Bridging 2D Vision Language Models to 3D World via Feature Distillation

Yang Fu[1*] Sifei Liu[2] Hongxu Yin[2] Benjamin Eckart[2] Jan Kautz[2]
Xiaolong Wang[1] Arash Vahdat[2] Chao Liu[2]
[1]UC San Deigo [2]NVIDIA

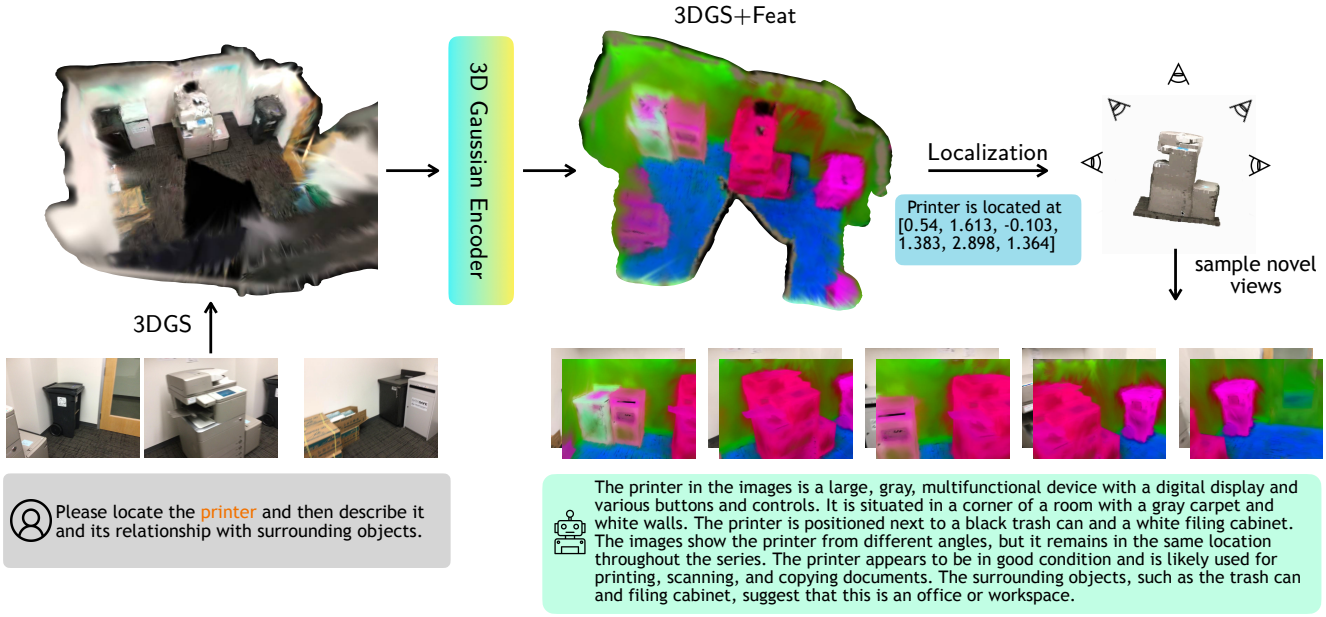

Figure 1. In this paper, we propose a generalist pipeline leveraging 3D Gaussian Splatting as an intermediate representation to bridge the gap between 2D Vision Language Models (VLMs) and the 3D world for 3D visual understanding. Starting from multi-view images, we first reconstruct the 3D Gaussian Splatting (3DGS) representation (details omitted for simplicity). Our learned 3D Gaussian encoder then generates language-aligned per-Gaussian features from the reconstructed 3D Gaussians. With these feature-enriched 3D Gaussians, regions of interest can be precisely located in 3D space. Furthermore, corresponding feature maps can be efficiently rendered from nearby novel viewpoints and utilized by 2D VLMs, enabling effective 3D scene understanding and reasoning without additional 3D training data.

Abstract

Recent advancements in Vision Language Models (VLMs) have demonstrated remarkable proficiency in 2D visual understanding tasks. However, extending these models to comprehend 3D environments remains a significant challenge, primarily due to their training on predominantly 2D data. To address this limitation, we propose bridging the gap between the inherent 2D representations of VLMs and 3D scenes using 3D Gaussian Splatting (3DGS)—an efficient and interpretable representation for 3D reconstruction and rendering. Specifically, we introduce a feed-forward framework, termed SplatDistill, which processes 3D Gaussian parameters to generate 3D visual features by distilling knowledge from 2D vision foundation models, guided by a rendering-and-comparison mechanism. The learned 3D Gaussian features enable zero-shot semantic understanding and support a range of 3D-based question answering tasks by seamlessly integrating with pretrained 2D vision-language models.

1. Introduction

Recent advances in Vision-Language Models (VLMs) [2, 7, 16, 48, 49, 67] have significantly improved their capabilities in a wide range of image and video understanding and reasoning tasks. Despite these achievements, current 2D VLMs face substantial limitations in perceiving and reasoning about the 3D world. This constraint stems from their fundamental architecture, which inherently utilizes 2D visual features and relies exclusively on training with images and video data.

Some recent studies attempt to develop the 3D VLMs

*This work was done while Yang Fu was an intern at NVIDIA.

by integrating 3D encoders to extract features from point clouds as inputs for VLMs. These 3D encoders are typically trained from scratch on 3D point cloud datasets with ground-truth question-answering pairs. However, due to the scarcity of large-scale 3D datasets, the 3D encoders are often less powerful and robust compared to 2D visual encoders, such as CLIP [62], DINOv2 [58], and RADIO [63], which are trained on billions of 2D images. To leverage these powerful 2D foundation models, several approaches [22, 24, 35] utilize 2D segmentation and CLIP features to obtain pixel-aligned semantic features and construct object-level 3D scene graphs for various 3D understanding tasks. Nevertheless, these approaches suffer from computationally intensive per-scene processing and complex implementation pipelines. In contrast, some efforts [83, 88] have focused on building 3D VLMs on top of powerful 2D VLMs by incorporating 3D positional information from multi-view RGB-D inputs. While these approaches achieve impressive performance in 3D scene perception and reasoning, they still rely on fine-tuning with multi-view datasets that include ground-truth depth and question-answering pairs. This raises a compelling question: can we enable 2D VLMs to directly perform 3D understanding and reasoning tasks without training on specialized 3D data?

To fully leverage the strengths of 2D VLMs, it is essential to bridge the fundamental gap between the 2D training data and 3D inference environments. In this paper, we propose utilizing 3D Gaussian Splatting (3DGS) [40, 41] as an intermediate representation to connect this gap. 3DGS provides an efficient and interpretable 3D representation with real-time rendering capabilities, making it a strong candidate for seamlessly connecting 2D images to 3D environments. Building on 3D Gaussian as the scene representation, we aim to map its parameters to a feature space that is well-aligned with several 2D foundation models, enabling compatibility with pretrained VLMs. Recent studies [65, 73, 76, 86] have proposed optimizing an additional attribute as the visual feature alongside 3DGS optimization. However, these per-scene optimization approaches suffer from lengthy processing times and poor generalization to new scenes, limiting their practicality in diverse and complex environments. To address this challenge, we propose a feed-forward framework, *SplatDistill*, to distill the rich language-aligned knowledge from 2D foundation models into a set of 3D Gaussians. Specifically, we introduce a transformer-based network that takes 3D Gaussians, along with their attributes, as inputs to produce high-dimensional features for each Gaussian. The proposed Gaussian transformer is trained using a rendering-and-comparison mechanism: the input Gaussians with extracted features are splatted into 2D space to generate a 2D feature map, which is then compared with the 2D feature map generated by the pretrained 2D foundation model. Therefore, the learned 3D Gaussian features are well-aligned with 2D foundation models and can be directly applied to various downstream tasks, such as 3D segmentation and localization, without requiring task-specific fine-tuning. During inference, these feature-augmented 3D Gaussians are rendered into multiple views, and the resulting feature maps, combined with RGB images, are used as input to pretrained 2D VLMs for 3D visual question answering, as illustrated in Fig. 1.

To validate the effectiveness of our approach, we conduct extensive experiments on a wide range of 3D tasks and benchmarks [5, 9, 17, 52, 64] including open-vocabulary scene understanding, 3D visual question answering, dense captioning, and 3D visual grounding. As the Gaussian features are well-aligned with the 2D features through the distillation process, our approach demonstrates improved performance on zero-shot open-vocabulary scene understanding over existing methods that require fine-tuning on each individual scene. Moreover, by connecting 2D VLMs to the 3D Gaussians, our approach largely enhances the 3D spatial understanding and reasoning capabilities of the existing 2D LLMs without fine-tuning on 3D QA ground-truth pairs.

2. Related Work

3D Feature Distillation from 2D Foundation Models. Inspired by the advance in open-vocabulary representation learning for 2D foundation models [4, 46, 48, 50, 51, 59, 62, 84], recent works on open-vocabulary 3D scene understanding [11, 19, 20, 31, 34, 37, 55, 60, 72, 91] focus on distilling 3D features from 2D foundation models so that the training process is not hindered by the lack of high-quality 3D annotations. While these methods have achieved comparable performance to models trained with 3D annotations on 3D segmentation [37], object detection [20, 72], salient map prediction based on text queries [34], there is still a gap in the abilities of 2D foundation models for more fine-grained understanding tasks such as VQA, dense captioning, affordance understanding and reasoning due to lack of 3D training data. To tackle this limitation, task specific 3D datasets with high-quality annotations are proposed recently [18, 36, 39].

Another line of works [42, 44, 61, 65, 73, 76, 86, 93] is driven by the advance of 3D scene representations [40, 41, 54]. Given a set of posed images representing a single scene, these methods optimize the 3D features along with the 3D scene representation, minimizing both the reconstruction loss and the task-specific feature loss on the 2D images, where the features are extracted from pretrained 2D foundation models. Since the 3D features are represented in the same way (i.e., 3D Gaussian or neural field) as the geometric scene representation, high-quality 2D feature maps can be rendered and fed as the input to 2D VLMs. The downstream scene understanding tasks can benefit from the strong capability of 2D VLMs [42]; the optimized 3D features can be used in turn to improve the 3D awareness and consistency

of the 2D model [76]. However, per-scene optimization is time-consuming and thus limits both the scalability and the generalization ability of these methods. Our method aims to address this limitation by distilling knowledge from 2D foundation models into a 3D encoder such that at test time 3D features can be extracted in a feed-forward manner and generalize to unseen scenes.

3D Scene Understanding via VLMs. Recent advancements in Vision Language Models (VLMs) have extended their application toward 3D scene understanding, leveraging rich semantic information from 2D representations for 3D tasks. This progress has substantially benefited 3D visual grounding [3, 8, 9, 15, 25, 32], captioning [10, 14, 15, 38, 68], and visual question answering [5, 12, 27]. Initial efforts, such as 3DJCG [8] and D3Net [10], unify multiple 3D scene tasks but rely on task-specific heads that limit adaptability. More generic strategies for integrating 3D data into VLMs have included using point cloud encoders for scene-level features (LL3DA [12]) and segmenting scenes into objects with object-level encoders (LEO [30], Chat3D [69]), but they struggle with capturing complex spatial relationships. Multi-view image methods like 3D-LLM [26] and Scene-LLM [22] provide richer integration but are resource-intensive and often project 3D features into pre-trained 2D VLMs, limiting full 3D modeling. To resolve these challenges, LLaVA-3D injects 3D spatial context into 2D VLMs using 3D position embeddings for enhanced 3D scene comprehension, while ConceptGraph [24] uses scene graphs to bypass direct 3D integration but struggles with spatial precision due to LLM limitations with coordinate data [53]. In contrast, our proposed SplatDistill offers a zero-shot approach to enhance 2D VLMs for 3D scene understanding by integrating accurate 3DGS geometry without fine-tuning, preserving the model's generalizability and enabling efficient 3D reasoning.

3. Method

Given a set of 3DGS reconstructed from a sequence of images along with camera intrinsics and extrinsics, our goal is to learn a distillation network that enables various 3D scene understanding and reasoning tasks. We detail our method in the following sections, starting with a brief review of 3DGS reconstruction in Sec. 3.1. Then, we propose to aggregate the pixel features from multiple views onto every Gaussian via multi-view fusion, in Sec. 3.2. Next, we introduce a feature distillation framework with a Gaussian encoder using 3D Gaussians as inputs, in Sec. 3.3.

3.1. 3D Gaussian Reconstruction

Preliminaries. The first step of our approach is to reconstruct the indoor scenes via 3D Gaussian Splatting [40]. Given a sequence of RGB images with camera intrinsics and extrinsics, we represent a scene with a set of 3D Gaussians as

$$\mathcal{G} = \{(\boldsymbol{\mu}, \mathrm{SH}, \boldsymbol{r}, \boldsymbol{s}, \alpha)_i\}_{i=1:M} \tag{1}$$

where (a) $\boldsymbol{\mu} \in \mathbb{R}^3$ is the 3D mean of the Gaussian, (b) $\mathrm{SH} \in \mathbb{R}^{(k+1)^2 \times 3}$ are the spherical harmonics (SH) coefficients that represent the Gaussian color, (c) $\boldsymbol{r} \in \mathbb{R}^4$ is its quaternion rotation factor, (d) $\boldsymbol{s} \in \mathbb{R}^3$ is the Gaussian scale and (e) $\alpha \in \mathbb{R}$ is the Gaussian opacity. Then, the covariance matrix $\boldsymbol{\Sigma}$ describes an ellipsoid configured by a scaling matrix $\boldsymbol{S} = \mathrm{diag}(\boldsymbol{s})$ and rotation matrix $\boldsymbol{R} = \mathrm{q2R}(\boldsymbol{r})$, where $\mathrm{q2R}(\cdot)$ is the formula for constructing a rotation matrix from a quaternion. Then, the covariance matrix can be computed as $\boldsymbol{\Sigma} = \boldsymbol{RSS}^\top \boldsymbol{R}^\top$. The rendered color can be formulated as the alpha-blending of N ordered points that overlap the pixel as follows,

$$\boldsymbol{c}_{\text{pix}} = \sum_i^N \mathrm{SH}_i \alpha_i \prod_j^{i-1} (1 - \alpha_j) \tag{2}$$

where α_i represents the density of this Gaussian computed from the per-Gaussian opacity weighted by the Gaussian covariance Σ, which we ignore in Eq. 2 for simplicity.

Training Data Setup. To create large amounts of 3DGS reconstruction examples for training the distillation network, we modify the original initialization strategy for better efficiency. In particular, instead of using SfM points, we initialize the Gaussian positions from the ground-truth scene mesh vertices. To reduce the number of initialized Gaussians, we perform the grid subsampling with a voxel size of 2cm on the original mesh. Finally, we remove the Gaussian growing step while only pruning Gaussians during optimization.

3.2. Multi-view Feature Fusion

For a given set of 3D Gaussians, our next step is to extract per-Gaussian features. As each scene can contain millions of Gaussians and feature maps from 2D foundation models are typically high-dimensional, directly optimizing 3DGS with high-dimensional per-Gaussian features can be time-consuming and computationally intensive. Therefore, we propose a multi-view feature fusion strategy to lift 2D feature maps into the 3D Gaussian space.

2D Feature Extraction. Previous works [60, 77] typically extract features from task-specific models - for instance, OpenScene utilizes features from a 2D segmentation model specifically for open-vocabulary scene understanding. This approach necessitates multiple model distillations to accomplish different tasks. To address this limitation, we propose to extract backbone features from the RADIO [63] model. Unlike task-specific features, RADIO's backbone features can be projected into various target embedding spaces (*i.e.*, DINO [58], CLIP [62], SigLIP [79], and SAM [43]) using pretrained lightweight adapters, significantly enhancing our approach's flexibility.

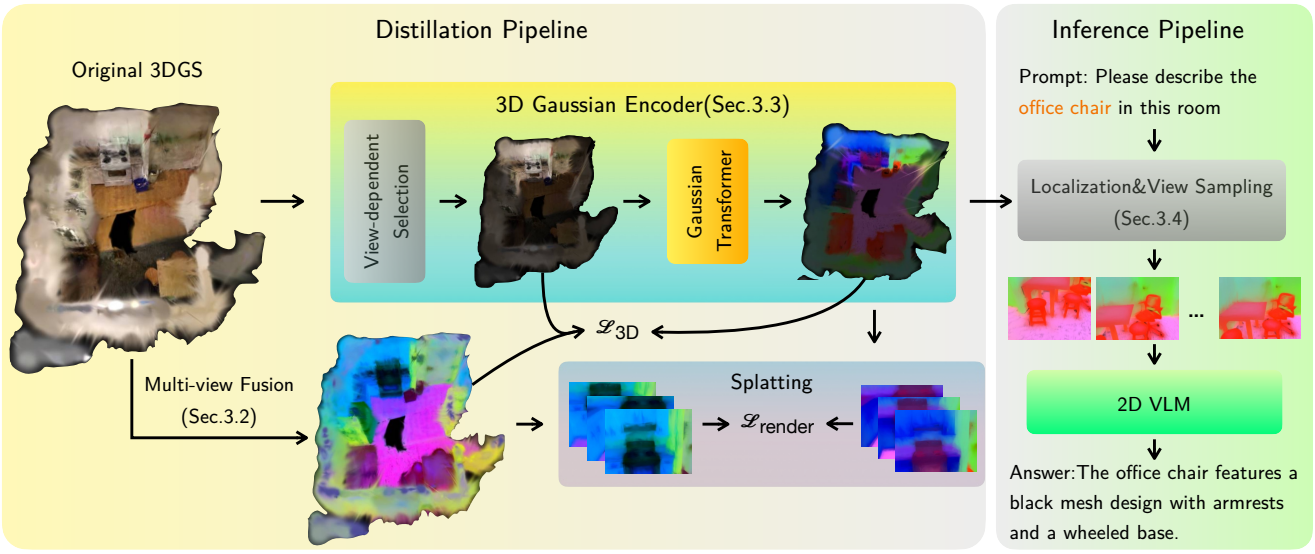

Figure 2. **Overview of *SplatDistill*.** (Left) The distillation pipeline begins with an original 3DGS reconstructed from multi-view images. Our 3D Gaussian Encoder (Sec.3.3) processes these Gaussians through view-dependent selection and a Gaussian Transformer to produce feature-enriched 3D Gaussians. The framework is supervised by two distillation losses: $\mathcal{L}_{3D}$ applied directly to 3D Gaussians via multi-view fusion (Sec. 3.2), and $\mathcal{L}_{render}$ applied to 2D rendered feature maps. (Right) The inference pipeline demonstrates how the feature-enriched 3D Gaussians enable object localization and strategic view sampling (Sec.3.4) in response to text prompts. The sampled views and their feature maps are fed to a 2D VLM to enable 3D scene understanding without specialized 3D training.

Multi-view Fusion. For each Gaussian g from the set of Gaussians $\mathcal{G}$, we first project it onto the image plane using the camera intrinsics K_j and world-to-camera extrinsics E_j of the jth frame. The corresponding pixel coordinate can be calculated as $\boldsymbol{u} = K_j \cdot E_j \cdot \boldsymbol{g}(\boldsymbol{\mu})$.(where the homogeneous representations of $\boldsymbol{u}$ and $\boldsymbol{\mu}$ are omitted for simplicity). We also conduct occlusion tests to guarantee that only visible Gaussians are considered by comparing them with the rendered depth map. More details about this can be found in the supplementary materials.

Given the projected pixel coordinate $\boldsymbol{u}$, the corresponding feature can be obtained via $\mathbf{f}_j = \mathbf{F}_j[\boldsymbol{u}]$, where $\mathbf{F}_j \in \mathbb{R}^{H,W,D}$ is the RADIO backbone feature map and D refers to the feature dimension. Assuming N views are available for fusion, the fused feature vector for Gaussian $\boldsymbol{g}$ can be computed as the mean of the corresponding features across these N views: $\mathbf{f}_{\text{fused}} = \text{mean}(\mathbf{f}_1, ..., \mathbf{f}_N)$. By repeating this fusion process for each Gaussian, we establish a feature-enriched 3D Gaussian set: $\mathcal{G}_{\text{fused}} = \{(\mathbf{f}_{\text{fused}}, \boldsymbol{\mu}, \text{SH}, \boldsymbol{r}, \boldsymbol{s}, \boldsymbol{\alpha})_i\}_{i=1:M}$.

3.3. 3D Feature Distillation

While the per-scene generated 3D feature Gaussians can be directly applied to various downstream tasks, this approach requires repeating the feature extraction and multi-view fusion procedure for each scene which can take several minutes to process. Moreover, such fused features can be noisy due to potentially inconsistent 2D predictions. Therefore, we propose to distill such knowledge from the 2D foundation model into a 3D network that takes a set of 3D Gaussians as input, as shown in Fig. 2.

Specifically, given a set of 3D Gaussians $\mathcal{G}$, we aim to learn a network ε_{3D} that outputs per-Gaussian embeddings $\mathbf{F}_{3D} = \varepsilon_{3D}(\mathcal{G})$, where $\mathbf{F}_{3D} = \{\mathbf{f}_{3D}^{\mathbf{1}}, ..., \mathbf{f}_{3D}^{\boldsymbol{M}}\}$. A strong limitation of learning such a network on 3D Gaussians is the large amounts of Gaussians, *i.e.* millions of Gaussians. Directly feeding these 3D Gaussians into a 3D network like PointTransformer [82] can cause the out-of-memory issue. Motivated by this finding, we explore methods to reduce the number of Gaussians. While a straightforward approach would be to voxelize the Gaussians with larger voxel sizes, this leads to the omission of too many tiny Gaussians, resulting in rendering artifacts. On the other hand, we observe that rendering any specific view typically utilizes only a small portion of the scene's 3D Gaussians. Based on this observation, we propose a view-dependent selection strategy that further reduces the input Gaussians after conventional voxelization. Specifically, we first sample several target views and construct view frustums from their camera extrinsics. We then filter out Gaussians outside these view frustums, using the remaining Gaussians as input for the 3D network. We utilize PointTransformer V3 [70] as our 3D backbone network ε_{3D} and change its output dimension to D.

To ensure consistency between output features and fused features, we employ cosine similarity loss as suggested by [60]. Additionally, to enable pre-trained adapter heads to serve as drop-in replacements for mapping backbone features to different target embedding spaces, we maintain the mag-

nitude of feature vectors using smooth L1 loss [23]. Thus, we employ a combination of cosine similarity and smooth L1 as,

$$\begin{aligned}\mathcal{L}_{\text{match}}(x,y) &= \alpha(1-\cos(x,y)) + \beta\mathcal{L}_{\text{l1-smooth}}(x,y)\\ \mathcal{L}_{\text{3D}} &= \sum_{i}^{M}\mathcal{L}_{\text{match}}(\mathbf{f}^{i}_{\text{3D}}, \mathbf{f}^{i}_{\text{fused}})\end{aligned} \tag{3}$$

where α, β are two balance factors. We empirically set $\alpha = 0.9$ and $\beta = 0.1$. To further ensure that the distilled features remain compatible with pretrained RADIO adapters, we apply the same matching loss between the 2D rendered feature maps and the RADIO 2D features after processing both through pretrained adapter ϕ_h,

$$\mathcal{L}_{\text{render}} = \sum_{h}^{H}\sum_{j}^{N}\mathcal{L}_{\text{match}}(\phi_h(\boldsymbol{\mathcal{R}}(\boldsymbol{\mathcal{G}}_{\text{fused}}, K_j, E_j)), \phi_h(\mathbf{F}_j)) \tag{4}$$

where $\boldsymbol{\mathcal{R}}$ is the feature rendering process from the given camera parameters and H is the number of adapter heads. Therefore, the final loss function is expressed as,

$$\mathcal{L}_{\text{distill}} = \lambda_{\text{3D}}\mathcal{L}_{\text{3D}} + \lambda_{\text{render}}\mathcal{L}_{\text{render}} \tag{5}$$

where λ_{3D} and λ_{render} are set to 1.0, 0.2, respectively.

3.4. Inference

Unlike inference with traditional 2D VLMs, which typically requires uniform sampling of multiple frames from a testing sequence, we design a two-stage inference pipeline that leverages the feature-enriched 3DGS to efficiently 'focus' on regions of interest. As demonstrated in Fig. 2 (right side), given the set of Gaussians with distilled features, we first identify a small subset of Gaussians that covers the content of interest. The distilled features naturally reside in the same language-aligned embedding space as RADIO, enabling us to evaluate semantic similarity between per-Gaussian features and arbitrary text prompts through cosine similarity computation. Subsequently, we strategically sample multiple viewpoints around these identified regions to generate both rendered images and their corresponding feature maps via Gaussian splatting. These visual inputs serve as the foundation for 2D VLM inference, allowing the model to reason effectively about the 3D scene from optimal perspectives. Additional implementation details are provided in the supplementary material.

4. Experiments

In this section, we introduce our experimental analysis, starting with the setup and datasets in Sec. 4.1. We then evaluate our method on 3D scene understanding, including 3D QA in Sec. 4.3, dense captioning in Sec. 4.4, and visual grounding in Sec. 4.5. Lastly, we perform an ablation study in Sec. 4.6 to analyze the contributions of key components.

4.1. Experimental Setup

Datasets and Benchmarks. Our training and evaluation experiments are conducted on the ScanNet dataset [17], an extensive indoor 3D scene dataset containing 1,513 multi-modal indoor scenes where 1,201 scenes are utilized for training and 312 scenes are for validation. We conduct our evaluation across several benchmarks adhering to these training/validation splits, including ScanNetV2/ ScanNet200 [17, 64] for 3D scene understanding, ScanQA [5] for 3D QA, Scan2Cap [14] for 3D dense captioning, and ScanRefer [9] for 3D visual grounding. We also conduct experiments with qualitative results on ScanNet++ [75] in supplementary materials.

Implementation Details. To prepare the training data, we first utilize the 3DGS reconstruction pipeline provided in Gsplat studio [74] to generate the 3DGS representation for each scene in ScanNet. To facilitate the optimization process, we instead initialize the Gaussian positions with the subsampled points from the given scene mesh vertex. For the feature distillation, we utilize the RADIOv2.5 ViT-H/16 [63] as the 2D foundation model to extract the 2D feature map from each input RGB frame for its higher resolution output. The Gaussian encoder is trained on 8×A100 80G GPUs for 100 epochs on the training scenes in ScanNetV2. During the inference, without the access of ground-truth meshes, we reconstruct the 3D Gaussians via LoopSplat [89]. To achieve the zero-shot scene segmentation, we directly utilize the distilled Gaussian feature. Meanwhile, for 3D QA tasks, we render these 3D Gaussians along with features to obtain multiple-view feature maps and feed into the pre-trained 2D VLMs. In the following experiments, we utilize pre-trained Llama-3-VILA1.5-8B [48] as the VLM backend by default unless specified.

In our experiments, we evaluate two variants of our approach based on how to obtain 3D feature Gaussians. "Ours (Fused)" refers to features obtained by multi-view fusion (Sec. 3.2) taking around 5-10 minutes, and "Ours (Distilled)" denotes features generated in a single forward pass through the 3D Gaussian network (Sec. 3.3) within 10 seconds.

4.2. Evaluation on Scene Understanding

To evaluate the quality of distilled 3D Gaussian features, we conduct experiments under ScanNet to explore the performance of zero-shot recognition and open-vocabulary instance segmentation.

3D Object Recognition. 3D object recognition aims to recognize the category of the query 3D instance. We compare our approach among all state-of-the-art (SOTA) 3D recognition models: PointCLIP (v1 and v2) [80, 90], CLIP^2 [78], CLIP2Point [33] and Uni3D [15]. We follow the same setting as CLIP^2 where the ground truth instance segmentation is provided for all the methods. As these methods do not use any scene from ScanNet during training, for a fair compari-

Method	Avg.	Bed	Cab	Chair	Sofa	Tabl	Door	Wind	Bksf	Pic	Cntr	Desk	Curt	Fridg	Bath	Showr	Toil	Sink
PointCLIP [80]	6.3	0.0	0.0	0.0	0.0	0.7	0.0	0.0	91.8	0.0	0.0	0.0	15.0	0.0	0.0	0.0	0.0	0.0
PointCLIP V2 [90]	11.0	0.0	0.0	23.8	0.0	0.0	0.0	0.0	7.8	0.0	**90.7**	0.0	0.0	0.0	0.0	64.4	0.0	0.0
CLIP2Point [33]	24.9	20.8	0.0	85.1	43.3	26.5	69.9	0.0	20.9	1.7	31.7	27.0	0.0	1.6	46.5	0.0	22.4	25.6
PointCLIP w/ TP.	26.1	0.0	55.7	72.8	5.0	5.1	1.7	0.0	77.2	0.0	0.0	51.7	0.3	0.0	0.0	40.3	85.3	49.2
CLIP2Point w/ TP.	35.2	11.8	3.0	45.1	27.6	10.5	61.5	2.6	71.9	0.3	33.6	29.9	4.7	11.5	72.2	92.4	86.1	34.0
CLIP^2 [78]	38.5	32.6	**67.2**	69.3	42.3	18.3	19.1	4.0	62.6	1.4	12.7	52.8	40.1	9.1	59.7	41.0	71.0	45.5
Uni3D [85]	45.8	58.5	3.7	78.8	83.7	**54.9**	31.3	39.4	70.1	**35.1**	1.9	27.3	**94.2**	13.8	38.7	10.7	88.1	47.6
OpenIns3D [34]	60.8	**85.2**	27.4	**87.6**	77.3	46.9	54.8	**64.2**	71.4	9.9	80.8	82.7	71.6	**61.4**	38.7	0.0	87.9	85.7
Ours (Distilled)	**70.7**	81.2	52.4	83.1	**87.5**	37.7	**84.6**	53.9	67.6	9.8	56.0	**89.3**	55.0	56.5	**97.7**	**95.8**	**96.4**	**100.0**

Table 1. **Quantitative comparison with SOTA models for 3D object recognition on ScanNetv2.** We report the top-1 classification accuracy.

Model	AP_{25}	AP_{50}	AP	AP_{Head}	AP_{Common}	AP_{Tail}
Closed-vocabulary, fully supervised						
ISBNet [56]	37.6	32.7	24.5	38.6	20.5	12.5
Mask3D [9]	36.2	41.4	26.9	39.8	21.7	17.9
Open-vocabulary, per-scene						
OpenScene [60](Fused) + Mask3D	17.8	15.2	11.7	13.4	11.6	9.9
OpenMask3D [66]	23.1	19.9	15.4	17.1	14.1	14.9
Open3DIS* [57]	27.3	23.1	18.6	24.7	16.9	13.3
OpenIns3D [34]	14.4	10.3	8.8	16.0	6.5	4.2
Ours (Fused)	28.1	24.3	19.1	22.6	18.6	11.1
Open-vocabulary, feed-forward						
OpenScene [60] (Distilled) + Mask3D	7.2	6.2	4.8	10.6	2.6	0.7
Ours (Distilled)	10.2	9.3	7.4	16.5	3.0	0.9

Table 2. **Quantitative comparison with SOTA models on ScanNet200 for 3D instance segmentation.** Metrics are respectively: AP averaged over an overlapping range, and AP evaluated at 50% and 25% overlaps.

son, we also compare our approach with OpenIns3D where a per-scene fusion strategy is utilized to map 2D segmentations into 3D space (note that no annotations are used). All results are summarized in Table 1 where our approach produces much more accurate recognition results than existing works.

Open-vocabulary 3D Instance Segmentation. To further show the effectiveness of the distilled Gaussian feature, we adopt the distilled Gaussian feature to open-vocabulary 3D instance segmentation on ScanNet200. We consider three groups of existing SOTA methods: (i) the fully supervised models that utilize the closed-set classification annotations, including ISBNet [56] and Mask3D [28]. (ii) the per-scene optimization approaches that utilize the class-agnostic 3D mask proposals and lift up the 2D feature/segmentation map from 2D foundation models, *i.e.*, CLIP, into 3D space via some multi-view fusion strategies, *i.e.*, OpenScene [60](using 2D fusion features), OpenMask3D [66], Open3DIS [57] and OpenIns3D [34]. (iii) the feed-forward models that directly map 3D representations to the language embedding space.

Since our approach generates per-Gaussian features whose positions do not align with the original point clouds, we cannot evaluate the 3D instance segmentation results on 3D Gaussians with the ground-truth annotations on point clouds. To address this issue, we utilize the class-agnostic 3D mask proposals to crop out the 3D Gaussians within each mask proposal and recognize the corresponding category. For a fair comparison, we utilize the class-agnostic 3D mask proposals generated by Mask3D [28] for the evaluation of most existing approaches unless otherwise specified. As reported in Table 2, our approach achieves the best performance among the per-scene optimization models and feed-forward models in terms of AP_{25}, AP_{50}, and AP. However, we have observed a decline in performance in the tail classes of ScanNet200. The decrease in performance can be attributed to the low-quality reconstruction of smaller objects in the ScanNet200 scene and the misalignment between the original point clouds and the Gaussian points.

4.3. Evaluation on 3D Question Answering

3D question answering requires a model to generate responses to the natural language queries questioning towards a 3D scene. In this section, we validate our approach on ScanQA [14] for 3D question answering.

Baseline Models. We mainly compare our approach with existing 2D (image/video) and 3D VLMs for different downstream tasks. Existing methods are split into three groups: (i) Discriminative task-specific models perform closed-set classification via detected object proposals. (ii) Zero-shot VLMs include the generalist VLMs trained on internet data while without access to the 3D ground-truth question-answer pairs from several benchmarks. (ii) Fine-tuned VLMs cover the

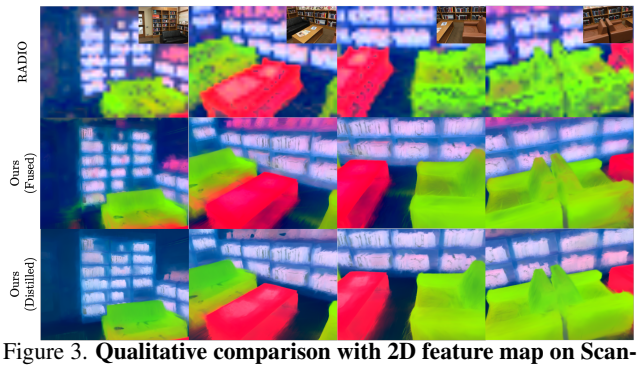

Figure 3. **Qualitative comparison with 2D feature map on ScanNet.** PCA visualizations of feature maps obtained via: RADIOv2.5 ViT-H/16 [63], rendered from multi-view fused 3D feature Gaussians and rendered from distilled 3D feature Guassians.

	C↑	B-4↑	M↑	R↑	EM@1↑
Task-specific models					
ScanQA [5]	64.9	10.1	13.1	33.3	21.1
3D-VisTA [92]	69.6	10.4	13.9	35.7	22.4
Fine-tuned 3D VLMs					
3D-LLM (FlanT5) [26]	69.4	12.0	14.5	35.7	20.5
LL3DA [13]	76.8	13.5	15.9	37.3	–
Chat-3D v2 [29]	87.6	14.0	–	–	–
LEO [30]	101.4	13.2	20.0	49.2	24.5 (47.6)
Scene-LLM [22]	80	12.0	16.6	40.0	27.2
LLaVA-3D [88]	91.7	14.5	20.7	50.1	27.0 (45.0)
Zero-shot 2D VLMs					
VideoChat2 [47]	49.2	9.6	9.5	28.2	19.2
LLaVA-NeXT-Video [81]	46.2	9.8	9.1	27.8	18.7
Llama3-VILA-1.5 [48]	64.4	–	13.5	35.2	22.4
GPT-4V	59.6	–	13.5	33.4	–
Gemini	68.3	–	11.3	35.4	–
Claude	57.7	–	10.0	29.3	–
Ours (Fused)	72.2	–	16.0	38.8	25.4 (42.1)
Ours (Distilled)	67.5	–	15.2	37.7	23.8 (40.4)

Table 3. **Quantitative comparison with SOTA models on ScanQA for 3D QA task**. "C" stands for "CIDEr", "B-4" for "BLEU-4", "M" for "METEOR", "R" for "ROUGE", and "EM@1" for top-1 exact match. Gray indicates evaluation results with refined exact-match protocol as suggested in [30].

VLMs finetuned with these 3D ground-truth question-answer pairs for particular tasks and multi-tasks.

Results on ScanQA. ScanQA is a subset of ScanNet that contains 41,363 questions about 800 scenes and we evaluate our approach on its validation set which consists of 4,675 questions about 71 scenes. Following prior works, we adopt BLEU scores, METEOR, ROUHE-L, CIDEr and EM ("exact match") as our evaluation metrics. For existing zero-shot 2D Language-Vision Models (VLMs), we uniformly sample N frames (where N=8 in our experiments) as they exclusively take multi-view images as input. As reported in Table 3, all evaluated zero-shot 2D VLMs demonstrate relatively poor performance compared to recent 3D VLMs, which are trained with 3D ground-truth annotations and incorporate carefully designed 3D-awareness modules. In contrast, our approach has never been fine-tuned on any 3D QA pairs,

	C@0.5↑	B-4@0.5↑	M@0.5↑	R@0.5↑
Task-specific models				
Scan2Cap [14]	39.08	23.32	21.97	44.78
3D-VisTA [92]	61.60	34.10	26.80	55.00
Fine-tuned 3D VLMs				
LL3DA [13]	65.19	36.79	25.97	55.06
LEO [30]	72.4	38.2	27.9	58.1
LLaVA-3D [88]	79.21	41.12	30.21	63.41
Zero-shot 2D VLMs				
LLama-3.2 [21]	3.97	4.81	20.86	28.00
Phi-3 [1]	4.21	4.97	24.36	34.14
InternVL2 [16]	3.68	4.83	23.81	35.23
LLaVA-OneVision [45]	21.29	8.88	25.21	41.44
Owen2-VL [6]	10.96	6.78	25.60	38.93
Llama3-VILA-1.5 [48]	23.34	9.45	26.67	43.71
Ours (Fusion)	30.08	11.74	27.05	46.01
Ours (Distilled)	29.17	11.22	26.58	45.46

Table 4. **Quantitative comparison with SOTA models on Scan2Cap for dense captioning task**. "C" stands for "CIDEr", "B-4" for "BLEU-4", "M" for "METEOR", and "R" for "ROUGE".

and yet outperforms all existing zero-shot VLMs by a large margin and even achieves comparable performance to some fine-tuned 3D VLMs, *i.e.*, LL3DA [13] and 3D-LLM [26]. This significant improvement can be attributed to the more relevant views sampled from the ROI obtained by our 3D Gaussian localization pipeline.

4.4. Evaluation on 3D Dense Captioning

3D dense captioning requires the model to localize the objects of interest in a 3D scene and then generate descriptive sentences for these objects. Following the previous work [30, 92], we utilize the off-the-shelf segmentation model Mask3D [28] to generate object proposals. Then we replace the object positions obtained from the localization step mentioned in the 3D QA section with these object proposals. To generate the 3D dense captioning of the given object, we follow the same viewpoints sampling strategy described before (Sec 4.3) to generate input images for the pre-trained 2D VLM. We utilize the textual instructions that prompt the model to describe the object's appearance and the spatial relations with nearby objects. As shown in Table 4, our approach achieves comparable performance with finetine 3D VLMs while consistently surpassing the zero-shot 2D VLMs across all evaluation metrics.

4.5. Evaluation on 3D Visual Grounding

3D visual grounding aims to localize the target object in the 3D scene using natural language descriptions. We evaluate the performance on the ScanRefer [9] benchmark. For quantitative comparisons, we found none of the existing zero-shot 2D VLMs can output a reasonable object localization due to the lack of explicit 3D representation. On the other hand, our approach using Gaussian features enables us to calculate the similarity between each Gaussian and the text prompt. We

	Acc@0.25	Acc@0.5
Task-specific models		
ScanRefer [9]	37.3	24.3
ReGround3D [87]	53.1	41.1
Fine-tuned 3D VLMs		
LLM-Grounder [71]	17.1	5.3
3D-LLM [26]	30.3	–
Chat3D-v2 [29]	35.9	30.4
LLaVA-3D [88]	54.1	42.2
Zero-shot 2D VMMs		
None of existing zero-shot 2D LLMs can achieve this task.		
Ours (Fusion)	40.2	16.0
Ours (Distilled)	43.2	18.2

Table 5. **Quantitative comparison with SOTA models on ScanRefer for 3D VG task.** We report the top-1 accuracy with 3D bounding box IOU over 25% and 50%.

	ScanNet200			ScanRefer	
Method	AP_{25}	AP_{50}	AP	Acc@0.25	Acc@0.5
Ours w.o. render loss	8.0	7.3	5.8	41.5	17.1
Ours w. render loss	10.2	9.3	7.4	43.2	18.2
improvements	+2.2	+2.0	+2.6	+1.7	+1.1

Table 6. **Ablation study on Rendering loss.**

Model	AP_{25}	AP_{50}	AP	Time(min)
Feature-3DGS [86]	17.4	16.0	14.6	∼120
LEGaussian [65]	8.2	7.8	7.7	∼60
LangSplat [61]	14.3	13.8	12.2	∼ 100
Ours (Fused)	28.9	26.2	22.8	∼5
Ours (Distilled)	17.9	17.6	16.3	≤1

Table 7. **Comparison results with 3DGS-based optimization approaches on ScanNet200.**

set a threshold to figure out several regions with higher similarity scores that could belong to the target object. As the distribution of 3D Gaussians can be noisy (*e.g.*, like floaters in the air), we further utilize the DBSCAN algorithm to filter out outliers and the final output is a set of 3D bounding boxes that cover each Gaussian grouping. Since this process might result in multiple candidate regions, we choose the one that has the highest IOU score with the ground-truth 3D bounding box when calculating the top-1 accuracy.

4.6. Ablation Study

Effectiveness of Rendering Loss. We first show the effectiveness of the rendering loss used during 3D Gaussian feature distillation. We assess performance on two tasks: 3D instance segmentation on ScanNet200 and 3D visual grounding on ScanRefer benchmarks. As shown in Table 6, incorporating the rendering loss yields improvements of 2.6% AP on ScanNet200 and 1.1% Acc@0.5 on ScanRefer, demonstrating enhanced scene understanding capability.

Comparison with per-scene optimization. To show the quality of distilled Gaussian feature from the feed-forward Gaussian encoder, we compare it with some per-scene optimization approaches, *i.e.*, LangSplat [61], Feature3DGS [86] and LEGaussian [65] on the task of open vocabulary instance segmentation as shown in Table 7. Note that the results are evaluated on 12 scenes randomly sampled from the ScanNet validation set due to the computational demands of per-scene optimization methods. Our approach achieves better performance with much less time, *i.e.*, 1 min *vs*. 100 mins.

	ScanQA		Scan2Cap		
Method	C↑	EM@1↑	B-4@0.5↑	M@0.5↑	R@0.5↑
LEO [30]	101.4	24.5	38.2	27.9	58.1
LLaVA-3D [88]	91.7	27.0	41.1	30.2	63.4
Video-3D LLM [83]	102.1	30.1	41.3	–	–
Ours(Finetuned)	101.9	32.5	30.2	25.9	61.6

Table 8. **Comparison results with existing 3D VLMs under fine-tuning setting.**

# Views	C↑	M↑	R↑	EM@1↑
16	68.6 (65.6)	15.5 (15.1)	37.2 (36.5)	23.6 (22.7)
32	71.7 (67.1)	15.8 (15.4)	38.4 (37.4)	25.1 (23.3)
64	72.2 (67.5)	16.0 (15.2)	38.8 (37.7)	25.4 (23.8)

Table 9. **Ablation study on the number of sampling views.** Gray indicates evaluation results using the distillation model.

Finetune on 3D data. Although the goal of our approach is to enable the 3D spatial understanding ability of 2D VLM without fine-tuning on 3D data, our approach can be further fine-tuned with ground-truth 3D QA pairs similar to LLAVA-3D [88]. The comparison results are listed in Table 8. Under a similar fine-tuning setting, our approach achieves comparable results with state-of-the-art on ScanQA and Scan2Cap.

Number of Sampling views. We also show comparison results by varying the number of views sampled from the scene for the evaluation 3D visual question answering. While maintaining a constant input of 8 frames for the 2D VLM, we vary the number of candidate views from which these frames are selected. As presented in Table 9, performance tends to improve with an increasing number of candidate sampling views. However, this improvement comes with increased computational cost, presenting a trade-off between performance and efficiency.

5. Conclusion

In this work, we presented *SplatDistill*, a novel framework that bridges 3D Gaussian Splatting with foundation models through feature distillation. Given a set of 3D Gaussians, *SplatDistll* predicts the per-Gaussian features that align well with the 2D foundation models via the proposed 3D Gaussian Network. A view-dependent selection and two matching losses are introduced to distill the knowledge of the 2D foundation model onto 3D Gaussians. Based on *SplatDistill*, we design a flexible end-to-end 3D scene understanding pipeline using the extracted Gaussian Features and pre-trained 2D VLMs that achieves improved generalization and efficiency compared to existing methods on various tasks.

References

[1] Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024. 7

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1

[3] Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas Guibas. Referit3d: Neural listeners for fine-grained 3d object identification in real-world scenes. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 422–440. Springer, 2020. 3

[4] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022. 2

[5] Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. Scanqa: 3d question answering for spatial scene understanding. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19129–19139, 2022. 2, 3, 5, 7

[6] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023. 7

[7] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*, 2023. 1

[8] Daigang Cai, Lichen Zhao, Jing Zhang, Lu Sheng, and Dong Xu. 3djcg: A unified framework for joint dense captioning and visual grounding on 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16464–16473, 2022. 3

[9] Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. Scanrefer: 3d object localization in rgb-d scans using natural language. In *European conference on computer vision*, pages 202–221. Springer, 2020. 2, 3, 5, 6, 7, 8

[10] Dave Zhenyu Chen, Qirui Wu, Matthias Nießner, and Angel X Chang. D 3 net: A unified speaker-listener architecture for 3d dense captioning and visual grounding. In *European Conference on Computer Vision*, pages 487–505. Springer, 2022. 3

[11] Runnan Chen, Youquan Liu, Lingdong Kong, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, Yu Qiao, and Wenping Wang. Clip2scene: Towards label-efficient 3d scene understanding by clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7020–7030, 2023. 2

[12] Sijin Chen, Xin Chen, Chi Zhang, Mingsheng Li, Gang Yu, Hao Fei, Hongyuan Zhu, Jiayuan Fan, and Tao Chen. Ll3da: Visual interactive instruction tuning for omni-3d understanding reasoning and planning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26428–26438, 2024. 3

[13] Sijin Chen, Xin Chen, Chi Zhang, Mingsheng Li, Gang Yu, Hao Fei, Hongyuan Zhu, Jiayuan Fan, and Tao Chen. Ll3da: Visual interactive instruction tuning for omni-3d understanding reasoning and planning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26428–26438, 2024. 7

[14] Zhenyu Chen, Ali Gholami, Matthias Nießner, and Angel X Chang. Scan2cap: Context-aware dense captioning in rgb-d scans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3193–3203, 2021. 3, 5, 6, 7

[15] Zhenyu Chen, Ronghang Hu, Xinlei Chen, Matthias Nießner, and Angel X Chang. Unit3d: A unified transformer for 3d dense captioning and visual grounding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 18109–18119, 2023. 3, 5

[16] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *arXiv preprint arXiv:2312.14238*, 2023. 1, 7

[17] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Niessner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2, 5

[18] Alexandros Delitzas, Ayca Takmaz, Federico Tombari, Robert Sumner, Marc Pollefeys, and Francis Engelmann. Scenefun3d: Fine-grained functionality and affordance understanding in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14531–14542, 2024. 2

[19] Runyu Ding, Jihan Yang, Chuhui Xue, Wenqing Zhang, Song Bai, and Xiaojuan Qi. Pla: Language-driven open-vocabulary 3d scene understanding. In *CVPR*, 2023. 2

[20] Runyu Ding, Jihan Yang, Chuhui Xue, Wenqing Zhang, Song Bai, and Xiaojuan Qi. Lowis3d: Language-driven open-world instance-level 3d scene understanding. In *TPAMI*, 2024. 2

[21] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 7

[22] Rao Fu, Jingyu Liu, Xilun Chen, Yixin Nie, and Wenhan Xiong. Scene-llm: Extending language model for 3d visual understanding and reasoning. *arXiv preprint arXiv:2403.11401*, 2024. 2, 3, 7

[23] Ross Girshick. Fast r-cnn. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, 2015. 5

[24] Qiao Gu, Alihusein Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, Chuang Gan, Celso Miguel de Melo, Joshua B. Tenenbaum, Antonio Torralba, Florian Shkurti, and Liam Paull. Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning. *arXiv*, 2023. 2, 3

[25] Ziyu Guo, Yiwen Tang, Renrui Zhang, Dong Wang, Zhigang Wang, Bin Zhao, and Xuelong Li. Viewrefer: Grasp the multi-view knowledge for 3d visual grounding with gpt and prototype guidance. *arXiv preprint arXiv:2303.16894*, 2023. 3

[26] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *arXiv preprint arXiv:2307.12981*, 2023. 3, 7, 8

[27] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *Advances in Neural Information Processing Systems*, 36:20482–20494, 2023. 3

[28] Ji Hou, Xiaoliang Dai, Zijian He, Angela Dai, and Matthias Nießner. Mask3d: Pre-training 2d vision transformers by learning masked 3d priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13510–13519, 2023. 6, 7

[29] Haifeng Huang, Zehan Wang, Rongjie Huang, Luping Liu, Xize Cheng, Yang Zhao, Tao Jin, and Zhou Zhao. Chat-3d v2: Bridging 3d scene and large language models with object identifiers. *arXiv preprint arXiv:2312.08168*, 2023. 7, 8

[30] Jiangyong Huang, Silong Yong, Xiaojian Ma, Xiongkun Linghu, Puhao Li, Yan Wang, Qing Li, Song-Chun Zhu, Baoxiong Jia, and Siyuan Huang. An embodied generalist agent in 3d world. *arXiv preprint arXiv:2311.12871*, 2023. 3, 7, 8

[31] Rui Huang, Songyou Peng, Ayca Takmaz, Federico Tombari, Marc Pollefeys, Shiji Song, Gao Huang, and Francis Engelmann. Segment3d: Learning fine-grained class-agnostic 3d segmentation without manual labels. In *European Conference on Computer Vision*, pages 278–295. Springer, 2024. 2

[32] Shijia Huang, Yilun Chen, Jiaya Jia, and Liwei Wang. Multi-view transformer for 3d visual grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15524–15533, 2022. 3

[33] Tianyu Huang, Bowen Dong, Yunhan Yang, Xiaoshui Huang, Rynson WH Lau, Wanli Ouyang, and Wangmeng Zuo. Clip2point: Transfer clip to point cloud classification with image-depth pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22157–22167, 2023. 5, 6

[34] Zhening Huang, Xiaoyang Wu, Xi Chen, Hengshuang Zhao, Lei Zhu, and Joan Lasenby. Openins3d: Snap and lookup for 3d open-vocabulary instance segmentation. In *European Conference on Computer Vision*, pages 169–185. Springer, 2025. 2, 6

[35] Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Tao Chen, Shuang Li, Ganesh Iyer, Soroush Saryazdi, Nikhil Keetha, Ayush Tewari, et al. Conceptfusion: Open-set multimodal 3d mapping. *arXiv preprint arXiv:2302.07241*, 2023. 2

[36] Baoxiong Jia, Yixin Chen, Huangyue Yu, Yan Wang, Xuesong Niu, Tengyu Liu, Qing Li, and Siyuan Huang. Sceneverse: Scaling 3d vision-language learning for grounded scene understanding. In *European Conference on Computer Vision*, pages 289–310. Springer, 2024. 2

[37] Li Jiang, Shaoshuai Shi, and Bernt Schiele. Open-vocabulary 3d semantic segmentation with foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21284–21294, 2024. 2

[38] Yang Jiao, Shaoxiang Chen, Zequn Jie, Jingjing Chen, Lin Ma, and Yu-Gang Jiang. More: Multi-order relation mining for dense captioning in 3d scenes. In *European Conference on Computer Vision*, pages 528–545. Springer, 2022. 3

[39] Bu Jin, Yupeng Zheng, Pengfei Li, Weize Li, Yuhang Zheng, Sujie Hu, Xinyu Liu, Jinwei Zhu, Zhijie Yan, Haiyang Sun, et al. Tod3cap: Towards 3d dense captioning in outdoor scenes. In *European Conference on Computer Vision*, pages 367–384. Springer, 2024. 2

[40] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 2, 3

[41] Bernhard Kerbl, Andreas Meuleman, Georgios Kopanas, Michael Wimmer, Alexandre Lanvin, and George Drettakis. A hierarchical 3d gaussian representation for real-time rendering of very large datasets. *ACM Transactions on Graphics (TOG)*, 43(4):1–15, 2024. 2

[42] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. Lerf: Language embedded radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19729–19739, 2023. 2

[43] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. 3

[44] Sosuke Kobayashi, Eiichi Matsumoto, and Vincent Sitzmann. Decomposing nerf for editing via feature field distillation. *Advances in Neural Information Processing Systems*, 35:23311–23330, 2022. 2

[45] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. 7

[46] Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024. 2

[47] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22195–22206, 2024. 7

[48] Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26689–26699, 2024. 1, 2, 5, 7

[49] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 1

[50] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024. 2

[51] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. Llava-next: Improved reasoning, ocr, and world knowledge, 2024. 2

[52] Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. Sqa3d: Situated question answering in 3d scenes. *arXiv preprint arXiv:2210.07474*, 2022. 2

[53] Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul Mcvay, Oleksandr Maksymets, Sergio Arnaud, Karmesh Yadav, Qiyang Li, Ben Newman, Mohit Sharma, Vincent Berges, Shiqi Zhang, Pulkit Agrawal, Yonatan Bisk, Dhruv Batra, Mrinal Kalakrishnan, Franziska Meier, Chris Paxton, Sasha Sax, and Aravind Rajeswaran. Openeqa: Embodied question answering in the era of foundation models. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3

[54] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2

[55] Mahyar Najibi, Jingwei Ji, Yin Zhou, Charles R Qi, Xinchen Yan, Scott Ettinger, and Dragomir Anguelov. Unsupervised 3d perception with 2d vision-language distillation for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8602–8612, 2023. 2

[56] Tuan Duc Ngo, Binh-Son Hua, and Khoi Nguyen. Isbnet: a 3d point cloud instance segmentation network with instance-aware sampling and box-aware dynamic convolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13550–13559, 2023. 6

[57] Phuc Nguyen, Tuan Duc Ngo, Evangelos Kalogerakis, Chuang Gan, Anh Tran, Cuong Pham, and Khoi Nguyen. Open3dis: Open-vocabulary 3d instance segmentation with 2d mask guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4018–4028, 2024. 6

[58] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2, 3

[59] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2

[60] Songyou Peng, Kyle Genova, Chiyu Jiang, Andrea Tagliasacchi, Marc Pollefeys, Thomas Funkhouser, et al. Openscene: 3d scene understanding with open vocabularies. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 815–824, 2023. 2, 3, 4, 6

[61] Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. Langsplat: 3d language gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20051–20060, 2024. 2, 8

[62] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2, 3

[63] Mike Ranzinger, Greg Heinrich, Jan Kautz, and Pavlo Molchanov. Am-radio: Agglomerative vision foundation model reduce all domains into one. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12490–12500, 2024. 2, 3, 5, 7

[64] David Rozenberszki, Or Litany, and Angela Dai. Language-grounded indoor 3d semantic segmentation in the wild. In *European Conference on Computer Vision*, pages 125–141. Springer, 2022. 2, 5

[65] Jin-Chuan Shi, Miao Wang, Hao-Bin Duan, and Shao-Hua Guan. Language embedded 3d gaussians for open-vocabulary scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5333–5343, 2024. 2, 8

[66] Ayça Takmaz, Elisabetta Fedele, Robert W Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. Openmask3d: Open-vocabulary 3d instance segmentation. *arXiv preprint arXiv:2306.13631*, 2023. 6

[67] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 1

[68] Heng Wang, Chaoyi Zhang, Jianhui Yu, and Weidong Cai. Spatiality-guided transformer for 3d dense captioning on point clouds. *arXiv preprint arXiv:2204.10688*, 2022. 3

[69] Zehan Wang, Haifeng Huang, Yang Zhao, Ziang Zhang, and Zhou Zhao. Chat-3d: Data-efficiently tuning large language model for universal dialogue of 3d scenes, 2023. 3

[70] Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler faster stronger. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4840–4851, 2024. 4

[71] Jianing Yang, Xuweiyi Chen, Shengyi Qian, Nikhil Madaan, Madhavan Iyengar, David F Fouhey, and Joyce Chai. Llm-grounder: Open-vocabulary 3d visual grounding with large language model as an agent. *arXiv preprint arXiv:2309.12311*, 2023. 8

[72] Jihan Yang, Runyu Ding, Zhe Wang, and Xiaojuan Qi. Regionplc: Regional point-language contrastive learning for open-world 3d scene understanding. In *CVPR*, 2024. 2

[73] Mingqiao Ye, Martin Danelljan, Fisher Yu, and Lei Ke. Gaussian grouping: Segment and edit anything in 3d scenes. In *European Conference on Computer Vision*, pages 162–179. Springer, 2024. 2

[74] Vickie Ye, Ruilong Li, Justin Kerr, Matias Turkulainen, Brent Yi, Zhuoyang Pan, Otto Seiskari, Jianbo Ye, Jeffrey Hu, Matthew Tancik, and Angjoo Kanazawa. gsplat: An open-source library for Gaussian splatting. *arXiv preprint arXiv:2409.06765*, 2024. 5

[75] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2023. 5

[76] Yuanwen Yue, Anurag Das, Francis Engelmann, Siyu Tang, and Jan Eric Lenssen. Improving 2D Feature Representations by 3D-Aware Fine-Tuning. In *European Conference on Computer Vision (ECCV)*, 2024. 2, 3

[77] Yuanwen Yue, Anurag Das, Francis Engelmann, Siyu Tang, and Jan Eric Lenssen. Improving 2d feature representations by 3d-aware fine-tuning. In *European Conference on Computer Vision*, pages 57–74. Springer, 2025. 3

[78] Yihan Zeng, Chenhan Jiang, Jiageng Mao, Jianhua Han, Chaoqiang Ye, Qingqiu Huang, Dit-Yan Yeung, Zhen Yang, Xiaodan Liang, and Hang Xu. Clip2: Contrastive language-image-point pretraining from real-world point cloud data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15244–15253, 2023. 5, 6

[79] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023. 3

[80] Renrui Zhang, Ziyu Guo, Wei Zhang, Kunchang Li, Xupeng Miao, Bin Cui, Yu Qiao, Peng Gao, and Hongsheng Li. Pointclip: Point cloud understanding by clip. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8552–8562, 2022. 5, 6

[81] Yuanhan Zhang, Bo Li, haotian Liu, Yong jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. Llava-next: A strong zero-shot video understanding model, 2024. 7

[82] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16259–16268, 2021. 4

[83] Duo Zheng, Shijia Huang, and Liwei Wang. Video-3d llm: Learning position-aware video representation for 3d scene understanding. *arXiv preprint arXiv:2412.00493*, 2024. 2, 8

[84] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. 2

[85] Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3d: Exploring unified 3d representation at scale. *arXiv preprint arXiv:2310.06773*, 2023. 6

[86] Shijie Zhou, Haoran Chang, Sicheng Jiang, Zhiwen Fan, Zehao Zhu, Dejia Xu, Pradyumna Chari, Suya You, Zhangyang Wang, and Achuta Kadambi. Feature 3dgs: Supercharging 3d gaussian splatting to enable distilled feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21676–21685, 2024. 2, 8

[87] Chenming Zhu, Tai Wang, Wenwei Zhang, Kai Chen, and Xihui Liu. Empowering 3d visual grounding with reasoning capabilities. *arXiv preprint arXiv:2407.01525*, 2024. 8

[88] Chenming Zhu, Tai Wang, Wenwei Zhang, Jiangmiao Pang, and Xihui Liu. Llava-3d: A simple yet effective pathway to empowering lmms with 3d-awareness. *arXiv preprint arXiv:2409.18125*, 2024. 2, 7, 8

[89] Liyuan Zhu, Yue Li, Erik Sandström, Shengyu Huang, Konrad Schindler, and Iro Armeni. Loopsplat: Loop closure by registering 3d gaussian splats. *arXiv preprint arXiv:2408.10154*, 2024. 5

[90] Xiangyang Zhu, Renrui Zhang, Bowei He, Ziyu Guo, Ziyao Zeng, Zipeng Qin, Shanghang Zhang, and Peng Gao. Pointclip v2: Prompting clip and gpt for powerful 3d open-world learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2639–2650, 2023. 5, 6

[91] Xiaoyu Zhu, Hao Zhou, Pengfei Xing, Long Zhao, Hao Xu, Junwei Liang, Alexander Hauptmann, Ting Liu, and Andrew Gallagher. Open-vocabulary 3d semantic segmentation with text-to-image diffusion models. *arXiv preprint arXiv:2407.13642*, 2024. 2

[92] Ziyu Zhu, Xiaojian Ma, Yixin Chen, Zhidong Deng, Siyuan Huang, and Qing Li. 3d-vista: Pre-trained transformer for 3d vision and text alignment. *arXiv preprint arXiv:2308.04352*, 2023. 7

[93] Xingxing Zuo, Pouya Samangouei, Yunwen Zhou, Yan Di, and Mingyang Li. Fmgs: Foundation model embedded 3d gaussian splatting for holistic 3d scene understanding. *International Journal of Computer Vision*, pages 1–17, 2024. 2