IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 10 March 2026, accepted 29 March 2026, date of publication 7 April 2026, date of current version 13 April 2026.

Digital Object Identifier 10.1109/ACCESS.2026.3681732

RESEARCH ARTICLE

One-Shot 3-D Affordance Learning for Multi-Stage Robotic Manipulation

HYUNSEO KIM[1], YEON-JI SONG[1], MINSU LEE[2], AND BYOUNG-TAK ZHANG[1,3], (Member, IEEE)

[1]Interdisciplinary Program in Neuroscience, Seoul National University, Seoul 08826, Republic of Korea
[2]School of AI Convergence, Sungshin Women's University, Seoul 02844, Republic of Korea
[3]Artificial Intelligence Institute, Seoul National University, Gwanak-gu, Seoul 08826, Republic of Korea

Corresponding authors: Minsu Lee (mslee@sungshin.ac.kr) and Byoung-Tak Zhang (btzhang@bi.snu.ac.kr)

This work was supported in part by Korea Research Institute for Defense Technology Planning and Advancement (KRIT) grant funded by the Defense Acquisition Program Administration (DAPA), Development of AI Researchers Based on Deep Reinforcement Learning and Establishment of Virtual Combat Experiment Environment, under Grant KRIT-CT-23-003(20%); in part by Institute of Information and Communications Technology Planning and Evaluation (IITP) under Grant RS-2021-II211343-GSAI(10%), Grant RS-2022-II220951-LBA(10%), and Grant RS-2022-II220953-PICA(10%); in part by National Research Foundation of Korea (NRF) under Grant RS-2024-00353991-SPARC(10%), Grant RS-2023-00274280-HEI(10%), and Grant RS-2024-00358416-AutoRL(10%); in part by Korea Planning and Evaluation Institute of Industrial Technology (KEIT) under Grant RS-2025-25453780(10%); and in part by Korea Institute for Advancement of Technology (KIAT) funded by Korean government under Grant RS-2025-25460896(10%).

ABSTRACT Human-assistant robots must understand human-object interactions to execute collaborative manipulation tasks described in natural language. While affordance learning addresses this need, current approaches face a fundamental trade-off: 2D methods capture action-relevant object semantics but lack robust 3D geometric reasoning, whereas 3D methods demand labor-intensive point cloud datasets. To bridge this gap, we propose a one-shot 3D affordance learning framework that grounds action verbs directly into 3D Gaussian Splatting representations. Our key insight is that verb-centric affordances from sparse 2D signals can be lifted into view-consistent 3D representations without explicit 3D annotations. Specifically, our framework requires only a single affordance-labeled image per scene during training, and zero reference images at inference. Operating seamlessly with natural language, this approach eliminates the need for explicit object- or part-level queries, enabling rapid inference crucial for multi-stage tasks. Extensive real-world experiments demonstrate that our approach outperforms baselines reliant on 2D affordances or part-level reasoning, particularly in challenging long-horizon multi-stage settings.

INDEX TERMS 3D vision, vision for robotics, affordance learning.

I. INTRODUCTION

Affordance learning, which enables robots to perceive actionable object properties, is a fundamental component of autonomous robotic manipulation [1], [2]. By capturing how objects can be functionally interacted with, affordances support a wide range of downstream tasks, including grasp generation, motion planning, and scene understanding, particularly in unstructured environments [3]. Importantly, affordances allow robots to reason about interactions at the level of human-intuitive actions, *e.g.*, pick up the basket, rather than requiring explicit, low-level part specifications, *e.g.*, grasp the handle of the basket, making affordances especially suitable for language-guided manipulation.

The associate editor coordinating the review of this manuscript and approving it for publication was Ghazaleh Khodabandelou.

Despite their success, conventional affordance learning approaches are predominantly confined to the 2D image plane, representing affordances as heatmaps or segmentation masks [4], [5]. However, real-world robotic manipulation inherently requires reasoning about 3D spatial structure. To address this limitation, existing methods typically adopt one of two strategies: either projecting these 2D features into 3D space using RGB-D sensors, or developing native 3D models that infer affordances directly from point clouds [6], [7].

However, these paradigms present critical limitations for robust robotic manipulation. First, due to the lack of explicit geometric priors, purely 2D methods often yield inconsistent affordance predictions under the drastic viewpoint changes encountered in dynamic robotic operations [8], [9]. Second, methods that rely on projecting 2D affordance into 3D

space using RGB-D observations are highly susceptible to sensor noise, particularly when encountering transparent or reflective surfaces [9], [10]. While explicit 3D reasoning is essential to overcome these sensory and geometric ambiguities, native point cloud-based approaches introduce a different bottleneck. Lastly, learning affordances directly from point clouds typically requires densely annotated 3D datasets [6]. Collecting such data is not only prohibitively costly but also typically restricted to simulated shapes, creating a significant domain gap when transferring to visually complex real-world objects.

Recent advances in 3D scene representations have motivated the adoption of 3D Gaussian Splatting (3DGS) [11] for the efficient and high-fidelity modeling of complex environments. By representing scenes as differentiable Gaussian primitives, 3DGS enables dense, view-consistent encoding of geometry and appearance, making it an attractive representation for robotic perception. However, leveraging such representations for downstream robotic manipulation remains challenging. Existing approaches that infer 3D affordances directly within 3DGS suffer from similar bottlenecks as point cloud-based methods, namely the reliance on exhaustively annotated 3DGS datasets [12], [13]. To alleviate this annotation burden, some methods employ CLIP-based object recognition; however, they still require explicit part-level specifications to precisely identify interaction regions [14], [15], [16]. Consequently, a significant gap remains between expressive 3D scene representations and semantically grounded, language-conditioned affordances for robust robotic manipulation.

To bridge this gap, we propose **AffoRo-GS**, an affordance learning framework that grounds natural language actions in 3D Gaussian scene representations for multi-stage robotic manipulation. Given an affordance-labeled image, our method learns action-aligned visual features without requiring 3D affordance annotations, relying solely on single-image 2D supervision per scene. These features are subsequently lifted into a dense 3D Gaussian representation, enabling robust and view-consistent localization of actionable regions within the 3D scene. By aligning these 3D features with textual embeddings derived from action verbs, our model enables robots to interpret and execute multi-stage manipulation tasks directly from high-level language commands, without requiring explicit part-level instructions (see Fig. 1).

Our contributions are summarized as follows:

- We propose a one-shot 3D affordance framework that lifts 2D-supervised visual features into 3D Gaussians, eliminating the need for 3D affordance annotations while ensuring view-consistent spatial grounding.
- We extend affordance reasoning from isolated point clouds to entire cluttered 3D Gaussian scenes, enabling high-level semantic language grounding without explicit part-level queries.
- We validate our approach on seven real-world multi-stage robotic manipulation tasks, achieving an average success rate of 40% across challenging action sequences consisting of three to seven steps.

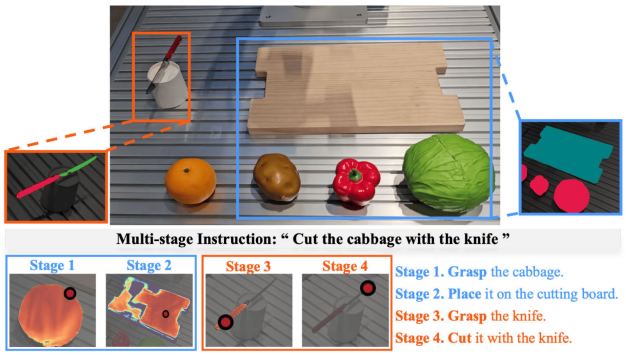

FIGURE 1. Task overview. A natural language instruction, *e.g.*, "Cut the cabbage with the knife", is decomposed into an ordered sequence of manipulation stages. At each stage, the model grounds the specified action, *e.g.*, grasp, place, cut, to a task-relevant 3D affordance region on the corresponding object or tool. The top panel displays the scene and the objects involved, while the bottom panels visualize example affordance localizations for different stages.

II. RELATED WORK

A. AFFORDANCE LEARNING IN ROBOTICS

Affordance learning has emerged as a fundamental paradigm for connecting perception and action in robotics [19], [20], [21]. Early works primarily focused on projecting 2D affordances into 3D maps via RGB-D cameras [22] but required exhaustive manual point cloud annotations. To address this, recent works distill indirect annotations from human-object interaction videos into 3D representations [10], [18], [23]. However, methods like Splat-MOVER [18] extract task-agnostic affordances, lacking the task-relevant and semantically meaningful features needed for robust and complex manipulation.

Recent Vision-Language-Action (VLA) models leverage external large foundational models, such as LLMs and VLMs, for affordance reasoning [20], [24], [25]. While demonstrating strong generalization, their reliance on chain-of-thought reasoning introduces significant inference latency, making them suboptimal for real-time control. Conversely, lightweight language-guided robotic manipulation frameworks use vision-language models like CLIP [26] for target localization [14], [27], [28], [29]. Yet, standard CLIP features lack fine-grained affordance understanding, relying heavily on explicit, part-specific textual descriptions. Consequently, these methods fail to explicitly model the action-centric affordances inherent in natural language instructions.

B. 3D AFFORDANCE

3D affordance learning has been studied as a fundamental mechanism to guide robotic interaction and grasping [6], [7], [9], [30]. Early methods projected affordances from simulated data onto real-world point clouds using architectures like PointNet [31] and establishing geometric correspondences. However, acquiring diverse, manually annotated

TABLE 1. Comparison of our method against prior work on key capabilities for sequential manipulation tasks.

Method	3D Representation	3D Annotation Required	Per-Scene Supervision	References at Inference	3D Scene Language Grounding	Multi-stage Robotic Manipulation
AffCorrs [17]	–	–	Single image	Required	-	No
OOAL [8]	–	–	Single image	Not required	-	-
O^3Afford [9]	Point Cloud	Yes	Two Point Clouds	Not required	-	No
SeqAffordSplat [13]	3DGS	Yes	Many	Not required	Verb-level	No
GraspSplats [14]	3DGS	No	–	Not required	Part-level	Yes
Splat-MOVER [18]	3DGS	No	–	Not required	Part-level	Yes
Ours (AffoRo-GS)	**3DGS**	**No**	**Single image**	**Not required**	**Verb-level**	Yes

3D point cloud datasets remains a highly labor-intensive bottleneck. To mitigate this, 2D-to-3D projection techniques [32], [33], [34], [35] utilize 2D object categories to enhance generalizability and reduce 3D data requirements. Nevertheless, they still fundamentally rely on 3D point clouds for supervision.

Recently, 3D Gaussian Splatting (3DGS) [11] has emerged as a robust, photorealistic alternative to raw point clouds, mitigating occlusion and depth sensor noise. Although current 3DGS-based affordance inference methods [12], [13], [36], [37] yield high geometric precision, their dependence on custom, explicitly annotated 3DGS-based datasets [12], [13] severely restricts scalability. To overcome this limitation, our approach combines 3DGS with a 2D-to-3D projection paradigm. By training solely on 2D images and directly projecting the inferred affordances into 3D, we completely eliminate the need for 3D training data and annotation.

C. ONE-SHOT AFFORDANCE LEARNING

The definition of task-oriented affordances varies significantly across studies, particularly regarding the granularity of segmentation masks or 3D point cloud labels. Consequently, acquiring large-scale, consistently annotated datasets remains a major bottleneck, driving the adoption of one-shot learning [9], [17], [38], [39].

Recent methods leverage foundation models such as DINO [40] or CLIP [26] for low-data reasoning. However, these methods still present practical limitations. For instance, AffCorrs [17] strictly relies on explicitly stored support sets of reference images at inference time to compute feature similarities for retrieving and transferring affordances from the most closely matched reference. Conversely, while methods like OOAL [8] eliminate the need for reference images, they operate primarily in the 2D image plane. By relying on per-view 2D predictions, these techniques fail to produce the spatially coherent and view-consistent 3D representations crucial for reliable real-world robotic manipulation.

While O^3Afford [9] extends one-shot learning to 3D using annotated point clouds, its formulation focuses strictly on object-to-object interactions. It requires explicitly paired source and target point clouds (*e.g.*, a tool and a target object) to predict geometric interaction points. Since their affordances are not language-conditioned and assume a fundamentally different problem setting, a direct comparison with our scene-level framework is infeasible.

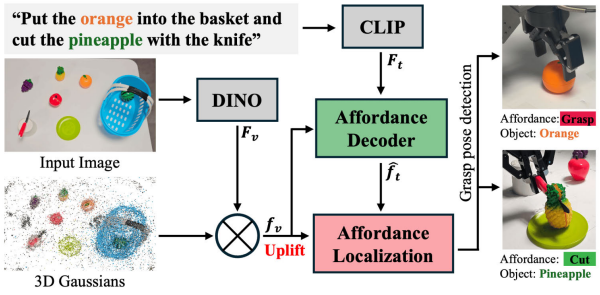

FIGURE 2. Model overview. Our model begins by encoding action verbs from multi-stage instructions into text embeddings F_t and multi-view RGB images into 2D image embeddings F_v. The 2D embeddings are uplifted into 3D Gaussians to form 3D visual features f_v. Finally, the model localizes task-relevant object regions by computing the similarity between the affordance text embeddings $\hat{f}_t$ and uplifted 3D visual features f_v. The localized regions are passed through a grasp pose detection model to estimate the 6D grasp pose for robotic execution.

As summarized in Tab. 1, our approach resolves these limitations by uniquely combining a 3DGS-based representation, single-image supervision, and verb-level language grounding in 3D scene. Operating in a one-shot setting, our framework requires no reference images at test time. Instead, we leverage vision-language models to learn task-oriented affordances from sparse 2D annotations and directly align the visual feature representation with action verbs, eliminating the need for explicit part-level textual queries. Rather than relying on 2D planes, we lift these learned 2D affordance features into a view-consistent 3D Gaussian representation, entirely avoiding the viewpoint inconsistencies inherent in prior 2D methods. Ultimately, this unified formulation enables scalable 3D affordance inference derived purely from minimal 2D supervision, making it particularly suitable for long-horizon multi-stage robotic manipulation.

III. METHOD

In this section, we present **AffoRo-GS**, a one-shot 3D affordance learning framework that grounds natural language actions in 3D Gaussian scene representations. Given a single affordance-labeled image, our method learns action-centric affordance features in 2D and uplifts them into a unified 3D representation that can be reused across viewpoints and task stages. Our approach does not require 3D affordance annotations or explicit part labels. As illustrated in Fig. 2,

the pipeline consists of three stages: (i) uplifting affordance-aware 2D visual features into a 3D Gaussian representation, (ii) aligning these features with action-conditioned text embeddings, and (iii) localizing affordance-consistent regions to guide robotic grasp execution.

A. PROBLEM DEFINITION

We aim to enable a robot to execute high-level natural language instructions by inferring task-relevant affordances directly in 3D space. Given a language command and a single affordance-labeled image captured from an arbitrary viewpoint, our goal is to localize action-consistent regions within a 3D scene representation to guide robotic manipulation. To this end, we formulate affordance grounding as a multimodal alignment problem between visual features and language-conditioned action semantics as illustrated in Fig. 2. We employ a transformer decoder architecture, termed the affordance decoder, to fuse visual and textual information, enabling flexible affordance grounding without the need for part-level annotations.

B. PRELIMINARY: 3D GAUSSIAN SPLATTING

3DGS represents a scene as a set of differentiable 3D Gaussian primitives, enabling high-fidelity reconstruction together with real-time novel-view rendering. Each primitive G_i is parameterized by a 3D mean position $\mu_i \in \mathbb{R}^3$, a full covariance matrix $\Sigma_i \in \mathbb{R}^{3\times3}$, an opacity parameter $\alpha_i \in (0, 1)$, and view-dependent color attributes encoded using spherical harmonics (SH) coefficients.

Given a camera with viewing direction d, each 3D Gaussian is projected onto the image plane, resulting in a 2D Gaussian footprint. For a pixel p, let $\mathcal{S}_{d,p}$ denote the set of projected Gaussians that overlap p, sorted in ascending order of depth along the viewing ray. The rendered color $\hat{C}(p, d)$ is obtained via front-to-back alpha compositing:

$$\hat{C}(p, d) = \sum_{i\in\mathcal{S}_{d,p}} w_i(p, d)\, c_i(d), \tag{1}$$

where $c_i(d)$ denotes the view-dependent color of i-th Gaussian, and the compositing weight $w_i(p, d)$ is defined as:

$$w_i(p, d) = \alpha_i(p, d) \prod_{j\in\mathcal{S}_{d,p}, j<i} \big(1 - \alpha_j(p, d)\big). \tag{2}$$

Here, $\alpha_i(p, d)$ represents the opacity contribution of Gaussian i at pixel p for viewing direction d.

C. UPLIFT GAUSSIAN AFFORDANCE

Given a preconstructed 3DGS scene, our goal is to infer task-relevant and view-consistent 3D affordances by uplifting multi-view 2D visual features into the Gaussian-based 3D representation. To achieve this without introducing additional networks or modifying the underlying 3DGS formulation, we adapt the feature aggregation strategy from LUDVIG [16].

Specifically, let $F_v(d, p) \in \mathbb{R}^C$ denote a 2D visual feature extracted at pixel p from view d. The set of (view, pixel) pairs contributing to a specific Gaussian G_i is defined as:

$$S_i = \{(d, p) \mid i \in \mathcal{S}_{d,p}\}, \tag{3}$$

where $\mathcal{S}_{d,p}$ denotes the ordered set of Gaussians contributing to pixel p in view d. We denote the aggregated 3D visual features of the entire scene as $f_v \in \mathbb{R}^{N\times C}$, where N is the total number of Gaussians. The feature corresponding to the i-th Gaussian, denoted as $f_{v,i} \in \mathbb{R}^C$, is computed as:

$$f_{v,i} = \sum_{(d,p)\in S_i} \bar{w}_i(d, p)\, F_v(d, p), \tag{4}$$

with normalized weights defined as:

$$\bar{w}_i(d, p) = \frac{w_i(d, p)}{\sum_{(d,p)\in S_i} w_i(d, p)}, \tag{5}$$

where $w_i(d, p)$ is the rendering weight in Eq. 2.

This procedure lifts 2D visual features into a geometry-aware and view-consistent 3D affordance representation anchored at individual Gaussian primitives. Importantly, the uplifting process relies solely on the rendering weights of the preconstructed 3DGS scene and does not require training additional networks.

D. AFFORDANCE TRANSFORMER

The affordance transformer learns a modality-aligned embedding space in which task-relevant textual affordance descriptions and visual features are explicitly aligned. Our model consists of three components: (i) a part-aware visual encoder, (ii) a semantically rich text encoder, and (iii) a cross-attention transformer decoder that produces visually grounded affordance text embeddings.

1) AFFORDANCE VISUAL ENCODER

To accurately infer affordance visual features, it is crucial to utilize an encoder that generates part-aware representations. This necessitates a visual encoder capable of extracting fine-grained, part-aware representations. Accordingly, we employ DINOv2 [40], which is renowned for its ability to produce high-resolution features with strong part-level semantic correspondence. To effectively capture the affordances from multiple parts of objects, we aggregate hierarchical features from the final layers of the DINOv2 backbone. This fusion strategy provides a comprehensive representation that encodes both the overall object structures and the fine-grained details essential for robust affordance reasoning.

2) CLIP TEXT ENCODER

While the CLIP text encoder excels at visual-text alignment, it struggles to comprehend domain-specific vocabularies, such as affordances [8]. Naive fine-tuning is suboptimal, as it risks degrading CLIP's zero-shot generalization capabilities. Although methods like OOAL [8] attempt to mitigate this using CoOp-based prompt tuning [41], optimizing context vectors inherently biases the text embeddings toward the training distribution. To overcome this limitation, we adopt

an ensemble of manually designed prompts instead of learnable prompts. This strategy preserves a broad semantic coverage, avoiding the narrow bias of tuned embeddings. By delegating the cross-modal alignment to the affordance decoder, we allow the model to refine the broad text embeddings by conditioning them on visual features, rather than overfitting the embeddings themselves.

3) AFFORDANCE DECODER

To align the DINO visual features F_v with the affordance text embeddings F_t, we employ a cross-attention transformer decoder, where the text embedding F_t serves as the query, while the visual features F_v provide the corresponding keys and values. This attention process is guided by the visual backbone's *CLS* token, which provides a global summary of the visual input. Our transformer decoder finally outputs a visually-grounded text embedding $\hat{F}_t$, aligned with the specific visual affordances of the object.

Our transformer decoder is trained with F_v from one-shot affordance-labeled images and tested with 3D uplifted visual features f_v that are computed in Eq. 4. The lack of 3D ground truth data precludes the supervised training of the transformer decoder with 3D data. To adapt our transformer decoder for 3D affordance visual features f_v, generating the guidance token M_{cls} is problematic as it originates from a 2D-specific backbone. To resolve this, we introduce a single linear layer network ϕ_c' designed specifically to predict the M_{cls} token directly from any visual feature set, decoupling it from the 2D backbone. During training, we stochastically replace the backbone-derived token with the one inferred directly from the new network, with the replacement probability increasing to a maximum of 0.7.

$$M_{cls} = \begin{cases} \text{sigmoid}\left(\dfrac{\phi_c(L_{cls})K^T}{\sqrt{d_k}}\right) & \text{if } u \geq p \\ \text{sigmoid}\left(\dfrac{\phi_c'(F_v)K^T}{\sqrt{d_k}}\right) & \text{otherwise,} \end{cases} \tag{6}$$

where d_k is a scaling factor equal to the dimensions of the keys, $u \sim \mathcal{U}(0,1)$ and $p = \min(0.7, \frac{\texttt{epoch}}{\texttt{max\_epoch}})$. Consequently, when processing 3D features at inference time, ϕ_c' in Eq. 6 provides the necessary M_{cls} token, enabling our original transformer decoder to operate without modification.

$$Q = \phi_q(F_t), \quad K = \phi_k(f_v), \quad V = \phi_v(f_v) \tag{7}$$

$$M_{cls} = \text{sigmoid}\left(\frac{\phi_c'(f_v)K^T}{\sqrt{d_k}}\right) \tag{8}$$

$$\hat{f}_t = \text{softmax}\left(QK^T/\sqrt{d_k}\right) \cdot M_{cls}V + F_t. \tag{9}$$

E. AFFORDANCE LOCALIZATION AND ROBOTIC MANIPULATION

To localize the target affordances within the scene, we evaluate the similarity between the aggregated 3D visual features $f_v \in \mathbb{R}^{N\times C}$ and the text embeddings of the affordance queries. For a given stage in a multi-stage manipulation task, we define m text queries: one positive query, the target affordance specified by the instruction, and $m-1$ negative queries. These negative queries consist of a default 'background' class and, optionally, other irrelevant affordances present in the scene.

Let $\hat{f}_t \in \mathbb{R}^{m\times C}$ in Eq. 9 denote the stacked text embedding matrix for these queries. The raw similarity scores are computed simultaneously via matrix multiplication:

$$sim_{Aff} = f_v \cdot \hat{f}_t^{\mathsf{T}} \in \mathbb{R}^{N\times m}. \tag{10}$$

Following standard CLIP querying practices [14], [42], we process these raw scores sim_{Aff} computed in Eq. 10 to robustly suppress false positives. Detailed formulations of this scoring mechanism are provided in the Appendix A

3D Gaussians whose final affordance scores exceed a predefined threshold $\tau = 0.6$ are identified as the target affordance regions. To physically execute the robotic manipulation, we extract 6-DoF grasp poses directly from these localized Gaussians using the Grasp Pose Detection (GPD) algorithm [43], similar to the approach used in GraspSplats [14]. First, the point cloud used for GPD is generated by using the learned centers μ of the subset of 3D Gaussians as the point coordinates. Then, we densified the resulting point cloud for the target object region to reach approximately 60k points and processed it using the GPD algorithm to obtain a set of grasp poses. The grasp poses with their translation vector to be within a certain distance of the filtered Gaussian points are selected. Finally, the selected grasp poses are sorted by score, with the highest-scoring poses being prioritized for robotic manipulation.

IV. EXPERIMENTS

A. DATASET AND TASK SCENARIOS

Existing affordance datasets predominantly consist of internet-sourced images with single-view observations or decontextualized interactions. As a result, they lack the multi-view coverage and geometric consistency required to evaluate view-consistent 3D affordance inference in cluttered, task-oriented robotic environments. These limitations render them unsuitable for studying affordance localization in 3D Gaussian representations.

1) DATASET

To address this gap, we collect a real-world multi-view affordance dataset comprising 11 task-oriented scenes, encompassing over 19 distinct affordance types. To reflect a realistic low-supervision and one-shot learning scenario, only a single image per scene is annotated with task-oriented affordance labels. Affordance annotations are generated using SAM2 [44], and are used solely during training. This dataset design enables rigorous evaluation of a model's ability to generalize affordance predictions across unseen viewpoints and 3D geometry while operating under extreme annotation sparsity. Details about dataset collection and affordance annotations are described in the Appendix B. Beyond our custom dataset, we utilized five datasets from

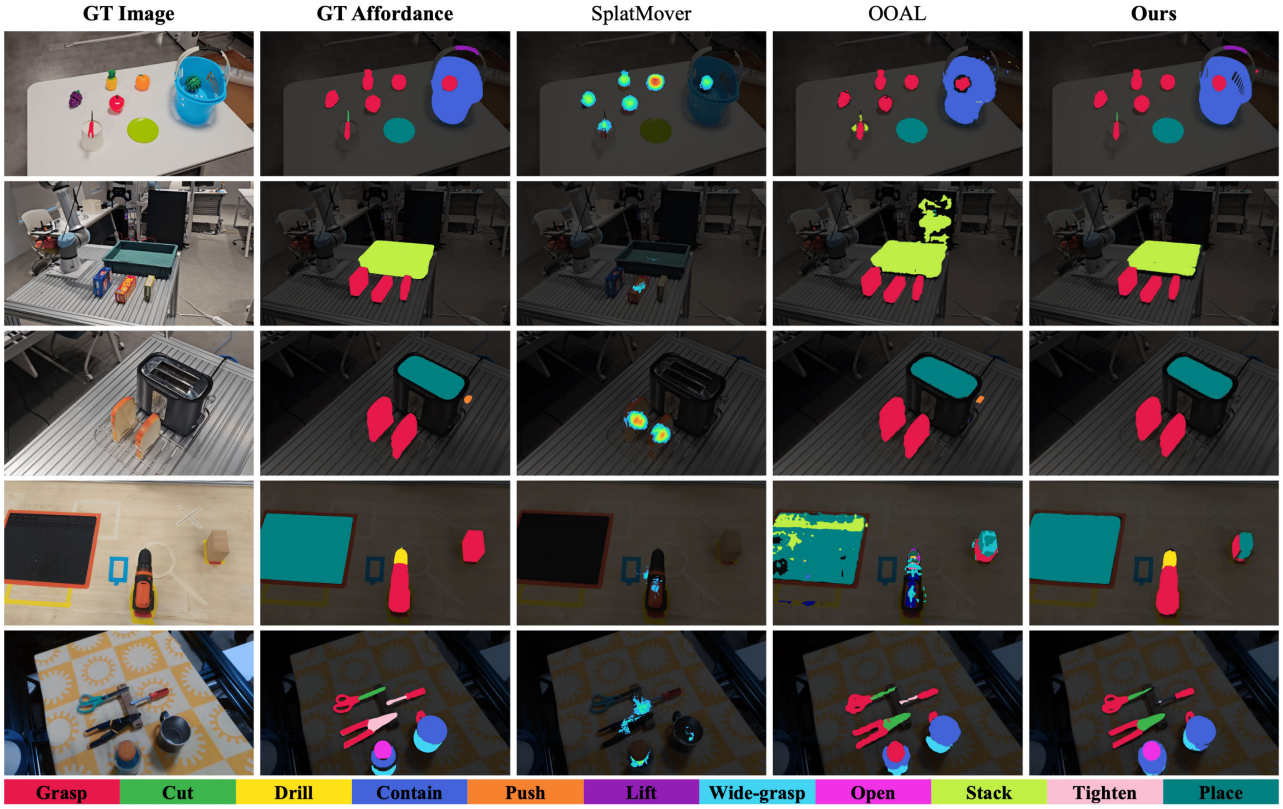

FIGURE 3. Qualitative comparison of multi-view task-oriented affordance inference. Ground-truth affordances are annotated using SAM2. Splat-MOVER relies on the Vision-Robotic Bridge (VRB) foundation model and produces action-agnostic interaction regions, while OOAL performs per-view inference and exhibits significant viewpoint inconsistency. Our method uplifts action-aligned visual features into a unified 3D Gaussian representation, resulting in stable and view-consistent affordance localization across diverse viewpoints.

TABLE 2. Quantitative comparison of task-oriented affordance prediction accuracy averaged across 16 tasks.

Affordance	Grasp	Cut	Drill	Contain	Push	Lift	Wide-grasp	Open	Stack	Tighten	Place
					SSIM ↑						
Splat-MOVER [18]	0.13 ± 0.04	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
OOAL [8]	0.69 ± 0.13	0.20 ± 0.09	0.10 ± 0.06	0.39 ± 0.06	$\mathbf{0.10} \pm 0.08$	0.22 ± 0.08	0.10 ± 0.01	0.25 ± 0.02	0.05 ± 0.01	$\mathbf{0.04} \pm 0.02$	0.43 ± 0.08
Ours	$\mathbf{0.74} \pm 0.10$	$\mathbf{0.35} \pm 0.05$	$\mathbf{0.29} \pm 0.09$	$\mathbf{0.74} \pm 0.07$	0.00 ± 0.00	$\mathbf{0.78} \pm 0.09$	$\mathbf{0.37} \pm 0.04$	$\mathbf{0.61} \pm 0.03$	$\mathbf{0.45} \pm 0.03$	0.01 ± 0.01	$\mathbf{0.54} \pm 0.08$
					mIoU ↑						
Splat-MOVER [18]	0.12 ± 0.04	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
OOAL [8]	0.64 ± 0.14	0.18 ± 0.10	0.08 ± 0.06	0.38 ± 0.06	$\mathbf{0.04} \pm 0.10$	0.19 ± 0.11	0.09 ± 0.01	0.24 ± 0.02	0.05 ± 0.01	$\mathbf{0.02} \pm 0.02$	0.39 ± 0.11
Ours	$\mathbf{0.70} \pm 0.11$	$\mathbf{0.34} \pm 0.05$	$\mathbf{0.28} \pm 0.09$	$\mathbf{0.71} \pm 0.08$	0.00 ± 0.00	$\mathbf{0.76} \pm 0.10$	$\mathbf{0.36} \pm 0.04$	$\mathbf{0.60} \pm 0.03$	$\mathbf{0.44} \pm 0.03$	0.01 ± 0.01	$\mathbf{0.53} \pm 0.09$

Splat-MOVER [18] and GraspSplats [14] to ensure a comprehensive evaluation across diverse object categories and environments.

2) TASK DESIGN

To evaluate task-oriented affordance reasoning under sequential decision-making settings, we design a set of interactive robotic manipulation tasks composed of multiple action stages. Each stage corresponds to a distinct action (*e.g.*, pick, place, insert) and requires grounding the action to a specific functional region of the target object. Each task is formalized as an ordered sequence of action–object pairs, as summarized in Table 4. This formulation enables fine-grained evaluation of whether a model can correctly localize affordances that are not only spatially precise but also semantically aligned with the intended action at each stage. In contrast to single-action affordance benchmarks, our task design explicitly tests the model's ability to disambiguate object regions whose functional relevance varies across different actions.

B. EXPERIMENTAL SETUP

Our affordance transformer is trained using one affordance-labeled image per scene, while the corresponding 3D Gaussian scene representations are optimized independently. This separation ensures that we can update each module separately when there are changes, enabling efficient updates, and also

ensures that other comparison methods in 3D affordance predictions operate on identical geometric reconstructions during inference. At test time, the predicted affordances are uplifted from 2D image space into the 3D Gaussian representation and subsequently used to guide grasp pose generation via the GPD framework [43]. Further implementation details about the model training and subsequent robotic manipulation are described in the Appendix C.

C. BASELINES

To evaluate our method's capabilities under practical conditions, we benchmark against two primary categories of baselines: affordance prediction models and a language-guided robotic manipulation framework. For multi-view consistency evaluation in task-oriented affordance prediction, we compare our method against Splat-MOVER [18], a zero-shot 3D task-agnostic affordance learning approach, and OOAL [8], a one-shot 2D task-oriented affordance model. To assess the accuracy of grasp pose localization via natural language instructions, we compare our framework against GraspSplats [14], which is a language-guided robotic manipulation framework that requires part-specific queries. Detailed rationales for baseline selection are provided in the Appendix D.

For a rigorous evaluation, we adapt GraspSplats, which typically requires explicit part-level queries to identify interaction regions, under two settings: with such queries (denoted as 'w/ obj.') and without them. These queries typically specify structural components, such as 'the handle of' or 'the knob of'. In contrast, our method relies solely on natural language action commands, requiring no explicit part-level supervision. Additionally, to ensure a fair comparison, we standardize the grasp generation module across all methods by employing Grasp Pose Detection (GPD) [43]. The detailed parameters and grasp filtering strategies for GPD are provided in the Appendix E.

D. EVALUATION METRICS

We evaluate our framework's performance at two distinct levels: 2D affordance prediction and robotic execution in 3D space. To assess task-oriented affordances in 2D, we quantify the prediction quality using two standard segmentation metrics: similarity (SIM) [34] and mean Intersection over Union (mIoU). To ensure robustness, the results for each affordance are averaged across 16 scenes, with each scene containing between 13 and 33 distinct test viewpoints.

For a fair comparison with 2D baselines, we follow the evaluation protocol established by OOAL [8]. Specifically, we render the 3D visual features $f_v \in \mathbb{R}^{N\times C}$ into a specific 2D viewpoint to obtain the 2D feature maps $F_v \in \mathbb{R}^{H\times W\times C}$. Given the text embeddings of the affordance queries $\hat{f}_t \in \mathbb{R}^{19\times C}$ (representing total 19 affordance categories including the background), the 2D raw similarity scores sim^{2D}_{Aff} are computed via matrix multiplication:

$$sim^{2D}_{Aff} = F_v \cdot \hat{f}_t^{\intercal} \in \mathbb{R}^{H\times W\times 19}. \tag{11}$$

Finally, the 2D affordances maps are inferred by applying the argmax operation over the channel dimension of the softmax probabilities derived from these raw similarity scores. For the baseline Splat-MOVER [18], the model inherently predicts task-agnostic affordance heatmaps. Empirically, we find that these heatmaps primarily highlight graspable areas. Therefore, to enable a fair 2D task-oriented comparison, we map the affordance predictions of Splat-MOVER directly to the 'grasp' category.

Manipulation performance is evaluated using a two-stage protocol. First, we measure grasp pose accuracy, defined as the ratio of predicted grasp poses that are correctly localized on the target object's functionally appropriate region. Second, for all poses deemed accurate, we evaluate physical grasp success based on real-world execution outcomes.

E. MULTI-VIEW TASK-ORIENTED AFFORDANCE PREDICTION RESULTS

We first evaluate the robustness of task-oriented affordance inference under multi-view observations, where models are trained exclusively on single-view annotations without multi-view or view-specific supervision. This setting reflects a realistic robotic manipulation scenario: an eye-in-hand camera observes objects from diverse viewpoints, yet annotated affordances are available for only a single reference image.

As qualitatively shown in Fig. 3, while baselines in multi-view task-oriented predictions struggle with generic hotspots and viewpoint inconsistencies, our approach maintains high stability by anchoring predictions to a unified 3D representation. In Tab. 2, our method demonstrates strong performance, particularly for fine-grained affordances, although specific challenges remain for the *push* and *tighten* actions.

For the *push* task, although the model correctly localizes the affordance region, it yields low-confidence scores that are heavily penalized under the strict 2D evaluation protocol. Nevertheless, as demonstrated in Fig. 4, our 3D localization framework effectively recovers these suppressed predictions to successfully infer accurate grasp poses. Similarly, the performance for *tighten* is hindered by visual ambiguities with the *cut* action. The 2D affordance model occasionally exhibits multi-view inconsistencies causing these predictions to be filtered out during the 3D fusion stage. Ultimately, these results highlight the strong regularizing effect of explicit 3D grounding, ensuring robust affordance localization even in cluttered, multi-view environments.

F. ROBOTIC MANIPULATION RESULTS GIVEN NATURAL LANGUAGE

We evaluate whether the inferred 3D affordances enable effective robotic manipulation when tasks are specified using natural language commands. As shown in Fig. 4, our method robustly localizes affordances from natural language, whereas baselines require explicit part specifications to function.

Tab. 3 reports the number of successes out of 10 independent trials using the top-ranked grasp. Our method

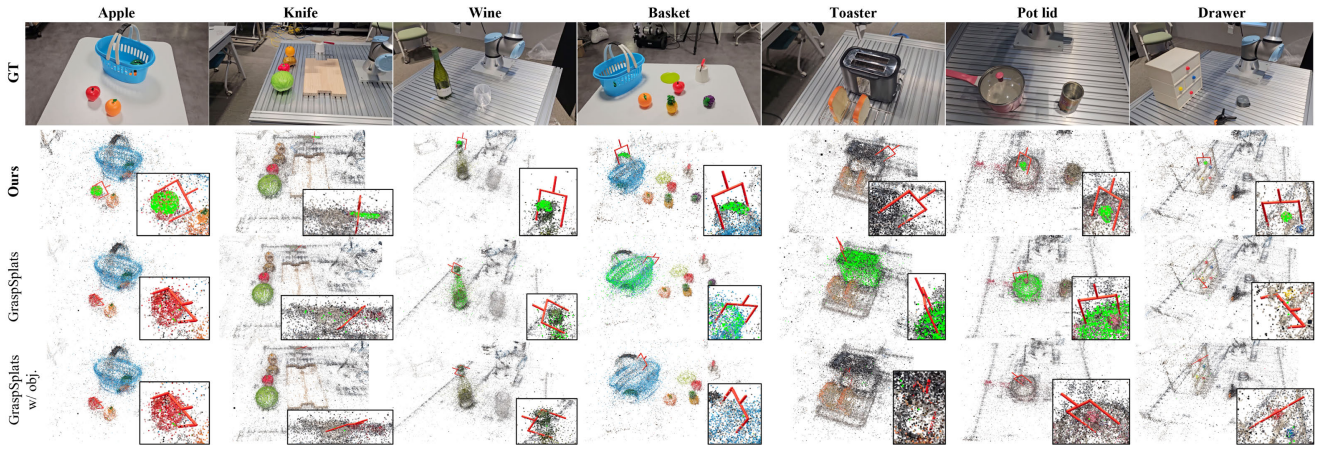

FIGURE 4. Comparison of grasp pose generation given natural language instructions. Instructions: "grasp the apple", "grasp the knife", "open the wine bottle", "pick up the basket", "press the toaster lever", "lift the lid", and "pull out the drawer". Our method identifies relevant object affordances (highlighted in green) and visualizes the top-ranked grasp pose based on the GPD score from the highlighted point cloud. Conversely, the baseline (GraspSplats w/ obj.)requires explicit part specifications in the instruction (*e.g.*, "handle" or "knob") to function.

TABLE 3. Evaluation of grasp pose generation and real-world execution. Aff. denotes the ratio of inferred poses that correctly cover the target part, and SR reports the physical execution success rate achieved using those valid grasps.

	Apple		Knife		Wine		Basket		Toaster		Pot lid		Drawer	
	Aff.	SR	Aff.	SR	Aff.	SR	Aff.	SR	Aff.	SR	Aff.	SR	Aff.	SR
GraspSplats [14]	8/10	5/8	3/10	3/3	2/10	2/2	0/10	0/0	2/10	2/2	0/10	0/0	2/10	0/2
GraspSplats w/ obj.	8/10	5/8	10/10	9/10	10/10	8/10	9/10	3/9	3/10	2/3	7/10	4/7	10/10	4/10
Ours	**9/10**	**6/9**	**10/10**	**8/10**	**10/10**	**9/10**	**10/10**	**5/10**	**10/10**	**4/10**	**10/10**	**9/10**	**10/10**	**6/10**

significantly outperforms baselines in both localization and execution. Notably, even with explicit part queries, GraspSplats struggles with small parts (*e.g.*, toaster lever).

Regarding the time efficiency in real robot manipulation, our affordance inference (approx. 1.1s) incurs only a marginal overhead compared to GraspSplats (0.8s). Given the improved robustness, we consider this overhead acceptable. The time required for final grasp pose generation remains consistent across all methods at approximately 0.5s.

G. RESULTS ON MULTI-STAGE MANIPULATION

Finally, we evaluate our method's performance on long-horizon, multi-stage manipulation tasks that require sequential reasoning over multiple action–object interactions. Specifically, we anchor the multi-stage execution trajectory to a canonical home pose [45] to isolate affordance accuracy from complex motion planning dependencies. Tab. 4 reports the task success rates across all scenes in our dataset for each manipulation scenario.

We define a multi-stage task as successful strictly if the robot correctly executes all intermediate stages to complete the final objective. Our method demonstrates promising performance across these challenging scenarios, validating that stable and view-consistent 3D affordance grounding effectively translates to successful grasp execution. Notably, most failure cases stem from physical execution issues, such as slippage or hardware limitations, rather than incorrect affordance localization. This observation underscores that our proposed affordance representation provides a robust foundation for complex, language-conditioned robotic manipulation in real-world environments.

TABLE 4. Summary of multi-stage manipulation tasks used for evaluation. We report the number of required action stages, the number of successful and failed executions, and the resulting task success rate tested with real-world robots.

Task	# of stage	Success	Fail	Rate
Put the tool in the drawer	3	2	3	40%
Put the apple in the basket	3	3	2	60%
Cut the apple with the knife	4	3	2	60%
Pour the sauce into the pot	4	1	3	25%
Toast two slices of bread with the toaster	5	1	3	25%
Stack three snack boxes in the container	6	2	2	50%
Fry the cabbage with the spatula	6	1	3	25%

H. ABLATION STUDY

In this section, we investigate the contribution of text feature ensembling and text–visual alignment to the framework's generalization capabilities. We compared our full model against five variants in Tab.5. We retrained the models on 7 scenes and evaluated them on 9 held-out scenes containing novel objects and challenging materials, such as transparent and reflective surfaces.

TABLE 5. Ablation study on model variations for Grasp and Place affordances (Mean ± Std). The results highlight the contribution of text feature ensembling and text–visual alignment through the decoder to overall affordance prediction performance.

Method	Grasp		Place	
	mIoU ↑	SIM ↑	mIoU ↑	SIM ↑
Ours	0.56 ± 0.11	0.60 ± 0.10	$\mathbf{0.30} \pm 0.06$	$\mathbf{0.31} \pm 0.06$
w/o ensemble	0.62 ± 0.12	0.66 ± 0.11	0.26 ± 0.08	0.27 ± 0.06
coop only	0.64 ± 0.12	0.69 ± 0.10	0.19 ± 0.04	0.19 ± 0.04
Ours (No Decode)	0.63 ± 0.12	$\mathbf{0.70} \pm 0.10$	0.22 ± 0.05	0.22 ± 0.05
w/o ensemble	0.60 ± 0.11	0.65 ± 0.10	0.27 ± 0.07	0.27 ± 0.06
coop only	$\mathbf{0.65} \pm 0.11$	$\mathbf{0.70} \pm 0.10$	0.19 ± 0.04	0.19 ± 0.04

To isolate the effect of prompt design, we evaluated two CLIP prompt strategies: removing the prompt ensemble (*w/o ensemble*) and using only CoOp prompts (*coop only*). In addition, we removed the affordance decoder (*No Decode*) to examine the role of the explicit text–visual alignment module.

Results indicate that while decoder-free variants achieve comparable or even higher performance on the *grasp* task, they struggle significantly on the more complex *place* task. This behavior arises because grasp affordances are largely determined by geometric cues such as curvature or edges, for which feature similarity alone provides sufficient discriminability. Meanwhile, explicit text–visual alignment is essential for capturing broader functional semantics that generalize across diverse object shapes. Furthermore, reducing prompt diversity (*w/o ensemble* or *coop only*) leads to noticeable performance drops on *place*, demonstrating that text feature ensembling stabilizes language-conditioned predictions. Overall, the decoder and prompt ensembling are critical for interpreting semantically complex affordances. This reflects a deliberate trade-off in our architecture: prioritizing generalization across diverse functional semantics over specialization in purely geometric tasks.

Finally, to verify that our method's robustness extends beyond specific backbone choices, we compared the original CLIP against MaskCLIP [46] and OpenCLIP [47]. Results on the novel object test set indicate negligible performance variance: the average mIoU for four major affordances remained consistent at 0.44 (CLIP), 0.45 (OpenCLIP), and 0.44 (MaskCLIP), with corresponding SIM scores of 0.46, 0.47, and 0.46. This suggests that our method is agnostic to the specific CLIP variant.

V. LIMITATIONS AND FUTURE WORK

While our framework demonstrates robust performance, it has three primary limitations that present clear avenues for future research. First, to ensure training stability and geometric consistency, we formulate our approach as one-shot 3D affordance learning rather than open-vocabulary manipulation. This design avoids the ambiguity of mapping diverse text embeddings to identical segmentation masks, prioritizing operational safety over exhaustive vocabulary coverage. Future work will investigate robust semantic alignment to enable true open-vocabulary generalization without sacrificing 3D stability. Second, our grasp selection relies solely on geometric cues and does not explicitly account for physical properties such as object mass distribution or friction. Incorporating physical reasoning and dynamic adaptation will be necessary to improve manipulation reliability. Finally, we assume a static 3D scene representation, which precludes modeling dynamic changes in the environment, such as object motion or scene reconfiguration during execution. Extending our method to dynamic representations (*e.g.*, 4D Gaussians) will be a critical step toward adaptive, interactive manipulation in unstructured real-world settings.

VI. CONCLUSION

We presented a one-shot 3D affordance learning framework that bridges high-level natural language instructions and robotic manipulation by grounding action verbs in a unified, view-consistent 3D affordance field. By uplifting sparse 2D affordance supervision into a 3D Gaussian Splatting representation of entire cluttered scenes, our method enables robust and action-aware affordance localization without relying on brittle part-level specifications or expensive 3D annotations. Through extensive real-world experiments on complex multi-stage manipulation tasks, we demonstrated that our approach significantly outperforms existing 2D and part-based methods in both affordance prediction and robotic execution. These results suggest that explicit 3D affordance grounding, when combined with language-conditioned reasoning, provides a strong foundation for intuitive and flexible human–robot interaction in unstructured environments.

APPENDIX A
3D AFFORDANCE SCORING

To robustly localize 3D affordances and suppress false positives, we employ a hard-negative-aware softmax ensemble rather than relying solely on raw similarity scores.

Let s^+ denote the raw similarity score of a 3D Gaussian for the positive query, and $\{s_j^-\}_{j=1}^{m-1}$ denote its similarity scores for the $m-1$ negative queries (including the 'background' class). Specifically, we pair the positive target score with each negative score to compute pairwise softmax probabilities. The probability of the positive class against the j-th negative query is computed as:

$$p_j^+ = \frac{\exp(\alpha \cdot s^+)}{\exp(\alpha \cdot s^+) + \exp(\alpha \cdot s_j^-)}, \tag{12}$$

where α is a temperature scaling factor (set to 10 in our implementation) that sharpens the probability distribution.

The final affordance score S_{final} for a 3D Gaussian is then determined by the minimum positive probability across all negative pairs (i.e., comparing against the hardest negative):

$$S_{final} = \min_{j \in \{1,\ldots,m-1\}} p_j^+. \tag{13}$$

This conservative scoring ensures that a Gaussian is highly activated only if its visual feature is distinctively aligned with the target verb and easily distinguishable from all other elements in the scene.

APPENDIX B
DETAILS IN DATASET COLLECTION AND AFFORDANCE LABELING

Each scene is captured from multiple viewpoints using a handheld Galaxy Z Flip 5 smartphone at 30 frames per second. For each scene, we record approximately 30 seconds of video while freely moving the camera around the workspace to ensure sufficient viewpoint diversity. Video frames are subsequently extracted at a rate of 10 fps, resulting in approximately 280–330 RGB images per scene. The extracted multi-view images are used to reconstruct a Gaussian representation for each scene.

In this work, we annotated 18 distinct affordances across various objects and their specific parts. The first four affordances listed below, excluding background, represent the primary categories utilized in our ablation studies for backbone selection.

For the one-shot affordance training, we annotated only a single frame for each of the 16 scenes using the SAM2 image predictor. Conversely, for the quantitative evaluation of 2D multi-view affordance inference, we employed the SAM2 video predictor. From these predictions, we manually filtered and selected frames with high-quality labels to serve as robust ground-truth data.

1) Background
2) Grasp: apple, orange, potato, red pepper, cabbage, pineapple, grape, bread, pincer, tape line, handle of knife, handle of pan, handle of spatula, snack boxes, block, handle of drill, handle of sponge, handle of spray bottle, handle of cup, handle of scissors, handle of screwdriver, handle of pliers, handle of pot, handle of fork
3) Contain: body of basket, sink, inside of cup, inside of bottle, body of pan, body of pot, sauce can
4) Place: dish, top of toaster, mat, cutting board, top of drawer, burner, top of stove, board
5) Lift: knob of pot lid, handle of basket
6) Cut: blade of knife, blade of scissors
7) Drill: chuck of drill
8) Push: lever of toaster
9) Wide-grasp: body of cup, body of bottle
10) Open: cap of bottle
11) Stack: container
12) Tighten: jaws of pliers, shaft of screwdriver
13) Pour: wine glass, inside of pot
14) Pull: knob of drawer
15) Scrub: body of sponge
16) Spray: trigger of spray bottle
17) Flip: head of spatula
18) Poke: head of fork
19) Turn: switch of stove

APPENDIX C
DETAILS IN MODEL TRAINING AND ROBOTIC MANIPULATION

Our affordance transformer is trained for 3k iterations using an SGD optimizer with a learning rate of 0.001, while the 3D Gaussian representation for each scene is trained for 30k iterations. Model training was conducted on an NVIDIA A100 GPU (40GB), while inference was performed on an NVIDIA GeForce RTX 4090 GPU.

We evaluate our method on real-world manipulation tasks using a UR5e manipulator equipped with a Robotiq 2F-85 gripper. Unlike traditional setups that rely on real-time camera feeds, our framework operates on offline-reconstructed 3D environments. Specifically, from the pre-collected dataset, we reconstruct the 3DGS scene, which is integrated into a simulation environment, where it is spatially aligned with the physical robot's base coordinate system. Once the visual and kinematic workspaces are registered, target grasp poses are inferred within the simulation and subsequently dispatched to the real-world UR5e for execution.

APPENDIX D
BASELINE SELECTION

We selected the specific baselines in our primary evaluation for two main reasons. First, we utilize one-shot baselines leveraging vision-language foundation models rather than traditional support-set-based methods such as AffCorrs [17], as this better reflects real-world generalization without reference images. We also include zero-shot baselines [18] to demonstrate the behavioral differences between zero-shot and one-shot generalizations, focusing on novel object generalization and multi-view robustness. Second, unlike prior 3D affordance methods confined to isolated point clouds or single-object representations [35], [37], our method reconstructs entire cluttered scenes using 3D Gaussians. Therefore, benchmarking against a scene-level, language-guided framework like GraspSplats ensures a fair assessment in similar complex environments.

Finally, certain 3D affordance methods fall outside the scope of our benchmark due to fundamentally different problem formulations. For instance, O^3Afford [9] focuses on inferring object-to-object affordances, requiring explicitly paired source and target point clouds. As the method is not language-conditioned and predicts geometric interaction points rather than scene-level semantic affordances, a fair comparison with ours is infeasible. Furthermore, SeqAffordSplat [13] relies heavily on affordance-annotated simulated 3D point clouds [48] without demonstrating transferability to real-world scenarios. Adapting such a data-intensive method to our setting would require an excessively high cost for real-world 3D affordance annotations, making it incompatible with our minimal-supervision paradigm.

APPENDIX E
GRASP POSE GENERATION

All evaluated baselines share the identical pre-trained GPD model weights from GraspSplats [14]. However, specific configuration parameters (*e.g.*, constraints on approaching vectors) were fine-tuned for our specific setup and applied uniformly across all methods to ensure a fair comparison.

Under these identical execution protocols, we introduced rigorous safety constraints. Specifically, we implemented a grasp filter that discards any candidate if the gripper's approach angle deviates by more than 1.0 rad from the vertical axis, or if the spatial distance to the target exceeds 0.02m.

Furthermore, because all methods generate grasp pose candidates directly from the target object's segmented point cloud, we are able to strictly isolate and execute the single top-ranked pose based solely on the predicted grasp scores.

ACKNOWLEDGMENT

During the manuscript preparation, the authors used Google's Gemini to improve the clarity and readability of the text. After using this tool, they carefully reviewed and edited the manuscript, and take full responsibility for the final content.

REFERENCES

[1] P. Ardón, É. Pairet, K. S. Lohan, S. Ramamoorthy, and R. P. A. Petrick, "Affordances in robotic tasks—A survey," 2020, *arXiv:2004.07400*.

[2] X. Yang, Z. Ji, J. Wu, and Y.-K. Lai, "Recent advances of deep robotic affordance learning: A reinforcement learning perspective," *IEEE Trans. Cognit. Develop. Syst.*, vol. 15, no. 3, pp. 1139–1149, Sep. 2023, doi: 10.1109/TCDS.2023.3277288.

[3] J. J. Gibson, *The Ecological Approach to Visual Perception: Classic Edition*. New York, NY, USA: Psychology Press, Nov. 2014, doi: 10.4324/9781315740218.

[4] F.-J. Chu, R. Xu, and P. A. Vela, "Learning affordance segmentation for real-world robotic manipulation via synthetic images," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1140–1147, Apr. 2019, doi: 10.1109/LRA.2019.2894439.

[5] S. Bahl, R. Mendonca, L. Chen, U. Jain, and D. Pathak, "Affordances from human videos as a versatile representation for robotics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 1–13, doi: 10.1109/CVPR52729.2023.01324.

[6] Y. Li, N. Zhao, J. Xiao, C. Feng, X. Wang, and T.-S. Chua, "LASO: Language-guided affordance segmentation on 3D object," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 14251–14260, doi: 10.1109/CVPR52733.2024.01351.

[7] M. Chu, X. Zhang, Z. Zheng, and T.-S. Chua, "3D-TAFS: A training-free framework for 3D affordance segmentation," 2024, *arXiv:2409.10078*.

[8] G. Li, D. Sun, L. Sevilla-Lara, and V. Jampani, "One-shot open affordance learning with foundation models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 3086–3096, doi: 10.1109/CVPR52733.2024.00298.

[9] T. Tian, X. Kang, and Y.-L. Kuo, "O^{3}Afford: One-shot 3D object-to-object affordance grounding for generalizable robotic manipulation," in *Proc. 9th Conf. Robot Learn.* (Proceedings of Machine Learning Research), vol. 305. PMLR, Sep. 2025, pp. 1541–1561. [Online].Available: https://proceedings.mlr.press/v305/tian25b.html

[10] G. Li, N. Tsagkas, J. Song, R. Mon-Williams, S. Vijayakumar, K. Shao, and L. Sevilla-Lara, "Learning precise affordances from egocentric videos for robotic manipulation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2024, pp. 10581–10591.

[11] B. Kerbl, G. Kopanas, T. Leimkuehler, and G. Drettakis, "3D Gaussian splatting for real-time radiance field rendering," *ACM Trans. Graph.*, vol. 42, no. 4, pp. 1–14, Jul. 2023, doi: 10.1145/3592433.

[12] Z. Wei, J. Lin, Y. Liu, W. Chen, J. Luo, G. Li, and L. Lin, "3DAffordSplat: Efficient affordance reasoning with 3D Gaussians," in *Proc. 33rd ACM Int. Conf. Multimedia*, Oct. 2025, pp. 2821–2830, doi: 10.1145/3746027.3754778.

[13] D. Li, J. Feng, J. Chen, W. Dong, G. Li, Y. Zheng, M. Feng, and G. Shi, "SeqAffordSplat: Scene-level sequential affordance reasoning on 3D Gaussian splatting," 2025, *arXiv:2507.23772*.

[14] M. Ji, R.-Z. Qiu, X. Zou, and X. Wang, "GraspSplats: Efficient manipulation with 3D feature splatting," in *Proc. 8th Annu. Conf. Robot Learn.* (Proceedings of Machine Learning Research), vol. 270. PMLR, Nov. 2024, pp. 1443–1460. [Online]. Available: https://proceedings.mlr.press/v270/ji25a.html

[15] D. Wu, Y. Fu, S. Huang, Y. Liu, F. Jia, N. Liu, C. Dai, T. Wang, R. M. Anwer, F. S. Khan, and J. Shen, "RAGNet: Large-scale reasoning-based affordance segmentation benchmark towards general grasping," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2025, pp. 11980–11990.

[16] J. Marrie, R. Ménégaux, M. Arbel, D. Larlus, and J. Mairal, "LUDVIG: Learning-free uplifting of 2D visual features to Gaussian splatting scenes," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct.2025, pp. 7440–7450. [Online]. Available: https://doi.org/10.48550/arXiv.2410.14462

[17] D. Hadjivelichkov, S. Zwane, M. P. Deisenroth, L. Agapito, and D. Kanoulas, "One-shot transfer of affordance regions? AffCorrs!" in *Proc. 6th Conf. Robot Learn. (CoRL)*, 2022, pp. 550–560.

[18] O. Shorinwa, J. Tucker, A. Smith, A. Swann, T. Chen, R. Firoozi, M. Kennedy, and M. Schwager, "Splat-MOVER: Multi-stage, open-vocabulary robotic manipulation via editable Gaussian splatting," in *Proc. 8th Annu. Conf. Robot Learn.* (Proceedings of Machine Learning Research), vol. 270. PMLR, Nov. 2024, pp. 4748–4770. [Online]. Available: https://proceedings.mlr.press/v270/shorinwa25a.html

[19] Y. Wang, R. Wu, K. Mo, J. Ke, Q. Fan, L. J. Guibas, and H. Dong, *AdaAfford: Learning to Adapt Manipulation Affordance for 3D Articulated Objects via Few-Shot Interactions*. Cham, Switzerland: Springer, 2022, pp. 90–107. [Online]. Available: http://dx.doi.org/10.1007/978-3-031-19818-2_6

[20] S. Morin, K. Gupta, M. Sandhu, C. Gauthier, F. Argenziano, K. Ellis, and L. Paull, "Agentic scene policies: Unifying space, semantics, and affordances for robot action," 2025, *arXiv:2509.19571*.

[21] W. Huang, C. Wang, R. Zhang, Y. Li, J. Wu, and L. Fei-Fei, "VoxPoser: Composable 3D value maps for robotic manipulation with language models," in *Proc. 7th Conf. Robot Learn.* (Proceedings of Machine Learning Research), vol. 229. PMLR, Nov. 2023, pp. 540–562. [Online]. Available: https://proceedings.mlr.press/v229/huang23b.html

[22] P. Ardon, E. Pairet, R. P. A. Petrick, S. Ramamoorthy, and K. S. Lohan, "Learning grasp affordance reasoning through semantic relations," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 4571–4578, Oct. 2019, doi: 10.1109/LRA.2019.2933815.

[23] Y. Ju, K. Hu, G. Zhang, G. Zhang, M. Jiang, and H. Xu, *Robo-ABC: Affordance Generalization Beyond Categories via Semantic Correspondence for Robot Manipulation*. Springer, Nov. 2024, pp. 222–239. [Online]. Available: http://dx.doi.org/10.1007/978-3-031-72940-9_13

[24] Y. Tang, W. Huang, Y. Wang, C. Li, R. Yuan, R. Zhang, J. Wu, and L. Fei-Fei, "UAD: Unsupervised affordance distillation for generalization in robotic manipulation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2025, pp. 3822–3831, doi: 10.1109/ICRA55743.2025.11128868.

[25] J. Li, Y. Zhu, Z. Tang, J. Wen, M. Zhu, X. Liu, C. Li, R. Cheng, Y. Peng, Y. Peng, and F. Feng, "Coa-vla: Improving vision-language-action models via visual-text chain-of-affordance," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2025, pp. 9759–9769. [Online]. Available: https://doi.org/10.48550/arXiv.2412.20451

[26] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763, doi: 10.48550/arxiv.2103.00020.

[27] Y.-L. Wei, M. Lin, Y. Lin, J.-J. Jiang, X.-M. Wu, L.-A. Zeng, and W.-S. Zheng, "Afforddexgrasp: Open-set language-guided dexterous grasp with generalizable-instructive affordance," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Jun. 2025, pp. 11818–11828.

[28] Y. Zheng, X. Chen, Y. Zheng, S. Gu, R. Yang, B. Jin, P. Li, C. Zhong, Z. Wang, L. Liu, C. Yang, D. Wang, Z. Chen, X. Long, and M. Wang, "GaussianGrasper: 3D language Gaussian splatting for open-vocabulary robotic grasping," *IEEE Robot. Autom. Lett.*, vol. 9, no. 9, pp. 7827–7834, Sep. 2024, doi: 10.1109/LRA.2024.3432348.

[29] A. Rashid, S. Sharma, C. M. Kim, J. Kerr, L. R. Chen, A. Kanazawa, and K. Goldberg, "Language embedded radiance fields for zero-shot task-oriented grasping," in *Proc. 7th Conf. Robot Learn.*, Nov. 2023, pp. 178–200. [Online]. Available: https://proceedings.mlr.press/v229/rashid23a.html

[30] C. Yu, H. Wang, Y. Shi, H. Luo, S. Yang, J. Yu, and J. Wang, "SeqAfford: Sequential 3D affordance reasoning via multimodal large language model," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2025, pp. 1691–1701, doi: 10.1109/CVPR52734.2025.00165.
[31] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 77–85, doi: 10.1109/CVPR.2017.16.
[32] X. Gao, P. Zhang, D. Qu, D. Wang, Z. Wang, Y. Ding, and B. Zhao, "Learning 2D invariant affordance knowledge for 3D affordance grounding," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2025, vol. 39, no. 3, pp. 3095–3103, doi: 10.1609/aaai.v39i3.32318.
[33] H. Lian, L. Meng, Y. Qilang, Z. Yu, D. Xiang, and D. Gangyi, "Task-aware 3D affordance segmentation via 2D guidance and geometric refinement," in *Proc. AAAI Conf. Artif. Intell.*, Mar. 2025, pp. 4654–4662.
[34] T. Ma, Z. Wang, J. Zhou, M. Wang, and J. Liang, "GLOVER: Generalizable open-vocabulary affordance reasoning for task-oriented grasping," 2024, *arXiv:2411.12286*.
[35] Y. Yang, W. Zhai, H. Luo, Y. Cao, J. Luo, and Z.-J. Zha, "Grounding 3D object affordance from 2D interactions in images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 10871–10881, doi: 10.1109/ICCV51070.2023.01001.
[36] J. Wang and D. Luo, "GauTOAO: Gaussian-based task-oriented affordance of objects," 2024, *arXiv:2409.11941*.
[37] D. Lu, L. Kong, T. Huang, and G. H. Lee, "GEAL: Generalizable 3D affordance learning with cross-modal consistency," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2025, pp. 1680–1690, doi: 10.1109/CVPR52734.2025.00164.
[38] V. Holomjova, A. J. Starkey, B. Yun, and P. Meißner, "One-shot learning for task-oriented grasping," *IEEE Robot. Autom. Lett.*, vol. 8, no. 12, pp. 8232–8238, Dec. 2023, doi: 10.1109/LRA.2023.3326001.
[39] W. Zhai, H. Luo, J. Zhang, Y. Cao, and D. Tao, "One-shot object affordance detection in the wild," *Int. J. Comput. Vis.*, vol. 130, no. 10, pp. 2472–2500, Aug. 2022, doi: 10.1007/s11263-022-01642-4.
[40] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. HAZIZA, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, "DINOv2: Learning robust visual features without supervision," *Trans. Mach. Learn. Res.*, Jan. 2024. [Online]. Available: https://openreview.net/forum?id=a68SUt6zFt
[41] K. Zhou, J. Yang, C. C. Loy, and Z. Liu, "Learning to prompt for vision-language models," *Int. J. Comput. Vis.*, vol. 130, no. 9, pp. 2337–2348, Jul. 2022, doi: 10.1007/s11263-022-01653-1.
[42] W. Shen, G. Yang, A. C. L. Yu, J. Wong, L. P. Kaelbling, and P. Isola, "Distilled feature fields enable few-shot language-guided manipulation," in *Proc. 7th Annu. Conf. Robot Learn.* (Proceedings of Machine Learning Research), vol. 229. PMLR, Nov. 2023, pp. 405–424. [Online]. Available: https://proceedings.mlr.press/v229/shen23a.html
[43] A. Ten Pas, M. Gualtieri, K. Saenko, and R. Platt, "Grasp pose detection in point clouds," *Int. J. Robot. Res.*, vol. 36, nos. 13–14, pp. 1455–1473, Oct. 2017, doi: 10.1177/0278364917735594.
[44] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. K. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer, "SAM 2: Segment anything in images and videos," in *Proc. 13th Int. Conf. Learn. Represent.*, vol. 2025, 2025, pp. 28085–28128.
[45] R. Tedrake, *Robotic Manipulation*. Cambridge, MA, USA: MIT Press, 2024. [Online]. Available: http://manipulation.mit.edu
[46] C. Zhou, C. C. Loy, and B. Dai, *Extract Free Dense Labels From CLIP*. Cham, Switzerland: Springer, 2022, pp. 696–712. [Online]. Available: http://dx.doi.org/10.1007/978-3-031-19815-1_40
[47] M. Cherti, R. Beaumont, R. Wightman, M. Wortsman, G. Ilharco, C. Gordon, C. Schuhmann, L. Schmidt, and J. Jitsev, "Reproducible scaling laws for contrastive language-image learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 2818–2829.
[48] S. Deng, X. Xu, C. Wu, K. Chen, and K. Jia, "3D AffordanceNet: A benchmark for visual object affordance understanding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 1778–1787, doi: 10.1109/CVPR46437.2021.00182.

HYUNSEO KIM received the B.S. degree in bioscience from Seoul National University (SNU), Republic of Korea, in 2019, where she is currently pursuing the Ph.D. degree with the Interdisciplinary Program in Neuroscience. Her current research interests include 3-D reconstruction, robot vision, and robot arm manipulation.

YEON-JI SONG received the B.Eng. degree in electronic and computer engineering from The Hong Kong University of Science and Technology (HKUST). She is currently pursuing the Ph.D. degree with the Interdisciplinary Program in Neuroscience, Seoul National University (SNU). Her research interests include neurosymbolic visual generation via physical concept grounding, dynamic scene understanding for novel view synthesis and reconstruction from blurry monocular inputs, object-centric learning modeling static appearance and dynamic motion, and embodied AI leveraging learned representations for real-world robotics.

MINSU LEE received the B.S. degree in mathematics and the M.S. and Ph.D. degrees in computer science and engineering from Ewha Womans University, South Korea. She is an Assistant Professor at Sungshin Women's University. Previously, she was a Research Professor at the AI Institute of Seoul National University and Ewha Womans University and was a Visiting Scholar at Indiana University, USA. Her research interests focus on bio-inspired active learning, machine learning techniques for video understanding, and real-world reinforcement learning applications.

BYOUNG-TAK ZHANG received the B.S. and M.S. degrees in computer science and engineering from Seoul National University (SNU), South Korea, in 1986 and 1988, respectively, and the Ph.D. degree in computer science from the University of Bonn, Germany, in 1992. He is a POSCO Chair Professor of computer science and engineering at SNU and the Director of Artificial Intelligence Institute, SNU. He was the President of Korean Society for Artificial Intelligence, from 2010 to 2013, and Korean Society for Cognitive Science, from 2016 to 2017.