such as images, text, and demonstrations to learn generalizable mappings from language-conditioned visual inputs to robot actions. This paradigm shift has enabled robots to execute diverse tasks from natural-language prompts, adapt to new environments, and reuse visual and semantic representations learned from web-scale data. As a result, VLAs are emerging as a promising foundation for scalable, instruction-driven robotic intelligence.

Recent VLA architectures such as Pi_0, Groot, and OpenVLA have demonstrated impressive capabilities in linking natural language instructions to robotic actions. However, these models primarily rely on two-dimensional visual representations of scenes, which limits their spatial understanding and geometric reasoning—both essential for precise manipulation and planning in real-world settings.

Humans naturally perceive the world in three dimensions, allowing us to reason about depth, occlusion, and object relationships. In contrast, VLAs trained exclusively on two-dimensional projections must infer 3D relationships implicitly. This often leads to errors in tasks that require accurate spatial awareness, such as grasping occluded objects or estimating relative positions. While there are ongoing efforts to develop full 3D VLA models, the vast availability of 2D visual data across varied conditions, captured through videos, images, and simulators, makes it more practical to enhance existing 2D VLAs by incorporating 3D cues rather than replacing their visual encoders entirely.

This project aims to explore whether integrating explicit 3D scene information into pretrained 2D VLA architectures can improve their spatial reasoning capabilities. Given the limited computational resources and the scarcity of large-scale 3D datasets for general-purpose training, our goal is to inject 3D scene representations into the VLA pipeline in a lightweight and complementary manner. Specifically, we aim to incorporate geometric information without discarding the 2D vision tokens, thereby enriching the model's spatial understanding while retaining its pretrained 2D perceptual strengths.

III. LITERATURE REVIEW

2D Vision-Language-Action Models. Most existing VLAs operate purely on two-dimensional visual inputs. They are trained on large-scale multimodal datasets consisting of image–text–action triplets collected from human demonstrations or simulated rollouts. OpenVLA [1] demonstrated that scaling both data and model size can produce strong zero-shot generalization across robotic platforms, learning directly from diverse manipulation episodes paired with natural language instructions. NVIDIA's GR00T-N1 [2] extended this idea to humanoid control by pairing a multimodal transformer for perception and reasoning with a diffusion-based motion generator, allowing the robot to execute semantically grounded motor commands from textual prompts. Similarly, the π_0 and $\pi_{0.5}$ families [3] leverage pretrained vision–language encoders such as CLIP or SigLIP and co-train them on large collections of robot demonstrations. These 2D VLAs have shown remarkable versatility in mapping high-level instructions to low-level actions. However, because they rely solely on RGB images, their spatial understanding is fundamentally limited, making it difficult to reason about occlusions, depth, or the geometric relationships between objects in cluttered scenes.

3D-Aware Policies for Manipulation. A complementary research direction focuses on explicitly encoding 3D structure for policy learning. PerAct [4] introduced a transformer-based architecture that processes voxelized RGB-D observations to predict discrete 6-DoF actions, demonstrating that grounding control in 3D voxel space improves precision for fine-grained tasks. 3D Diffusion Policy (DP3) [5] and its successors extend this idea by conditioning action diffusion models directly on point clouds, achieving strong spatial reasoning and robustness to camera viewpoint changes. ManiCM [6] further improves geometric consistency by learning 3D scene flow alongside policy updates, showing that spatially aware features lead to more stable manipulation behavior. Although these models outperform 2D VLAs on spatially demanding

tasks, their reliance on large RGB-D datasets and heavy compute requirements make them difficult to scale to the same level as language-grounded VLA frameworks.

Injecting 3D Information into Pretrained 2D VLAs. A more recent line of work aims to bridge the gap between scalable 2D VLAs and geometry-aware 3D policies by incorporating 3D features into existing pretrained models without retraining them from scratch. PointVLA [7] introduces a lightweight point cloud encoder that extracts geometric features and fuses them with 2D vision tokens through cross-attention. This hybrid design enriches the model's spatial understanding while preserving the generalization capabilities and large-scale priors of the 2D backbone. Other concurrent efforts explore similar multi-view or depth-aware token fusion strategies to enhance object localization and affordance reasoning. These approaches motivate our work, which similarly investigates how explicit 3D scene representations can be integrated into pretrained 2D VLA architectures to improve spatial reasoning under realistic compute and data constraints.

IV. METHODOLOGY

To evaluate the feasibility of replacing hardware sensors with software-inferred geometry, we designed a comparative study using the 3D Diffusion Policy (DP3) as our primary baseline. Our methodology investigates two distinct strategies: (1) generating explicit "pseudo-3D" representations via monocular depth estimation, and (2) leveraging implicit 3D-aware features through large-scale pretrained vision transformers.

A. Simulation Environment and Tasks

All experiments were conducted within the **MetaWorld** benchmark [11], a multi-task robotics simulation platform based on the MuJoCo physics engine. MetaWorld provides a diverse suite of 50 manipulation tasks, ranging from simple object interaction (e.g., button pressing) to complex dexterous manipulation (e.g., bin picking and assembly).

Training a single generalist policy across all 50 tasks is computationally prohibitive and difficult to stabilize. Therefore, to ensure a tractable yet rigorous evaluation, we selected a representative subset of **8 tasks** that demand varying degrees of spatial precision, including *Box Close, Button Press, Door Lock*, and *Bin Picking*.

B. Baseline: Vanilla 3D Diffusion Policy

Our baseline architecture is the standard implementation of the 3D Diffusion Policy (DP3) [8]. The vanilla pipeline relies on a stereo camera setup to capture ground-truth depth information.

1) **Input Processing:** The system captures RGB-D data, which is converted into a 3D point cloud. Crucially, this baseline relies on ground-truth semantic segmentation to isolate the object of interest from the background.
2) **Encoding:** The segmented point cloud is downsampled to a low resolution of $N = 512$ points. These points are processed by a **PointNet** encoder, which aggregates the geometric data into a 64-dimensional latent vector.
3) **Policy:** This geometric embedding, concatenated with the robot's proprioceptive observations, serves as the conditioning input for the diffusion-based action head.

Figure 1 illustrates this standard architecture.

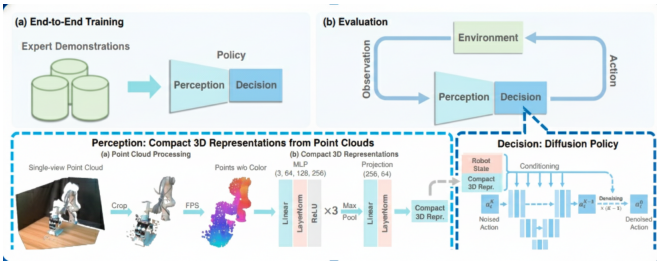


Fig. 1: The Vanilla DP3 Architecture. It utilizes stereo cameras and ground-truth segmentation to feed a 512-point point cloud into a PointNet encoder.

C. Pipeline 1: Pseudo-Depth with Depth Anything v3

The first proposed pipeline aims to remove the dependency on stereo cameras and ground-truth segmentation while maintaining an explicit 3D representation. We replace the hardware depth sensor with a "software sensor" pipeline utilizing **Depth Anything v3** [9].

In this approach (Figure 2):

- We capture only single-view RGB images from the environment.
- The *Depth Anything v3* model infers a high-fidelity dense depth map from the RGB input.
- This depth map is unprojected to create an "augmented" 3D point cloud. Unlike the vanilla baseline, this method does not require semantic segmentation masks, as the depth estimator provides global scene context.

A key advantage of this generative approach is the flexibility of resolution. Since the point clouds are computationally generated, we are not bound by sensor sparsity. We evaluate this pipeline at both the baseline resolution ($N = 512$) and a high-resolution setting ($N = 4096$) to investigate whether denser geometric cues improve manipulation precision.

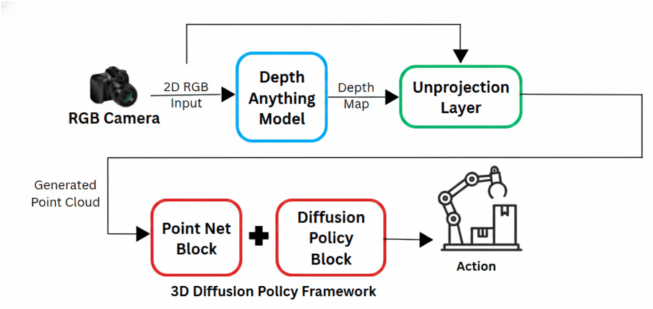


Fig. 2: Pipeline 1: Pseudo-3D Generation. We extract depth maps using Depth Anything v3 and reproject them into augmented point clouds (up to 4096 points), bypassing the need for stereo cameras.

D. Pipeline 2: Implicit 3D with DINOv2

The second pipeline challenges the necessity of calculating explicit geometric representations (such as point clouds) entirely. Instead, we hypothesize that modern Vision Transformers (ViTs) possess sufficient internal understanding of geometry and depth to guide manipulation tasks directly from 2D data.

In this configuration (Figure 3), we replace the explicit PointNet encoder with a **DINOv2** backbone [10].

- The model receives 2D RGB inputs directly.
- The DINOv2 encoder, pretrained on massive datasets using self-supervision, encodes both semantic and implicit geometric information into the latent dimension vector.
- This vector allows the policy to "infer" depth and spatial relationships without explicit coordinate inputs.

While newer architectures such as VGGT or DINOv3 theoretically offer superior 3D-aware features, we utilized DINOv2 (specifically the ViT-S/14 variant) due to compatibility constraints with the existing codebase. This pipeline represents a move toward fully 2D-native policies that retain 3D reasoning capabilities.

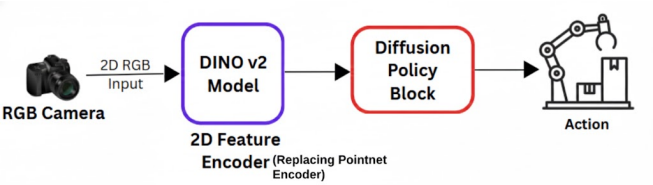


Fig. 3: Pipeline 2: Implicit Encoding. We replace the geometric PointNet encoder with a pretrained DINOv2 Vision Transformer, allowing the model to infer spatial information directly from 2D tokens.

V. Results

We evaluated our proposed pipelines against the vanilla DP3 baseline across a subset of MetaWorld tasks. Our analysis focuses on success rates, training efficiency, and the qualitative fidelity of the inferred 3D representations.

A. Baseline Performance

The vanilla 3D Diffusion Policy, utilizing ground-truth stereo depth and segmentation, achieved a **100% success rate** across all tested tasks at low resolution (128×128). When scaled to high resolution (512×512), the baseline also achieved **100% accuracy**. Notably, while the high-resolution baseline required more computational power per step, it converged to optimal performance with significantly fewer training iterations compared to the low-resolution variant.

B. Pipeline 1: Pseudo-Depth with Depth Anything v3

The performance of the "software-only" depth pipeline was highly task-dependent, revealing a

critical limitation in temporal consistency. Quantitative results are presented in Table I.

Planar Manipulation Success: For tasks where the manipulator and object remain primarily on the table surface, such as *Plate Slide Side*, the pipeline performed exceptionally well. We achieved a **100.0% success rate** at high resolution and **90.0%** at low resolution. In these planar scenarios, the relative scale of objects is preserved effectively by the depth estimator. As shown in Figure 4, the high-resolution pseudo-depth maps and generated point clouds (b) closely resemble the ground truth (a). Furthermore, Figure **??** demonstrates that the depth estimation remains consistent across different timeframes for these tasks, allowing the policy to track the object effectively.

Non-Planar Failure Modes: In contrast, tasks requiring the manipulator to leave the table surface or perform complex 3D reorientations (e.g., *Assembly*, *Pick-Place*, *Basketball*) resulted in a **0.0% success rate**. Qualitative analysis revealed that *Depth Anything v3* struggles to maintain temporal consistency when the camera perspective or object occlusion changes significantly. As illustrated in Figure 6, while the ground truth (a) captures the geometry clearly, the pseudo-depth (b) fluctuates inconsistently across frames. Without the aid of explicit segmentation, this noise propagates into the point cloud, causing the policy to lose track of the end-effector's spatial relationship to the target.

Despite these failures, the training throughput was efficient. Because *Depth Anything* runs inference rapidly, the training time for this pipeline was comparable to the low-resolution baseline, even when processing high-resolution inputs.

TABLE I: Depth Anything v3 Pipeline Results (Pseudo-3D)

Task	Res	Size	Points	Success (%)
Plate Slide Side	High	512×512	4096	**100.0**
Plate Slide Side	Low	128×128	512	90.0
Assembly	High	512×512	4096	0.0
Assembly	Low	128×128	512	0.0
Pick-Place Wall	High	512×512	4096	0.0
Basketball	Low	128×128	512	0.0

C. Pipeline 2: Implicit 3D with DINOv2

The DINOv2 feature injection pipeline demonstrated strong potential for implicit geometric reasoning, particularly at lower resolutions. Results are detailed in Table II.

Sample Efficiency vs. Compute: A major finding was the sample efficiency of the Vision Transformer backbone. Pipeline 2 was able to match the accuracy of the vanilla baseline on successful tasks (e.g., *Hammer*, *Basketball*) with approximately **half the training iterations**. However, the wall-clock training time was higher than the baseline due to the computational weight of the ViT architecture and the large size of the extracted feature vectors (384-dim).

Resolution Sensitivity: Interestingly, this pipeline performed significantly better at lower resolutions. The model achieved an **85.0% success rate on Hammer** and **80.0% on Basketball** at 128×128 resolution. However, scaling to high resolution (512×512) caused performance to drop (e.g., Basketball dropped to 50.0%, Hammer to 0.0%). This suggests that the DINOv2 features used (ViT-S/14) may optimize for semantic understanding over the fine-grained high-frequency spatial details required for high-res manipulation.

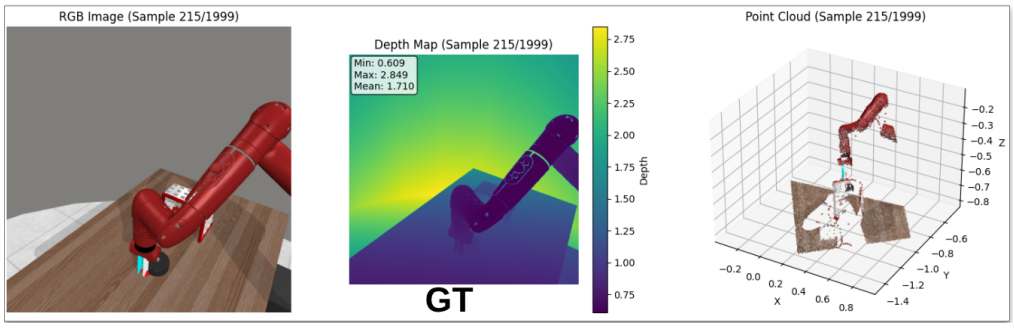
TABLE II: DINOv2 Pipeline Results (Implicit 3D Features)

Task	Res	Size	Success (%)
Hammer	Low	128×128	**85.0**
Basketball	Low	128×128	80.0
Basketball	High	512×512	50.0
Sweep Into	Low	128×128	50.0
Dial Turn	Low	128×128	15.0
Shelf Place	Low	128×128	5.0

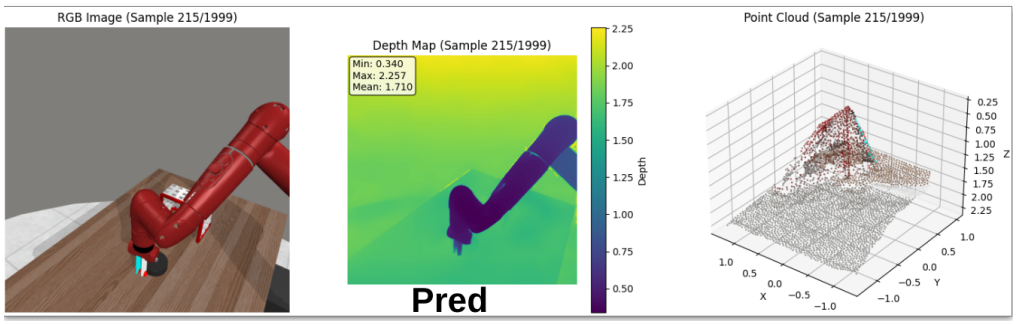
VI. Conclusion

This work systematically explored the feasibility of replacing hardware depth sensors with "software-only" inferred geometry for robotic manipulation. Our results highlight a distinct trade-off between geometric explicitness and semantic understanding.

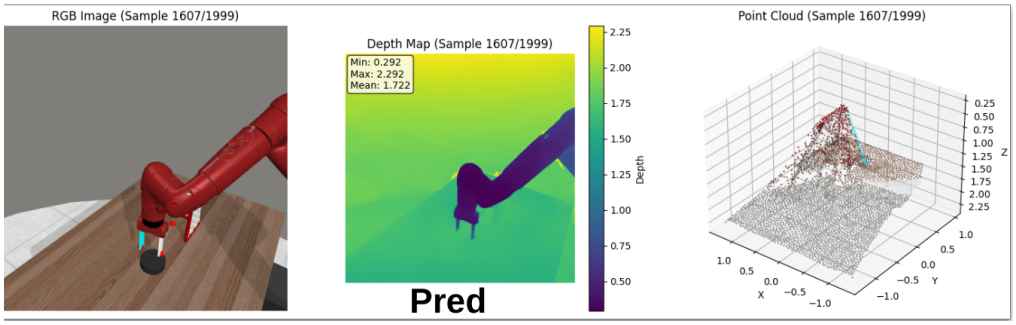
The **Depth Anything v3 pipeline** proved that single-view RGB can successfully emulate depth sensors for planar tasks, achieving 100% success rates when the workspace is constrained to a table surface. However, the lack of temporal consistency in the predicted depth maps caused catastrophic failure in dynamic 3D tasks where the manipulator



(a) Ground Truth (RGB-D-PC)

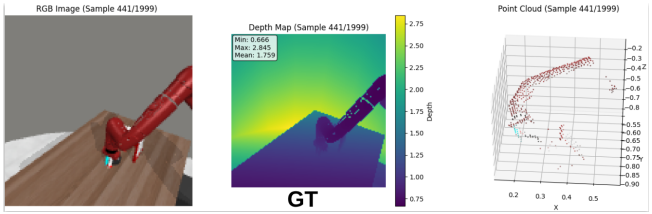


(b) Pseudo-3D (Frame t)

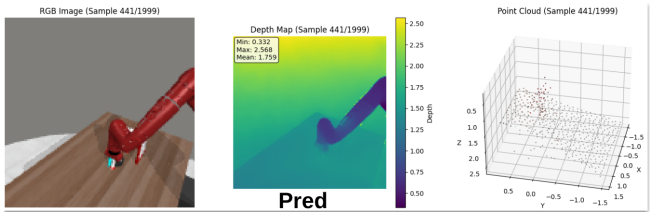


(c) Pseudo-3D (Frame $t + k$)

Fig. 4: High-Resolution Planar Task (Plate Slide). (a) The Ground Truth geometry. (b) The inferred depth and point cloud from Depth Anything v3. (c) A subsequent timeframe showing that depth estimation remains temporally consistent for planar actions.



(a) Ground Truth (Low Res)

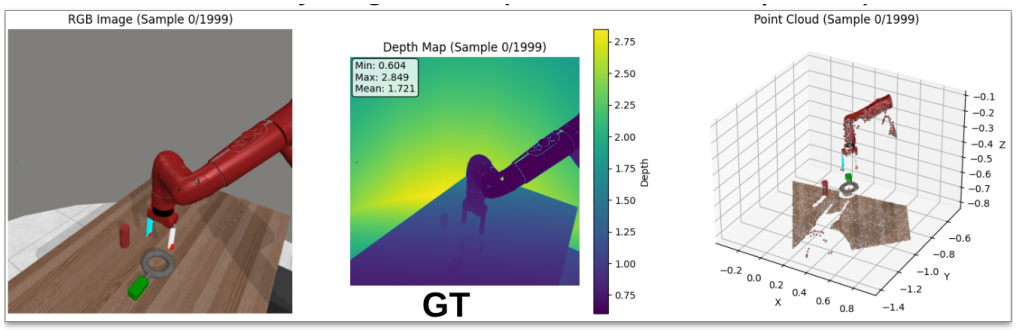


(b) Pseudo-3D (Low Res)

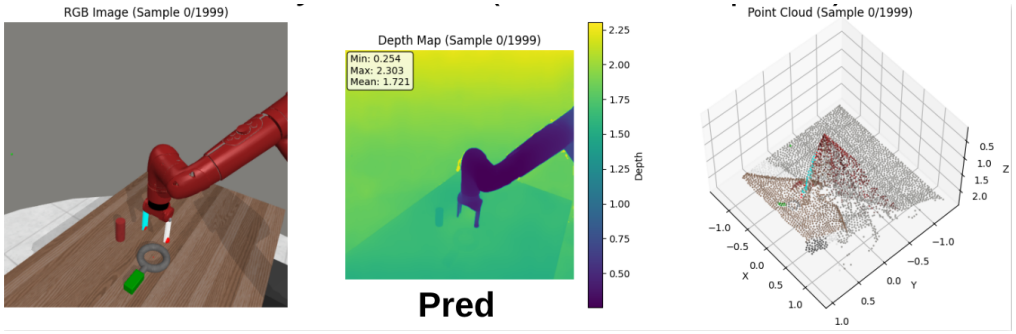
Fig. 5: Low-Resolution Planar Task (Plate Slide). Despite the reduction to 512 points, the inferred geometry (b) retains the structural properties of the ground truth (a), allowing for a 90% success rate.

breaks contact with the surface. Without ground-truth segmentation or stable depth propagation, the generated point clouds became too noisy for the policy to control.

Conversely, the **DINOv2 pipeline** demonstrated that explicit 3D coordinates are not always neces-



(a) Ground Truth



(b) Pseudo-3D (Frame t)

Fig. 6: Failure Case: Assembly Task. While ground truth (a) is stable, the inferred depth maps (b) show significant inconsistency and artifacts as the manipulator moves compared with the GT, causing policy failure (0% success).

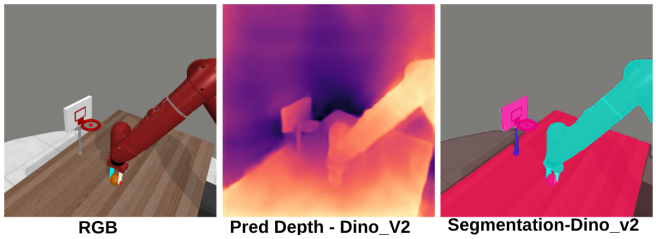


Fig. 7: Qualitative Analysis of Pipeline 2 (DINOv2). This frame visualizes the internal features extracted by the backbone. **Left:** The input RGB observation. **Middle:** The implicit depth map inferred from the attention features. **Right:** The PCA-based segmentation mask, demonstrating that DINOv2 naturally separates objects from the background without explicit supervision.

sary. By leveraging the implicit spatial awareness of large-scale Vision Transformers, we achieved high success rates (up to 85%) with significantly improved sample efficiency—reaching convergence in half the iterations of the baseline. However, this approach is computationally heavier and currently struggles to exploit high-resolution inputs effectively.

Ultimately, while inferred 3D representations show promise for scaling robot learning on internet-scale 2D data, our results indicate they currently lack the precision and temporal consistency of physical sensors for complex, fine-grained manipulation. To bridge this gap, future work should prioritize **multi-view reconstruction** techniques and temporally-consistent depth estimators, such as **UniK3D**, which can effectively propagate geometric cues across frames. Additionally, adopting more advanced 3D-aware backbones like **VGGT** or **DINOv3**—specifically trained to predict geometric information—could significantly enhance the model's ability to infer robust spatial structures directly from 2D inputs.

References

[1] X. Huang, J. Singh, et al. *OpenVLA: Scaling Vision-Language-Action Models for General Robot Control*, arXiv preprint arXiv:2410.23766, 2024.

[2] NVIDIA Research. *GR00T-N1: A Foundation Model for Humanoid Perception and Action*, Technical Report, 2025.

[3] B. Ichter, A. Brohan, et al. π_0 *and* $\pi_{0.5}$*: Open Foundation Models for Robot Learning*, arXiv preprints arXiv:2410.24164 (for π_0) and arXiv:2504.16054 (for $\pi_{0.5}$), 2024–2025.

[4] A. Shridhar, et al. *PerAct: Perception-Action Transformers for Robotic Manipulation*, Conference on Robot Learning (CoRL), 2022. arXiv preprint arXiv:2209.05451.

[5] H. Chi, et al. *Diffusion Policy: Visuomotor Policy Learning via Action Diffusion*, Robotics: Science and Systems (RSS), 2023. arXiv preprint arXiv:2303.04137.

[6] Y. Gao, et al. *ManiCM: Consistent 3D Diffusion Policies for Robotic Manipulation*, arXiv preprint arXiv:2406.01586, 2024.

[7] T. Nguyen, et al. *PointVLA: Enhancing Vision-Language-Action Models with 3D Tokens*, arXiv preprint arXiv:2503.07511, 2025.

[8] Y. Ze, et al. *3D Diffusion Policy: Generalizable Visuomotor Policy Learning via Simple 3D Representations*, arXiv preprint arXiv:2403.03954, 2024.

[9] H. Lin, et al. *Depth Anything 3: Recovering the Visual Space from Any Views*, arXiv preprint arXiv:2511.10647, 2025.

[10] M. Oquab, et al. *DINOv2: Learning Robust Visual Features without Supervision*, arXiv preprint arXiv:2304.07193, 2023.

[11] T. Yu, et al. *Meta-World: A Benchmark and Evaluation for Multi-Task and Meta Reinforcement Learning*, Conference on Robot Learning (CoRL), 2019.