Geometry-Aware Adaptation of Vision Foundation Models for 3D Affordance Perception

Lin Wu, Zhihao Lin, Mahmud Zango, Yanping Wu, and Jianglin Lan

***Abstract*— Understanding 3D affordances, i.e., identifying functionally relevant regions on objects, is critical for robotic manipulation and human–robot interaction. However, sparse point clouds and lack of texture make it challenging to leverage pretrained 2D vision foundation models (VFMs) such as DINO. Existing methods encode multi-view depth renderings, but domain gaps limit part-aware feature learning and reduce discriminability. We propose a geometry-aware adaptation framework that integrates a trainable geometric branch and LoRA modules, leveraging depth and normal information to adapt DINO to sparse 3D observations. A two-stage design enables cross-domain knowledge transfer while enforcing multi-view functional consistency. Experiments demonstrate improved affordance localization and robustness, supporting more reliable perception for downstream robotic interaction.**

I. INTRODUCTION

Affordance is a fundamental concept that describes the action possibilities an object or environment offers to an individual, conditioned on both the agent's capabilities and the object's properties. In 3D scenes, understanding affordances aims at identifying functionally meaningful and interactive regions on objects conditioned on semantic cues such as natural language descriptions [1]. It is a core capability for intelligent systems with wide applications in robotic perception, manipulation, and animation.

As illustrated in Fig. 1, successful interaction depends not only on recognizing objects but also on accurately localizing their functionally critical regions. For example, when a robotic system grasps a knife, focusing on the central handle region enables stable manipulation, while attending to less relevant regions may lead to suboptimal outcomes. This highlights that affordance understanding is essential for bridging perception and action in robotic systems.

Recent advances in affordance understanding have been largely driven by powerful visual foundation models (VFMs) pretrained on large-scale natural image datasets [2]. While these models demonstrate remarkable capability in extracting high-level semantics from 2D images, extending them to *3D affordance understanding* remains challenging due to the mismatch between data modalities and semantic structures. In practice, most existing methods, such as GEAL [3], adopt a *render-and-recognize* paradigm: 3D objects represented by point clouds or Gaussian primitives [4] are first rendered into multi-view images, and then processed by pretrained 2D vision encoders. The extracted visual features are subsequently fused with text-based affordance queries to predict affordance activation fields.

All authors are with the James Watt School of Engineering, University of Glasgow, Glasgow G12 8QQ, U.K.

Corresponding author: Jianglin Lan (Jianglin.Lan@glasgow.ac.uk).

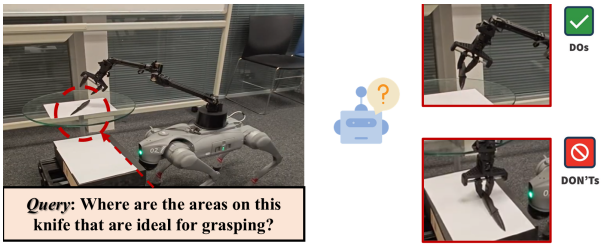

Fig. 1. An example of affordance-aware robotic interaction. Accurate localization of functionally critical regions enables stable manipulation, while inaccurate perception may lead to suboptimal or unstable interactions.

In 3D affordance understanding, object point clouds are typically sparse and lack color information, making simple renderings insufficient for effective feature extraction [5]. For example, GEAL [3] converts depth maps to RGB in order to better adapt to pretrained VFMs. However, this pipeline suffers from a critical domain gap. VFMs such as DINO [2] are pretrained on natural RGB images, whereas rendered depth maps and synthetic views exhibit drastically different appearance statistics. As a result, directly feeding depth renderings into these encoders fails to activate their inherent *partial recognition* and *part-aware* capabilities. Furthermore, when projected back into 3D space, the extracted features often collapse into low-discriminative embeddings, which undermines consistent supervision and limits the effectiveness of 3D affordance reasoning. This domain gap represents a key obstacle that prevents pretrained VFMs from fully leveraging their semantic understanding in 3D scenarios.

Motivated by these observations, we propose a geometry-aware adaptation framework to better transfer semantic knowledge from visual foundation models (VFMs) to sparse 3D data. Instead of directly applying pretrained encoders to depth renderings, our method introduces a geometry-aware mechanism that aligns 3D observations with the inductive biases of VFMs, bridging the gap between rendered geometry and 2D features. This enables more reliable and discriminative 3D affordance understanding.

In summary, our contributions are threefold:

- We identify a key limitation in existing pipelines: directly learning from depth renderings leads to weak part-aware features and reduced discrimination.
- We propose a geometry-aware adaptation strategy that enables DINO to effectively handle sparse 3D observations while enforcing multi-view consistency.
- We develop a two-stage framework to transfer 2D

Manuscript 242 submitted to 2026 IEEE International Conference on Systems, Man, and Cybernetics (SMC). Received March 18, 2026.

semantic knowledge to 3D affordance understanding, improving robustness and generalization.

II. Related Work

a) 3D Affordance Learning: 3D affordance understanding has attracted increasing attention due to its applicability in interactive systems. Early works, such as 3D AffordanceNet [5], established benchmarks that inspired subsequent methods. For instance, IAGNet [6] grounds 3D object affordances by aligning object region features from images and modeling interactive contexts. OpenAD [7] and KD-TPC [8] further explores open-vocabulary affordance detection by integrating textual information with geometric representations through contrastive learning.

Despite these advances, understanding affordances from sparse point clouds remains challenging due to the lack of texture and color information, as well as the presence of noise. Recent works have leveraged pretrained vision foundation models or vision-language models to enhance 3D perception, including GEAL [3] and LMAffordance3D [9]. Notably, GEAL introduces a 3D-to-2D rendering pipeline that enables the use of pretrained 2D models such as DINO, and employs a two-stage design to transfer 2D knowledge to 3D without adding inference cost to the 3D branch. This approach demonstrates a promising direction, and our method builds upon this idea by further enhancing geometry-guided adaptation for 3D affordance learning.

b) Pretrained 2D Vision Foundation Models: Pretrained 2D vision foundation models (VFMs), such as DINO [2] or SAM [10], provide robust high-level semantic representations from large-scale natural images. Their strong generalization and partial recognition capabilities make them promising for enhancing 3D affordance learning, where texture and color information is often missing.

Applying VFMs to sparse point cloud renderings is nontrivial, as these inputs differ substantially from natural images. Previous methods feed multiview depth images into a frozen DINO encoder and fuse the resulting features with text queries to predict affordance activation maps [3]. However, the sparse and synthetic nature of these images limits the discriminative power of the features, offering weak guidance to the 3D branch. Parameter-efficient adaptation methods such as LoRA [11] have shown success in adapting large pretrained models to downstream tasks with minimal parameter overhead. To fill this gap, we focus on adapting VFMs to sparse 3D geometry, aiming to better exploit their partial recognition capability for downstream 3D understanding.

III. Methodology

We adapt a pretrained DINO for 3D affordance learning using a two-stage framework. The stage1 performs 2D adaptation with discriminative and affordance supervision, and the stage2 guides 3D affordance prediction using the learned 2D semantic representations via a consistency loss.

A. Multi-View Rendering

As shown in Fig. 2, we first use a differentiable renderer to project the point cloud into multi-view 2D representations. Specifically, given an object point cloud $O \in \mathbb{R}^{N\times 3}$, we sample $V = 12$ viewpoints $\{\mathbf{c}_v\}_{v=1}^{V}$ and project the point cloud onto each image plane. For each viewpoint $\mathbf{c}_v$, the renderer produces three geometry-driven maps:

$$M_v = \mathcal{R}_{\text{geom}}(O, \mathbf{c}_v), \tag{1}$$

$$D_v = \mathcal{R}_{\text{depth}}(O, \mathbf{c}_v), \tag{2}$$

$$N_v = \mathcal{R}_{\text{normal}}(O, \mathbf{c}_v), \tag{3}$$

where M_v is a validity mask indicating whether a pixel receives any contribution from the projected point cloud, and D_v and N_v denote the depth map and the normal map, respectively, *with depth encoding per-pixel distance to the surface and normals capturing local surface orientation, thereby providing complementary geometric cues*.

Beyond pixel-wise observations, the renderer also provides pixel-to-point correspondence to preserve the geometric traceability:

$$\Pi_v : (x, y) \rightarrow \{(p_i, w_i)\}, \tag{4}$$

which associates each pixel location (x, y) with its contributing 3D point(s) p_i and the corresponding contribution weights w_i. This mapping explicitly encodes how image evidence originates from 3D geometry, enabling bidirectional information transfer between image space and point space.

B. Geometry-Aware DINO Adaptation

VFMs such as DINO are pretrained on natural images and encode strong part-level semantic priors. In sparse 3D affordance learning, objects are observed only through rendered geometry (depth maps D_v, normal maps N_v, and validity masks M_v), which differ significantly from natural images. Instead of treating DINO as a generic feature extractor, we adapt it so that sparse geometric inputs can activate its latent part-awareness without overwriting its pretrained knowledge.

a) Depth-to-RGB Translation: DINO expects three-channel RGB input. Following [3], we deterministically convert each depth map into a pseudo-RGB image as follows:

$$I_v^{\text{rgb}} = \mathcal{C}(D_v), \tag{5}$$

where $\mathcal{C}$ is a fixed colorization operator. This operation reshapes the raw depth information into a format interpretable by DINO's convolutional filters without introducing additional geometric reasoning.

b) Geometry-aware Token Embedding: Let $\mathbf{x}_v \in \mathbb{R}^{P\times d}$ denote the patch embeddings and $\mathbf{p}_v$ the original positional encoding of DINO for view v. To inject geometric awareness at the token level, we introduce a geometry encoder $\mathcal{E}_g$ that extracts patch-aligned geometric descriptors:

$$\mathbf{g}_v = \mathcal{E}_g(N_v, M_v), \tag{6}$$

where $\mathbf{g}_v \in \mathbb{R}^{P\times d}$ is projected to match the embedding dimension. The encoder $\mathcal{E}_g$ follows the same patch partitioning scheme as DINO to preserve the token correspondence.

Manuscript 242 submitted to 2026 IEEE International Conference on Systems, Man, and Cybernetics (SMC). Received March 18, 2026.

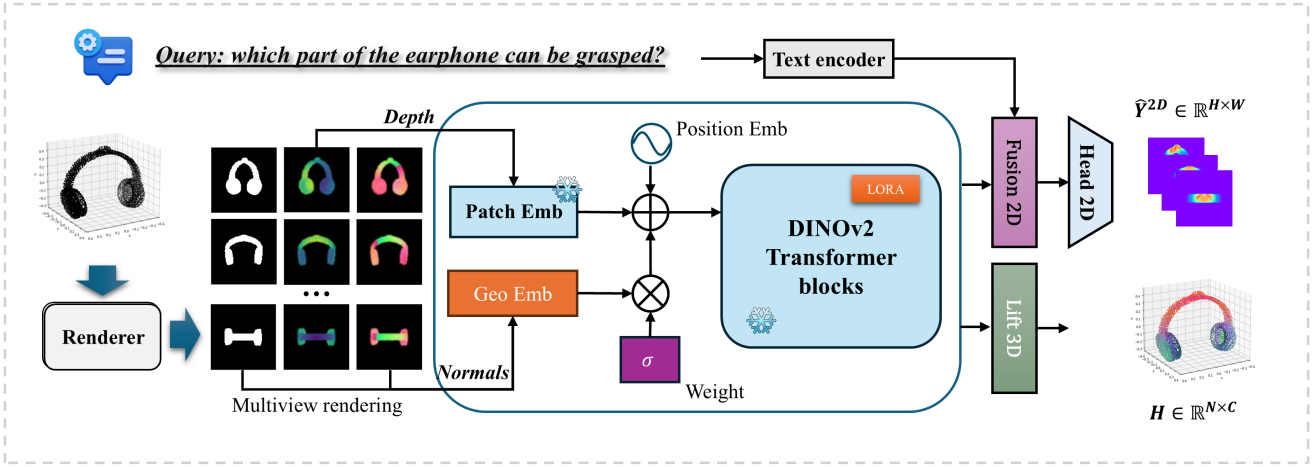

Fig. 2. Overview of our geometry-aware adaptation framework for affordance perception. Multi-view geometric renderings of sparse point clouds are generated using a differentiable renderer. Depth and normal with validity cues are encoded by separate patch embedders and fused as input to the transformer blocks. Training is guided by dual supervision, including a discriminative loss on lifted 3D features and a text-driven 2D affordance prediction loss.

Instead of replacing pretrained features, geometry is incorporated through residual modulation:

$$\tilde{\mathbf{x}}_v = \mathbf{x}_v + \sigma(\gamma) \cdot \mathbf{g}_v, \tag{7}$$

where $\sigma(\cdot)$ denotes the sigmoid function, constraining the gating factor to $(0, 1)$, and γ is a learnable parameter that controls the contribution of the geometric features. This formulation interprets geometry injection as a conditioning signal that perturbs patch tokens in a controlled and stable manner.

c) Geometry-Aware Positional Encoding: The positional encoding in DINO provides a fixed spatial prior over patch locations. Instead of learning a new encoding or altering its values, we reuse DINO's original positional embedding and allow geometry to influence spatial reasoning implicitly through geometry-conditioned token content. Formally, the input to the Transformer is: $\mathbf{z}_v = \tilde{\mathbf{x}}_v + \mathbf{p}_v$. Although the positional basis itself is unchanged, the geometry-modulated tokens ensure that spatial attention becomes geometry-sensitive, as pairwise interactions between patches now depend on geometry-conditioned query and key representations.

d) Parameter-Efficient Adaptation via LoRA: To accommodate geometric inputs while avoiding catastrophic forgetting, we apply Low-Rank Adaptation (LoRA) [11] to DINO. Each adapted linear layer is modified as: $\mathbf{W} \leftarrow \mathbf{W} + \mathbf{AB}$, where $\mathbf{W}$ is frozen and $\mathbf{A}, \mathbf{B}$ are learnable low-rank matrices. LoRA is applied to the query and value projection matrices in all transformer blocks, configured with rank $r = 4$, and scaling factor $\alpha = 8$. Although the geometry encoder branch already extracts geometric features, LoRA allows the backbone to adjust to geometry-conditioned tokens without altering its pretrained semantic parameters. This adaptation constrains learning to a low-dimensional subspace, providing a controlled mechanism to incorporate geometric information.

e) Feature Extraction and Lifting: After adaptation, DINO produces geometry-aware features for each rendered view v of the object through

$$\mathbf{F}_v = \mathcal{F}_{\text{DINO}}(D_v, N_v, M_v). \tag{8}$$

This integrates the pretrained semantic organization with geometry-induced structural cues.

Using the known camera intrinsics and depth, each 2D patch feature is back-projected to its corresponding 3D surface points. Formally, for a surface point $\mathbf{x}$ visible in view v, we have

$$\mathbf{h}_v(\mathbf{x}) = \mathbf{F}_v(\pi_v(\mathbf{x})), \tag{9}$$

where $\pi_v(\mathbf{x})$ maps the 3D point $\mathbf{x}$ to its corresponding patch in view v.

To consolidate multi-view information, we aggregate features across all visible views for each surface point using

$$\mathbf{H}(\mathbf{x}) = \frac{\sum_v w_v(\mathbf{x})\mathbf{h}_v(\mathbf{x})}{\sum_v w_v(\mathbf{x})}, \tag{10}$$

where $w_v(\mathbf{x})$ denotes the rendering-derived contribution weight of point $\mathbf{x}$ in view v.

The resulting 3D feature $\mathbf{H}$, aligned with the object geometry, lifts per-view DINO features into an object-centric representation that preserves both the pretrained semantic priors and geometry-aware cues for downstream 2D–3D alignment.

C. Part-Aware and Text-Aligned Representation Learning

Affordance learning involves both part-level discrimination and point-wise probability prediction. To capture these complementary objectives, we introduce two types of supervision: (i) a 3D discriminative loss to encourage part-aware feature representations, and (ii) a 2D prediction loss to supervise text-fused affordance probability maps.

a) 3D Discriminative Loss: Given the 3D feature embedding $\mathbf{H} \in \mathbb{R}^{N\times C}$ and a soft probability mask $\mathbf{m}_q \in [0,1]^N$ for the q-th affordance query, we define a prototype representation as the following weighted mean of points:

$$\boldsymbol{\mu}_q = \frac{\sum_i m_{q,i}\,\mathbf{H}_i}{\sum_i m_{q,i} + \epsilon}, \tag{11}$$

where i indexes over all the N points in the point cloud.

Intra-class compactness and inter-class separation are enforced via

$$\mathcal{L}_{\text{intra}} = \frac{1}{|Q^+|}\sum_{q\in Q^+}\frac{\sum_i m_{q,i}\|\mathbf{H}_i - \boldsymbol{\mu}_q\|_2^2}{\sum_i m_{q,i}+\epsilon}, \tag{12}$$

$$\mathcal{L}_{\text{inter}} = \frac{1}{|Q^+|(|Q^+|-1)}\sum_{q\neq r}\|\boldsymbol{\mu}_q - \boldsymbol{\mu}_r\|_2^2, \tag{13}$$

and the final discriminative prototype loss is defined as

$$\mathcal{L}_{\text{dis}} = \log\left(1 + \frac{\mathcal{L}_{\text{intra}}}{\mathcal{L}_{\text{inter}}+\epsilon}\right), \tag{14}$$

where ϵ is a small constant (10^{-6}) for ensuring numerical stability. This loss encourages the adapted DINO to produce part-aware, discriminative embeddings in 3D space.

b) 2D Prediction Loss: After DINO feature extraction, per-view features are fused with text-based affordance queries and passed through a prediction head to produce 2D probability maps. For this branch, we employ a combination of Binary Cross-Entropy (BCE) and Dice loss to supervise the predicted 2D probability maps.

Integrating the above two objectives gives the Stage 1 objective $\mathcal{L}_{\text{stage1}} = \alpha\cdot\mathcal{L}_{\text{dis}} + \mathcal{L}_{\text{2D}}$, which enables the model to learn part-aware 3D representations from lifted features that preserve pretrained semantic priors and geometry-aware cues, while producing text-aligned probabilistic predictions. The discriminative supervision further aligns features across views, promoting multi-view functional consistency.

D. 2D–3D Consistency Alignment

Since Stage 1 produces per-view 2D features fused with text queries, we introduce an alignment mechanism in Stage 2 to ensure semantic consistency between the 3D and 2D branches. The 3D branch extracts point-wise geometric features using PointNet++ [12] and fuses them with the corresponding text-based affordance queries, yielding 3D semantic features $\mathbf{f}^{\text{3D}}$.

Using the previously defined pixel-to-point correspondence Π_v, each 3D point feature $\mathbf{f}_i^{\text{3D}}$ is projected onto the 2D image plane of view v, producing multi-view 2D-projected features $\mathbf{f}_{j,v}^{\text{3D}\to\text{2D}}$ corresponding to pixels j. Independently, the 2D branch generates cross-modal fused features $\mathbf{f}_{j,v}^{\text{2D}}$ for each pixel by combining per-view visual features with the text queries.

To enforce semantic alignment, we apply a pixel-wise mean squared error over all pixels and views, given as

$$\mathcal{L}_{\text{align}} = \frac{1}{VM}\sum_{v=1}^{V}\sum_{j=1}^{M}\left\|\mathbf{f}_{j,v}^{\text{3D}\to\text{2D}} - \mathbf{f}_{j,v}^{\text{2D}}\right\|_2^2, \tag{15}$$

where M is the number of pixels per view.

In parallel, the 3D branch is supervised with task-specific losses (Binary Cross-Entropy and Dice) for point-wise affordance prediction. The overall Stage 2 training objective is: $\mathcal{L}_{\text{stage2}} = \mathcal{L}_{\text{3D}} + \beta\cdot\mathcal{L}_{\text{align}}$, where β is set as 0.1 to balance the two terms. The alignment is applied only during training, introducing no additional inference cost.

IV. Experiments

a) Dataset: We conduct experiments on LASO [13] and PIAD [6], two benchmarks that provide paired 3D point clouds and affordance annotations. LASO consists of 19,751 point cloud question pairs over 8,434 objects from 23 object classes and 17 affordance categories. PIAD contains 7,012 point clouds from the same categories, where some objects are completely unseen during training, offering a setting to examine the generalization to novel objects. Since PIAD does not include language annotations, we reuse the questions from LASO. We further evaluate performance on corrupted data built upon LASO [3]. Models are trained on clean data and tested under seven atomic corruption types including *Scale, Jitter, Rotate, Drop Global, Drop Local, Add Global, and Add Local*. Each corruption introduces a specific form of perturbation and is applied at five severity levels, allowing analysis of model behavior under increasing corruption strength.

b) Baselines and Metrics: We compare our method with state-of-the-art 3D affordance detection approaches, including GEAL [3] and IAGNet [6]. GEAL is a strong baseline that incorporates pretrained DINO with multi-view rendering and represents a competitive performance benchmark. Performance is reported using the standard metrics: aIoU [14], which quantifies the overlap between predicted and ground-truth affordance regions; AUC [15], which measures the ranking quality of point-wise predictions; SIM [16], which evaluates distribution similarity by summing the minimum values at each point; and MAE [17], which computes the average absolute error between predictions and ground truth. For fair evaluation, we adopt multi-seed voting, where predictions are averaged across multiple runs to reduce inference randomness.

c) Implementation Details.: Our model is implemented in PyTorch and trained using the Adam optimizer [18] for both stages. The base learning rate is 1×10^{-4} for most parameters, with a lower learning rate of 5×10^{-6} applied to the text encoder (RoBERTa) [19]. Weight decay is set as 1×10^{-3}, and a step learning rate scheduler reduces the learning rate by a factor of 0.5 every 10 epochs. Training is performed for 50 epochs on a single NVIDIA A6000 GPU with a batch size of 16. During training, the 2D backbone DINOV2 remains frozen except for the geometry encoder branch and LoRA parameters.

A. Quantitative Results

Tables I and II report the quantitative results on PIAD and LASO datasets under Seen and Unseen splits. Affordance prediction is inherently a probabilistic task, which requires

Manuscript 242 submitted to 2026 IEEE International Conference on Systems, Man, and Cybernetics (SMC). Received March 18, 2026.

balancing discriminative accuracy and regression quality. Our method achieves competitive performance across most metrics on both datasets. Specifically, it demonstrates leading aIoU scores while maintaining low MAE, indicating accurate and consistent probabilistic predictions. On Unseen splits, our approach outperforms the strong baselines such as GEAL in the overall performance, highlighting its generalization ability for novel object-affordance pairs.

TABLE I

QUANTITATIVE RESULTS ON THE PIAD DATASET UNDER SEEN/UNSEEN SPLITS. (↑ HIGHER IS BETTER, ↓ LOWER IS BETTER.)

Task	Method	aIoU ↑	AUC ↑	SIM ↑	MAE ↓
Seen	FRCNN [20]	12.0	76.1	0.429	0.136
	PointFusion [21]	12.3	77.5	0.432	0.135
	IAGNet [6]	20.5	84.9	0.545	0.098
	GEAL [3]	22.0	85.1	**0.588**	0.098
	Ours	**23.2**	**85.6**	**0.588**	**0.097**
Unseen	FRCNN [20]	5.1	61.9	0.332	0.195
	PointFusion [21]	5.3	61.9	0.330	0.193
	IAGNet [6]	8.0	71.8	0.352	0.127
	GEAL [3]	7.9	71.4	0.372	**0.106**
	Ours	**8.1**	**73.2**	**0.382**	**0.106**

TABLE II

QUANTITATIVE RESULTS ON THE LASO DATASET UNDER SEEN/UNSEEN SPLITS. (↑ HIGHER IS BETTER, ↓ LOWER IS BETTER).

Task	Method	aIoU ↑	AUC ↑	SIM ↑	MAE ↓
Seen	ReferTrans [22]	13.7	79.8	0.497	0.124
	3D-SPS [23]	11.4	76.2	0.433	0.138
	IAGNet [6]	17.8	82.3	0.561	0.109
	GEAL [3]	20.4	85.3	**0.610**	0.104
	Ours	**21.4**	**85.8**	0.605	**0.101**
Unseen	ReferTrans [22]	10.2	69.1	0.432	0.145
	3D-SPS [23]	7.9	68.8	0.402	0.158
	IAGNet [6]	12.9	77.8	0.443	0.129
	GEAL [3]	17.5	**81.8**	0.535	**0.106**
	Ours	**18.1**	81.0	**0.541**	**0.106**

B. Qualitative Results

Fig. 3 presents a comparison between our method and GEAL. Our approach achieves more accurate localization of affordance regions and improved point-wise prediction strength. For instance, it correctly identifies the valve region of a tap and the display area of a clock, capturing both spatial extent and functional relevance more reliably than the baseline, demonstrating the effectiveness of our method.

C. Further Discussion

a) Functional Part Discrimination: To evaluate the effectiveness of our method in distinguishing 3D functional regions, we analyze the multi-view DINO features extracted in Stage 1. These features are lifted to 3D and visualized using Principal Component Analysis (PCA) for dimensionality reduction, as shown in Fig. 4. The input point clouds are sparse and colorless. When using GEAL's representations, the visualization shows that features from different views are largely similar, capturing only the overall object semantics. In contrast, our method produces lifted 3D features that effectively separate different functional regions of the object, demonstrating multi-view functional consistency.

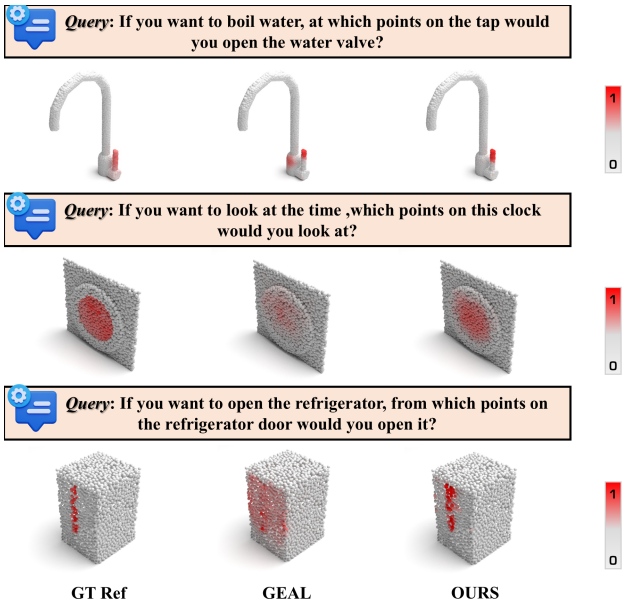

Fig. 3. Comparison of predicted affordance regions between our method and GEAL, showing improved localization and point-wise accuracy.

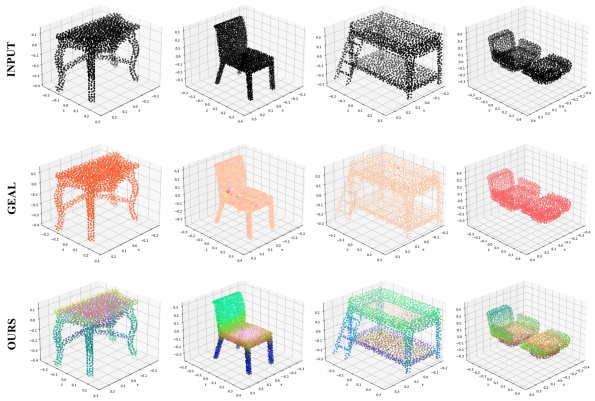

Fig. 4. PCA visualization of lifted 3D features showing functional part discrimination. Our method separates different functional regions, while features extracted from GEAL mainly capture overall object semantics.

b) Performance under Data Corruptions: We evaluate robustness by testing models on corrupted inputs after training on clean data. For each severity level, we report the average performance across all corruption types to assess the impact of increasing degradation. As shown in Table III, our method consistently outperforms GEAL, achieving higher AUC and aIoU while maintaining lower MAE under stronger corruptions. This improvement can be attributed to the dual-branch architecture and the 2D–3D consistency alignment, which leverage the pretrained 2D features and enforce cross-modal semantic consistency.

TABLE III

PERFORMANCE UNDER INCREASING CORRUPTION SEVERITY. FOR EACH METRIC, RESULTS ARE REPORTED ACROSS FIVE SEVERITY LEVELS

Metric	Method	S1	S2	S3	S4	S5
IOU ↑	GEAL	0.185	0.179	0.173	0.165	0.154
	Ours	**0.191**	**0.185**	**0.178**	**0.169**	**0.158**
AUC ↑	GEAL	0.836	0.830	0.825	0.818	0.808
	Ours	**0.842**	**0.837**	**0.832**	**0.823**	**0.813**
MAE ↓	GEAL	0.107	0.107	0.108	0.108	0.109
	Ours	**0.105**	**0.105**	**0.106**	**0.106**	**0.107**

c) Adaptation: Table IV presents an ablation study of Stage 1 variants on the LASO dataset. The Depth configuration corresponds to a frozen DINO branch with depth-to-RGB input, which is equivalent to the GEAL baseline. Normals denotes our trainable geometry branch, while LoRA refers to low-rank adaptation applied to the attention layers of the DINO transformer. Introducing the trainable normals branch leads to a clear improvement in 3D affordance prediction. Further incorporating LoRA yields a modest additional gain, suggesting that lightweight model adaptation complements geometry-guided features.

TABLE IV

STAGE 1 ABLATION ON MODEL VARIANTS AND THEIR IMPACT ON 3D AFFORDANCE PREDICTION. † INDICATES TRAINABLE COMPONENTS.

Depth	Normals†	LoRA†	aIoU↑	AUC↑	MAE↓
✓			20.4	85.3	0.104
✓	✓		21.1	85.5	0.103
✓	✓	✓	21.4	85.8	0.101

V. CONCLUSION AND LIMITATION

In this paper, we present a lightweight geometry-aware Adaptation for 3D affordance learning. The proposed method incorporates a trainable geometric branch and LoRA to activate DINO's understanding of multi-view rendered sparse point clouds. Experiments demonstrate that the learned representations capture part-aware functional regions in 3D space and contribute to improved downstream 3D affordance understanding. Although effective, the method requires depth and normal inputs, which adds some complexity to the pipeline. Future research will investigate more efficient adaptation strategies and explore self-supervised geometric cues to further simplify the pipeline while maintaining robust functional reasoning for intelligent robotic applications.

REFERENCES

[1] A. Delitzas, A. Takmaz, F. Tombari, R. Sumner, M. Pollefeys, and F. Engelmann, "Scenefun3d: Fine-grained functionality and affordance understanding in 3d scenes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 14 531–14 542.

[2] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby *et al.*, "Dinov2: Learning robust visual features without supervision," *arXiv preprint arXiv:2304.07193*, 2023.

[3] D. Lu, L. Kong, T. Huang, and G. H. Lee, "Geal: Generalizable 3d affordance learning with cross-modal consistency," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 1680–1690.

[4] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering." *ACM Trans. Graph.*, vol. 42, no. 4, pp. 139–1, 2023.

[5] S. Deng, X. Xu, C. Wu, K. Chen, and K. Jia, "3d affordancenet: A benchmark for visual object affordance understanding," in *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 1778–1787.

[6] Y. Yang, W. Zhai, H. Luo, Y. Cao, J. Luo, and Z.-J. Zha, "Grounding 3d object affordance from 2d interactions in images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 10 905–10 915.

[7] T. Nguyen, M. N. Vu, A. Vuong, D. Nguyen, T. Vo, N. Le, and A. Nguyen, "Open-vocabulary affordance detection in 3d point clouds," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 5692–5698.

[8] T. Van Vo, M. N. Vu, B. Huang, T. Nguyen, N. Le, T. Vo, and A. Nguyen, "Open-vocabulary affordance detection using knowledge distillation and text-point correlation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 13 968–13 975.

[9] H. Zhu, Q. Kong, K. Xu, X. Xia, B. Deng, J. Ye, R. Xiong, and Y. Wang, "Grounding 3d object affordance with language instructions, visual observations and interactions," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 17 337–17 346.

[10] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson *et al.*, "Sam 2: Segment anything in images and videos," *arXiv preprint arXiv:2408.00714*, 2024.

[11] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "LoRA: Low-rank adaptation of large language models," in *Proceedings of the International Conference on Learning Representations*, 2022.

[12] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[13] Y. Li, N. Zhao, J. Xiao, C. Feng, X. Wang, and T.-s. Chua, "Laso: Language-guided affordance segmentation on 3d object," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 14 251–14 260.

[14] M. A. Rahman and Y. Wang, "Optimizing intersection-over-union in deep neural networks for image segmentation," in *International symposium on visual computing*. Springer, 2016, pp. 234–244.

[15] J. M. Lobo, A. Jiménez-Valverde, and R. Real, "Auc: a misleading measure of the performance of predictive distribution models," *Global ecology and Biogeography*, vol. 17, no. 2, pp. 145–151, 2008.

[16] M. J. Swain and D. H. Ballard, "Color indexing," *International journal of computer vision*, vol. 7, no. 1, pp. 11–32, 1991.

[17] C. J. Willmott and K. Matsuura, "Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance," *Climate research*, vol. 30, no. 1, pp. 79–82, 2005.

[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, 2015. [Online]. Available: https://arxiv.org/abs/1412.6980

[19] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[20] X. Xu, S. Dong, T. Xu, L. Ding, J. Wang, P. Jiang, L. Song, and J. Li, "Fusionrcnn: Lidar-camera fusion for two-stage 3d object detection," *Remote Sensing*, vol. 15, no. 7, p. 1839, 2023.

[21] D. Xu, D. Anguelov, and A. Jain, "Pointfusion: Deep sensor fusion for 3d bounding box estimation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 244–253.

[22] M. Li and L. Sigal, "Referring transformer: A one-step approach to multi-task visual grounding," *Advances in neural information processing systems*, vol. 34, pp. 19 652–19 664, 2021.

[23] J. Luo, J. Fu, X. Kong, C. Gao, H. Ren, H. Shen, H. Xia, and S. Liu, "3d-sps: Single-stage 3d visual grounding via referred point progressive selection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 454–16 463.