RGB-Pointmap Pretraining for Unified 3D Scene Understanding

Ye Mao, Weixun Luo, Ranran Huang, Junpeng Jing*, Krystian Mikolajczyk

Imperial College London
https://yebulabula.github.io/UniScene3D/

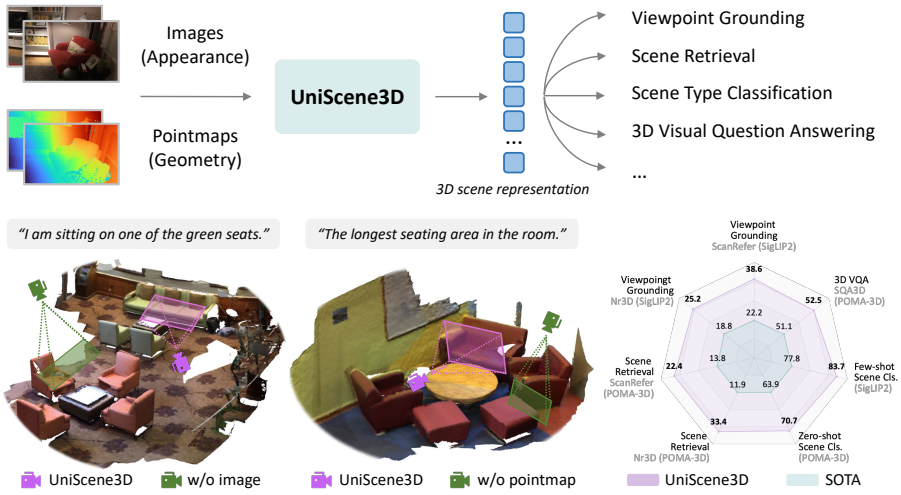

Fig. 1: Overview of UniScene3D. *Top:* UniScene3D takes multi-view images and pointmaps as input to learn 3D representations for viewpoint grounding, scene retrieval, zero-/few-shot scene type classification, and 3D visual question answering. *Bottom:* Example of viewpoint grounding. Image appearance cues enable correct color recognition (left), while pointmap geometry supports reasoning about spatial extent, enabling identification of the longest seat (right). The radar chart shows comparisons between UniScene3D and prior state-of-the-art methods across multiple tasks and benchmarks.

Abstract. Pretraining 3D encoders through alignment with Contrastive Language–Image Pre-training (CLIP) has emerged as a promising direction to learn generalizable representations for 3D scene understanding. In this paper, we propose UniScene3D, a transformer-based framework that learns unified 3D scene representations from multi-view RGB–Pointmap by leveraging the priors of a pretrained 2D foundation model. For robust RGB-Pointmap representation learning, we introduce novel cross-view geometric alignment and grounded view alignment to enforce geometry and semantic consistency across views. Extensive low-shot and task-specific fine-tuning across viewpoint grounding, scene retrieval, scene classification, and 3D visual question answering achieves state-of-the-art

* Corresponding author: j.jing23@imperial.ac.uk

performance. These results establish UniScene3D as an effective framework for unified 3D scene understanding.

Keywords: Representation Learning · 3D Scene Understanding

1 Introduction

3D scene understanding, covering tasks such as 3D segmentation [21, 48] and 3D visual question answering (3D VQA) [3, 51], is a fundamental capability for embodied agents and VR/AR systems [14, 31]. Learning generalizable 3D representations that can be transferred across these scene tasks has been a long-standing research topic. Recently, an emerging 3D representation learning paradigm is to align 3D encoders with pretrained vision–language models such as CLIP [43, 49].

Within this paradigm, recent works explore different input modalities for training CLIP-aligned 3D encoders, including point clouds [15, 34, 53, 63], multi-view images [13, 28, 64], depth maps [8, 37], and pointmaps [38]. Each modality offers distinct advantages but also exhibits inherent limitations. Point clouds capture explicit 3D geometry, but their irregular and unordered structure makes them incompatible with grid-based architectures, limiting the effective use of pretrained 2D priors. Multi-view images and depth maps preserve a regular grid structure, yet they lack globally consistent 3D geometry at the scene level. This is because each view is defined in its individual image or camera frame. In contrast, pointmaps utilize a shared world frame to encode coherent 3D coordinates while preserving an image-like grid representation, making them a promising modality for 3D representation learning with standard 2D backbones.

Despite its potential, research on pointmap-based representation learning remains in an early stage, and several key challenges remain underexplored. First, pointmap itself only has geometric coordinates and lacks appearance cues such as color and texture. It limits their ability to capture visually grounded semantics, especially in 3D understanding scenarios that require reasoning over visual attributes (see Fig. 1). Second, the geometric consistency across multiple pointmap views provides a natural signal for cross-view representation learning. This encourages robust scene representations that overcome the partial observability and view-dependent occlusions inherent to single-view modeling. Conventional multi-view learning strategies [13, 28, 37, 64] process views independently and thus struggle to fully exploit this structural property.

To address these challenges, we introduce UniScene3D, a 3D transformer encoder that learns **Uni**fied **Scene** representations jointly capturing appearance and geometry from multi-view image–pointmap pairs, which we refer to as ***RGB-Pointmap***. Specifically, we propose an early fusion module that integrates image and pointmap tokens at the patch embedding stage, enabling a single encoder to process both modalities. We further design multimodal alignment objectives to explicitly capture the cross-view relations of RGB-Pointmap. In particular, cross-view geometric alignment enforces similarity between spatially proximate views and distinction for distant ones. Grounded view alignment as-

sociates an object referring text with the views that observe the object, thereby encouraging aligned representation for views with shared object semantics.

The effectiveness of our method is validated across multiple 3D scene understanding tasks, including viewpoint grounding, scene retrieval, scene type classification, and 3D VQA. These tasks probe representation quality from object-level and view-level reasoning to holistic scene understanding, with potential applications in robotic navigation, embodied perception, and human–robot interaction [7,45]. Experimental results show that UniScene3D consistently outperforms state-of-the-art vision encoders across all tasks (see Fig. 1) under zero-shot, few-shot, and task-specific fine-tuning settings. These results highlight that our model effectively encodes both geometric and appearance cues in RGB-Pointmap to produce generalizable 3D representations. The key contributions are summarized below:

- We propose UniScene3D, a vision encoder that learns unified 3D scene representations by jointly modeling geometry and appearance from multi-view RGB-Pointmap.
- We introduce novel alignment objectives, including cross-view geometric alignment and grounded view alignment, to capture geometric and semantic consistency in RGB-Pointmap across viewpoints.
- We evaluate our method across diverse downstream 3D scene understanding tasks under zero-shot, few-shot, and task-specific fine-tuning settings, achieving state-of-the-art performance.

2 Related Work

In this section, we first review existing 3D representation learning methods based on vision–language pretraining, and then summarize commonly used 3D scene datasets for pretraining and the evaluation protocols for vision–language models.

3D Vision–Language Pretraining. 3D vision–language pretraining aligns a 3D encoder with pretrained CLIP models [17,43,47,49] and has become a common paradigm for 3D representation learning. Most previous works adopt point clouds as the input modality to train encoders. PointCLIP-series [60,65], ULIP-series [53,54], PointBind [20], CLIP^2 [58] OpenShape [34], Uni3D [63], MixCon3D [19], and OccTIP [41] pretrain point cloud encoders that show strong zero-shot performance on 3D object recognition and cross-modal 3D object retrieval. Subsequent work extends this paradigm to scene-level understanding [2, 8, 12, 25, 55, 66]. However, point cloud encoders operate in a modality misaligned with 2D pretrained backbones, preventing direct reuse of rich 2D priors and requiring large-scale 3D pretraining to maintain generalization.

To better exploit 2D pretrained priors, alternative approaches adopt 2D-compatible representations. DuoduoCLIP [28] uses multi-view images, while OpenDlign [37] and CLIP2Point [24] use depth maps to adapt 2D encoders to 3D tasks. However, these projections retain incomplete geometric information, which limits performance on complex and global spatial reasoning. Recent methods based on 3D Gaussian splatting [26,29,30,32,42] provide another direction,

but are primarily evaluated on 3D segmentation, restricting their demonstrated applicability. POMA-3D [38] is an early attempt at learning 3D representations from pointmaps. However, POMA-3D relies solely on pointmaps and thus misses important appearance cues. Its multi-stage training and JEPA-style cross-view alignment also incur substantial training costs. In contrast, our method jointly models geometry and appearance from images and pointmaps to learn unified 3D representations. The proposed geometric and grounded view alignment objectives further enable UniScene3D to exploit cross-view information more effectively while remaining computationally efficient.

3D Scene Pretraining Datasets. Collecting accurate 3D scene data at scale remains challenging due to the high cost and time-intensive nature of 3D scanning and reconstruction [5, 18, 22, 23, 46, 46, 56]. Existing real-world datasets, including ScanNet [10], 3RScan [50], ARKitScenes [4], HM3D [44], and MultiScan [40], contain only thousands of scenes, orders of magnitude fewer than the billions of images used for 2D pretraining. Some works [25] mitigate this limitation by incorporating synthetic datasets [11, 62]. However, the domain gap between synthetic and real-world environments often constrains transferability. Accordingly, many 3D pretraining methods [34, 53, 63] transfer knowledge from large-scale pretrained vision models using limited 3D data. Following this principle, UniScene3D is post-trained from a pretrained 2D image encoder and leverages thousands of real-world 3D scenes to learn transferable 3D representations.

Evaluation Protocol for Vision–Language Models. Low-shot evaluation (i.e., zero-shot and few-shot) provides a standardized protocol for assessing vision–language pretrained encoders by measuring representation quality under minimal supervision, thereby isolating encoder capacity. In the 2D domain, CLIP models [17, 43, 47, 49, 59] are commonly evaluated through low-shot image classification. In 3D, object-centric models [34, 37, 53, 63, 65] are typically assessed via low-shot 3D object classification and retrieval.

In contrast, most 3D scene representation models are evaluated only under task-specific fine-tuning on benchmarks [3, 6, 9, 10, 36, 39, 57] that focus on high-level reasoning or dense prediction tasks, such as 3D VQA, grounding, detection, captioning, and segmentation. To address the lack of systematic low-shot evaluation, we additionally assess our model on several low-shot tasks, including viewpoint grounding, scene retrieval, and scene classification, enabling a more comprehensive assessment of the scene representation quality.

3 UniScene3D

UniScene3D is a 3D vision–language pretraining framework that learns unified 3D scene representations from multi-view image–pointmap pairs (Fig. 2). In Sec. 3.1, we describe the model architecture for constructing RGB-Pointmap representations. In Sec. 3.2, we introduce four multimodal alignment objectives that jointly enable robust 3D feature learning.

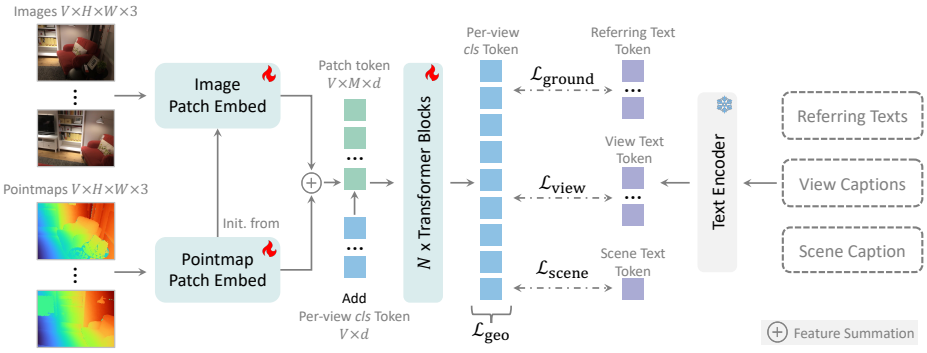

Fig. 2: Overview of UniScene3D pretraining. UniScene3D takes multi-view image–pointmap pairs as input and performs early fusion at the patch embedding stage. The fused tokens, added with absolute positional encodings, are then processed by N Transformer blocks to produce a unified RGB-Pointmap representation. During pretraining, UniScene3D is optimized with four alignment objectives: (1) *Cross-view geometric alignment* $\mathcal{L}_{\text{geo}}$; (2) *Grounded view alignment* $\mathcal{L}_{\text{ground}}$; (3) *View-level alignment* $\mathcal{L}_{\text{view}}$; and (4) *Scene-level alignment* $\mathcal{L}_{\text{scene}}$. Its blocks are highlighted in cyan.

3.1 RGB-Pointmap Representation

To effectively integrate appearance and geometric information, the input to UniScene3D consists of RGB-Pointmap represented as V aligned multi-view image–pointmap pairs $(\mathbf{I}_v, \mathbf{P}_v)_{v=1}^{V}$. For each view v, $\mathbf{I}_v \in \mathbb{R}^{H\times W\times 3}$ denotes the RGB image, and $\mathbf{P}_v \in \mathbb{R}^{H\times W\times 3}$ denotes the pixel-aligned pointmap expressed in the world coordinate frame. Each pointmap $\mathbf{P}_v$ is constructed by back-projecting the corresponding depth map using the camera intrinsics and extrinsics.

Built upon and initialized from the pretrained 2D image encoder of FG-CLIP [52], we introduce an early fusion strategy at the patch embedding stage to encode RGB-Pointmap. After patchification of each view, the image patch embedding layer ϕ_I projects M image patches of $\mathbf{I}_v$ into $\mathbf{z}_v^I \in \mathbb{R}^{M\times d}$, formulated as $\mathbf{z}_v^I = \phi_I(\mathbf{I}_v) + \mathbf{E}_{pos}$, where $\mathbf{E}_{pos} \in \mathbb{R}^{M\times d}$ denotes learnable positional encodings. In parallel, a pointmap patch embedding layer ϕ_P, sharing the same architecture and initialization as ϕ_I, maps geometric patches into $\mathbf{z}_v^P = \phi_P(\mathbf{P}_v) \in \mathbb{R}^{M\times d}$. The two modalities of patch tokens are then fused via element-wise summation, $\mathbf{z}_v = \mathbf{z}_v^I + \mathbf{z}_v^P$. In this manner, appearance and geometric cues are integrated at the earliest representation stage, enabling joint reasoning across modalities while preserving architectural compatibility with the 2D backbone. We compare this approach with other fusion choices in the experimental section.

Then, a learnable class token $\mathbf{z}^{cls} \in \mathbb{R}^d$ is replicated across views as $\mathbf{z}_v^{cls} = \mathbf{z}^{cls}$. The concatenation of the per-view class token and the fused patch tokens $[\mathbf{z}_v^{cls}; \mathbf{z}_v] \in \mathbb{R}^{(M+1)\times d}$ is subsequently processed by N transformer blocks. We take the output per-view class token as per-view representation, denoted as $\mathbf{h}_v$ at view v, then the RGB-Pointmap representation of the entire scene is therefore defined as the set of per-view representation $\{\mathbf{h}_v\}_{v=1}^{V}$.

Table 1: Alignment strategies in UniScene3D. RGBP denotes RGB-Pointmap embeddings. Contrastive pairs are sampled either from views within the same scene or from the entire training batch, including different scenes. Cross-view geometric alignment operates on RGBP–RGBP pairs, while the others perform RGBP–Text alignment. Scene-level alignment uses aggregated view embeddings (Scene embedding); the remaining ones operate on per-view embeddings.

Alignment Strategy	Contrastive Pairs	Modalities	Embedding	Target
Cross-view geometric	Scene	RGBP–RGBP	Per-view	Geometry consistency
Grounded view	Scene	RGBP–Text	Per-view	Object semantics
View-level	Batch	RGBP–Text	Per-view	View semantics
Scene-level	Batch	RGBP–Text	Scene	Scene semantics

3.2 Multimodal Alignment

To pretrain UniScene3D, we employ four multimodal alignment objectives. As summarized in Table 1, cross-view geometric alignment and grounded view alignment construct contrastive pairs within each scene to enforce cross-view consistency among pointmap view embeddings. View-level and scene-level alignments form view and scene pairs across the training batch, encouraging the model to learn discriminative representations across different scenes.

Cross-view Geometric Alignment. Learning effective RGB-Pointmap representations requires accurately encoding the underlying 3D geometry of a scene across viewpoints. An embedding space organized by geometric overlap should place adjacent views closer than distant ones, enabling the representation to reflect spatial relationships within the scene. To this end, we introduce a cross-view geometric alignment objective that enforces embedding similarities to correspond to the spatial proximity of views within the same scene.

Consider two views v and u within a scene, whose pointmaps $\mathbf{P}_v = \{\mathbf{x}_i\}_{i=1}^N$ and $\mathbf{P}_u = \{\mathbf{y}_i\}_{i=1}^N$ each consist of N 3D points, where $N = H \times W$, $\mathbf{x} \in \mathbb{R}^3$ and $\mathbf{y} \in \mathbb{R}^3$. The cross-view geometric dissimilarity is measured using the symmetric Chamfer distance [16]:

$$\mathrm{CD}(\mathbf{P}_v, \mathbf{P}_u) = \frac{1}{N} \sum_{\mathbf{x} \in P_v} \min_{\mathbf{y} \in P_u} \|\mathbf{x} - \mathbf{y}\|_2^2 + \frac{1}{N} \sum_{\mathbf{y} \in P_u} \min_{\mathbf{x} \in P_v} \|\mathbf{y} - \mathbf{x}\|_2^2. \tag{1}$$

For each anchor view v, all other views in the scene are sorted by increasing $\mathrm{CD}(\mathbf{P}_v, \mathbf{P}_u)$ to obtain an ordinal proximity index $r_v(u) \in \{0, 1, \dots\}$, where smaller values indicate stronger geometric proximity. For each anchor view v, the contrastive candidate set consists of all remaining views in the same scene, $\mathcal{V}_S \setminus v$, where $\mathcal{V}_S$ denotes the set of all views within that scene. We construct a rank-aware soft target distribution over this set:

$$p_{v,u}^{\mathrm{soft}} = \frac{\exp(-r_v(u)/\tau_r)}{\sum_{k \in \mathcal{V}_S \setminus \{v\}} \exp(-r_v(k)/\tau_r)}, \tag{2}$$

where $\tau_r > 0$ controls the concentration of probability mass over geometrically closer views. To stabilize optimization, this distribution is interpolated with a

hard nearest-neighbor target $p^{\text{hard}}_{v,u}$:

$$p_{v,u} = \alpha\, p^{\text{hard}}_{v,u} + (1-\alpha)\, p^{\text{soft}}_{v,u}, \tag{3}$$

where $p^{\text{hard}}_{v,u} = 1$ if $u = \arg\min_{k \neq v} \text{CD}(\mathbf{P}_v, \mathbf{P}_k)$ and 0 otherwise. The interpolation coefficient $\alpha \in [0,1]$ controls the contribution of the hard and soft targets: when $\alpha = 1$, the objective reduces to hard nearest-neighbor contrastive learning, whereas $\alpha = 0$ yields a fully rank-aware geometric alignment formulation. We define cross-view similarity logits as $\ell_{v,u} = \mathbf{h}_v^\top \mathbf{h}_u / \tau$, where $\tau > 0$ is a learnable temperature parameter. The geometric alignment loss is then defined as the soft-label cross-entropy over the candidate set:

$$\mathcal{L}_{\text{geo}} = -\sum_{v} \sum_{u \in \mathcal{V}_S \setminus \{v\}} p_{v,u} \log \frac{\exp(\ell_{v,u})}{\sum_{k \in \mathcal{V}_S \setminus \{v\}} \exp(\ell_{v,k})}. \tag{4}$$

Grounded View Alignment. While cross-view geometric alignment enforces spatial consistency across viewpoints, RGB-Pointmap also exhibit cross-view semantic relationships because different views may observe the same objects. To capture this property, we introduce grounded view alignment, which aligns object referring text embeddings with all per-view RGB-Pointmap embeddings that observe the referred objects. This objective encourages views sharing the same object semantics to be closer in the representation space.

Let $\mathbf{t}_o \in \mathbb{R}^d$ denote the embedding of the referring text associated with a grounded object o. Positive view–object pairs are determined based on geometric visibility. Specifically, for each object o, we examine whether there is an intersection between each view 3D pointmap $\mathbf{P}_v$ and the object's 3D mask from the pretrained SceneVerse [25] dataset. If the intersection exists, view v is considered to observe object o, and the pair (v, o) is included in the set of valid positives $\mathcal{P}$. A single scene may thus produce multiple valid view–object correspondences.

For each $(v, o) \in \mathcal{P}$, we compute cross-modal similarity logits $s_{v,o} = \mathbf{h}_v^\top \mathbf{t}_o / \tau$. The grounded view alignment loss is computed over all valid positive pairs:

$$\mathcal{L}_{\text{ground}} = \frac{1}{2|\mathcal{P}|} \sum_{(v,o) \in \mathcal{P}} \left[-\log \frac{\exp(s_{v,o})}{\sum_{o' \in \mathcal{O}_S} \exp(s_{v,o'})} - \log \frac{\exp(s_{v,o})}{\sum_{v' \in \mathcal{V}_S} \exp(s_{v',o})} \right], \tag{5}$$

where $\mathcal{O}_S$ and $\mathcal{V}_S$ denote the sets of grounded objects and views in a scene.

View-level and Scene-level Alignment. We further follow the alignment strategy of [38] to align the 3D encoder with the FG-CLIP [52] text encoder at both view and scene levels using all scenes within the training batch.

For view-level alignment, given N_v views in a batch, the similarity logit between the RGB-Pointmap embedding $\mathbf{h}_v$ of view v and the corresponding view caption embedding $\mathbf{t}^V_u$ of view u is defined as $k_{v,u} = \mathbf{h}_v^\top \mathbf{t}^V_u / \tau$. Only matched pointmap–caption pairs are treated as positives, while all other combinations within the batch serve as negatives. The view-level alignment objective can be

formulated as:

$$\mathcal{L}_{\text{view}} = \frac{1}{2N_v} \sum_{v=1}^{N_v} \left[-\log \frac{\exp(k_{v,v})}{\sum_{u=1}^{N_v} \exp(k_{v,u})} - \log \frac{\exp(k_{v,v})}{\sum_{u=1}^{N_v} \exp(k_{u,v})} \right]. \tag{6}$$

For scene-level alignment, the per-view RGB-Pointmap embeddings within a scene $\{\mathbf{h}_v\}_{v=1}^{N}$ are mean-pooled to obtain a scene-level embedding $\bar{\mathbf{h}}_s$. Given N_s scenes in a batch, the similarity logit between the scene-level embedding $\bar{\mathbf{h}}_i$ of scene i and the scene caption embedding $\mathbf{t}_j^S$ of scene j is defined as $m_{i,j} = \bar{\mathbf{h}}_i^\top \mathbf{t}_j^S / \tau$. We align $\bar{\mathbf{h}}_i$ with its corresponding caption $\mathbf{t}_j^S$ using the following objective:

$$\mathcal{L}_{\text{scene}} = \frac{1}{2N_s} \sum_{i=1}^{N_s} \left[-\log \frac{\exp(m_{i,i})}{\sum_{j=1}^{N_s} \exp(m_{i,j})} - \log \frac{\exp(m_{i,i})}{\sum_{j=1}^{N_s} \exp(m_{j,i})} \right]. \tag{7}$$

Total Loss. The total loss for UniScene3D pretraining is defined as below:

$$\mathcal{L}_{\text{total}} = \lambda \mathcal{L}_{\text{geo}} + \mathcal{L}_{\text{ground}} + \mathcal{L}_{\text{view}} + \mathcal{L}_{\text{scene}}. \tag{8}$$

where $\lambda = 0.1$ in our setting. Together, these four alignment losses encourage the learning of geometrically consistent and semantically rich RGB-Pointmap features across multiple scales.

4 Experiments

4.1 Experimental Setting

Evaluation Tasks. We evaluate the representation quality of UniScene3D on diverse 3D scene understanding tasks, including viewpoint grounding, scene retrieval, scene type classification, and 3D visual question answering (3D VQA). Viewpoint grounding assesses fine-grained spatial understanding by requiring the model to identify the view that best matches a referring expression. Scene retrieval aims to retrieve the correct 3D scene from descriptions at different levels of granularity. Scene type classification categorizes indoor environments into room types. These tasks are evaluated in zero-shot or few-shot settings to measure representation quality under minimal supervision. Such low-shot evaluation is common for vision-language encoders [43, 49]. 3D visual question answering (3D VQA) evaluates a model's ability to answer natural language questions about a 3D scene by reasoning over its spatial layout, object relationships, and semantic attributes. We evaluate UniScene3D on this task under task-specific fine-tuning to examine how effectively it serves as a backbone for downstream 3D models.
Datasets. We pretrain UniScene3D on 6,562 indoor scenes from ScanNet [10], 3RScan [50], and ARKitScenes [4]. For grounded view alignment, we use LLM-generated referring expressions from SceneVerse [25] and adopt LLM-generated view-level and scene-level captions from POMA-3D [38].

Table 2: Viewpoint grounding results on ScanRefer, Nr3D, and Sr3D. Input modalities include multi-view image (I), Point cloud (PC), Pointmap (PM), and RGB-Pointmap (RGBP). R@N (Recall@N) measures the percentage of cases where the correct view is ranked within the top N candidates retrieved. All methods are evaluated in the zero-shot setting. The best results are in **bold**.

Method	Input	ScanRefer			Nr3D			Sr3D		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
DFN [17]	I	20.9	59.2	77.6	17.3	51.8	71.8	12.6	43.5	65.6
SigLIP2 [49]	I	22.2	61.1	79.5	18.8	54.3	74.0	13.0	44.4	67.2
FG-CLIP [52]	I	20.2	58.5	77.6	17.3	52.6	72.8	13.4	44.5	66.6
Uni3D-g [63]	PC	4.2	19.0	35.5	3.6	17.2	33.4	3.4	16.5	32.2
POMA-3D [38]	PM	16.4	50.6	71.2	13.6	44.5	65.9	10.2	37.4	59.2
UniScene3D	RGBP	**38.6**	**72.5**	**85.7**	**25.2**	**64.0**	**80.9**	**23.6**	**62.8**	**80.4**

For evaluation, we reformulate existing visual grounding benchmarks, including ScanRefer [6], Nr3D, and Sr3D [1], to support viewpoint grounding and scene retrieval. Scene type classification is performed over 21 ScanNet categories (e.g., living room and bedroom). For 3D VQA, we evaluate on ScanQA [3] for commonsense spatial reasoning, SQA3D [36] for situated reasoning, and Hypo3D [39] for hypothetical reasoning. We follow the standard train–test splits used in prior work [25, 38, 66]. All evaluation code and datasets will be publicly released.

Baselines. We compare UniScene3D with diverse state-of-the-art perception encoders spanning different input modalities for 3D representation learning. For image-based methods, we include DFN [17], SigLIP2 [49], and FG-CLIP [52], which take multi-view images as input. All models use ViT-B/16 checkpoints pretrained at 224×224 resolution, matching the architecture of UniScene3D. For point cloud-based approaches, we evaluate Uni3D-g (Giant) [63], 3D-VisTA [66], and SceneVerse [25]. As Uni3D-g is pretrained on small object-level point clouds (10,000 points), directly applying it to full-scene point clouds degrades performance. We therefore convert multi-view pointmaps into per-view point clouds, extract view-level embeddings using Uni3D-g, and aggregate them into a scene representation. For pointmap-based methods, we compare with POMA-3D [38], which is most closely related to our approach.

4.2 Implementation Details.

We implement UniScene3D in PyTorch and train it on NVIDIA A100 GPUs. The model is pretrained for 80 epochs with a batch size of 64, using the ViT-B/16 variant of FG-CLIP [52] as the backbone. We optimize the model using AdamW [35] with $\beta_1 = 0.9$ and $\beta_2 = 0.98$. The learning rate is initialized at 1×10^{-4} and decayed to a minimum of 1×10^{-5} following a cosine schedule. For each 3D scene, we sample 32 image–pointmap pairs using the maximum coverage sampling strategy from Video-3D-LLM [61] and resize them to 224×224. This strategy selects a limited set of views that maximally cover the scene. For cross-view geometric alignment, we set $\alpha = 0.7$ and $\tau_r = 0.35$. See the Appendix for more details about downstream evaluation.

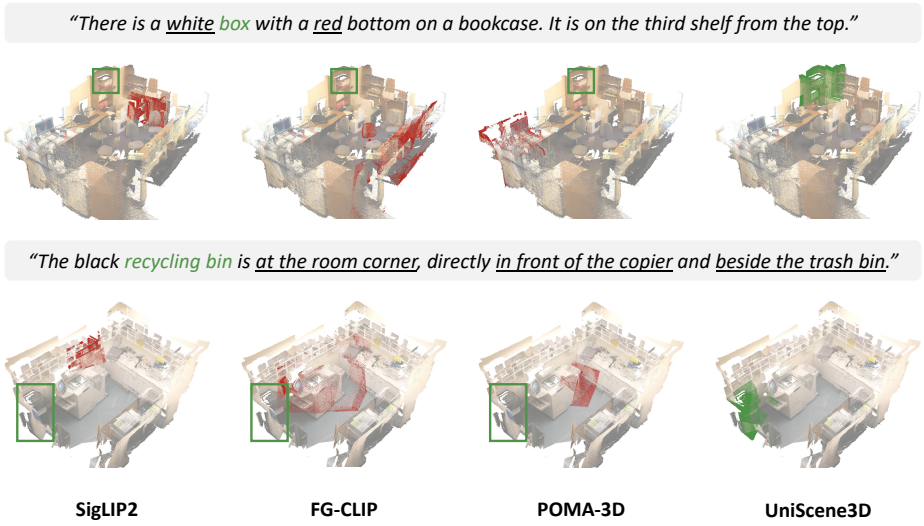

Fig. 3: Qualitative viewpoint grounding results. Correct and incorrect matches are highlighted in green and red, respectively, based on the referring texts. Underlined phrases denote contextual clues that guide the models in solving the grounding task.

4.3 Viewpoint Grounding

Setting. Given a referring expression describing an object, viewpoint grounding aims to identify the view within a 3D scene that best observes the referred object. We reformulate existing 3D visual grounding benchmarks by converting object-level annotations (object IDs and 3D bounding boxes) into view-level labels. Since an object may appear in multiple views, the ground-truth view is defined as the one where the object occupies the largest visible area. During evaluation, embeddings are computed for all views in a scene and compared with the referring text embedding. A prediction is considered correct if the most similar view matches the ground-truth view. Performance is reported using Recall@1 (R@1), Recall@5 (R@5), and Recall@10 (R@10). R@5 and R@10 additionally reflect how well the model identifies coarse viewpoints of the referred object.

Results. As shown in Table 2, UniScene3D consistently outperforms all baselines across metrics, achieving R@1 gains of at least 15% on ScanRefer, 8% on Nr3D, and 10% on Sr3D. In contrast, Uni3D-g performs worst, suggesting that object-level point cloud pretraining transfers poorly to scene-level reasoning and highlighting the need for 3D encoders designed specifically for scene understanding. Notably, the prior pointmap-based method POMA-3D underperforms image-based encoders, despite the task emphasizing 3D grounding and inter-object spatial reasoning that largely depend on geometric cues rather than appearance. UniScene3D nearly doubles the R@1 of POMA-3D on most bench-

Table 3: Scene retrieval results on ScanRefer, Nr3D, and Sr3D. We report results where each scene caption contains $n = 5$ or 10 utterances. All methods are evaluated in the zero-shot setting.

Method	Input	ScanRefer				Nr3D				Sr3D			
		$n = 5$		$n = 10$		$n = 5$		$n = 10$		$n = 5$		$n = 10$	
		R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5
DFN [17]	I	4.6	13.5	4.6	13.3	3.9	12.7	4.4	13.0	0.6	3.1	0.6	3.2
SigLIP2 [49]	I	1.5	5.3	1.8	6.6	0.7	3.0	1.0	3.8	0.1	0.3	0.1	0.4
FG-CLIP [52]	I	9.2	26.3	16.3	40.0	6.2	20.8	12.4	32.8	0.6	2.7	0.7	3.8
Uni3D-g [63]	PC	0.3	1.4	0.3	1.3	0.5	1.4	0.4	1.5	0.3	1.6	0.2	1.4
3D-VisTA [66]	PC	0.6	2.4	0.7	2.2	0.1	0.9	0.2	1.3	0.2	1.5	0.2	1.7
SceneVerse [25]	PC	0.5	2.1	0.6	2.7	0.5	1.3	0.5	1.4	0.4	1.8	0.1	1.9
POMA-3D [38]	PM	13.8	34.9	20.4	47.2	11.9	32.7	19.3	46.8	1.6	6.9	2.4	9.5
UniScene3D	RGBP	**22.4**	**48.6**	**33.4**	**62.9**	**19.7**	**46.3**	**30.7**	**61.7**	**3.0**	**11.8**	**4.6**	**17.5**

marks, indicating that our training strategy more effectively leverages geometric information from pointmaps for spatial reasoning.

The qualitative results in Fig. 3 further support these findings. UniScene3D successfully retrieves the correct views in large, cluttered scenes requiring either complex appearance reasoning (top) or geometric reasoning (bottom).

4.4 Scene Retrieval

Setting. The scene retrieval task aims to retrieve the scene that best matches a given caption by computing the similarity between scene embeddings and the caption embedding. Scene-level baselines (3D-VisTA and SceneVerse) directly produce scene embeddings, while for view-based methods, we obtain scene embeddings by mean-pooling view-level embeddings. Following POMA-3D [38], scene captions are constructed by concatenating referring texts from 3D visual grounding datasets. We evaluate under two settings where each caption contains ($n = 5$) or ($n = 10$) texts, and report R@1 and R@5.

Results. Table 3 demonstrates that UniScene3D achieves a clear performance advantage on this task, with results further improving as longer and more detailed scene captions are provided. In contrast, point cloud-based methods perform poorly, sometimes even falling behind image-based encoders such as DFN and FG-CLIP. This observation supports our assumption that training point cloud encoders from scratch limits their ability to exploit rich 2D foundation priors and increases the risk of overfitting to training descriptions. By contrast, pointmap-based methods that are fine-tuned from pretrained 2D encoders, including POMA-3D and UniScene3D, exhibit stronger generalization.

4.5 Scene Type Classification

Setting. Scene type classification evaluates a model's ability to predict the correct category of a 3D scene. We assess this task in both zero-shot and few-shot settings. In the zero-shot setting, scene embeddings are matched against

Table 4: Scene type classification accuracy (%). Linear probing is used for 1/5/10-shot evaluation. Our method consistently outperforms other methods.

Method	Input	0-shot	1-shot	5-shot	10-shot
DFN [17]	I	41.6	40.5	60.1	77.5
SigLIP2 [49]	I	36.0	36.6	57.8	77.9
FG-CLIP [52]	I	49.5	40.0	62.0	77.5
Uni3D-g [63]	PC	8.6	21.4	37.1	45.4
3D-VisTA [66]	PC	1.9	28.9	49.4	55.0
SceneVerse [25]	PC	0.8	24.0	51.1	64.2
POMA-3D [38]	PM	63.9	32.2	64.9	74.1
UniScene3D	RGBP	**70.7**	**43.2**	**72.4**	**83.7**

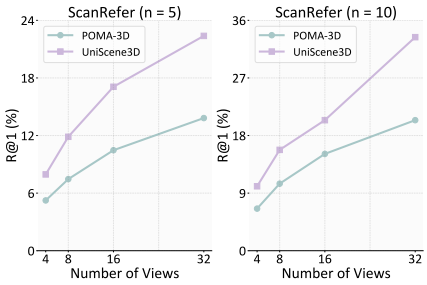

Fig. 4: Effect of view number on scene retrieval (ScanRefer). R@1 is reported under $n = 5$ and $n = 10$.

Table 5: 3D VQA results on ScanQA, SQA3D, and Hypo3D. We report Exact Match (EM@1/10) metrics. Higher value is better for all metrics.

Method	Input	ScanQA		SQA3D		Hypo3D	
		EM@1	EM@10	EM@1	EM@10	EM@1	EM@10
FG-CLIP [52]	I	20.9	49.9	49.5	89.7	31.1	82.1
3D-VisTA [66]	PC	22.4	52.1	48.5	85.6	31.0	81.2
SceneVerse [25]	PC	22.7	51.5	49.9	85.0	31.6	80.3
POMA-3D [38]	PM	22.3	52.3	51.1	91.2	33.4	84.8
UniScene3D	RGBP	**23.2**	**53.5**	**52.5**	**92.3**	**35.2**	**85.9**

21 candidate scene types using the prompt template "This room is a {}." A prediction is correct if the scene embedding has the highest similarity to the corresponding scene-type prompt. In the few-shot setting, we perform linear probing with L-BFGS [33] using N labeled examples per class (N shots) and evaluate on the remaining unseen scenes.

Results. Table 4 shows that UniScene3D ranks first across all settings. Point cloud–based methods perform the worst. Surprisingly, scene-level point cloud methods even underperform the object-level model Uni3D in the zero-shot setting. This may be because Uni3D is pretrained on much larger object-level point clouds than those available for scene-level models, leading to weaker zero-shot generalization for scene-level approaches. Other baselines exhibit fluctuating performance and only excel under specific supervision regimes. For example, POMA-3D performs best among baselines in the 0- and 5-shot settings but falls behind image-based methods under 1- and 10-shot evaluation. DFN achieves the strongest 1-shot result, while SigLIP2 performs best at 10-shot. Such variability suggests limited robustness across supervision levels. In contrast, UniScene3D maintains consistently high performance across all settings.

4.6 3D Visual Question Answering

Setting. Following prior work [25, 38], we evaluate 3D VQA by attaching a shallow QA head and a BERT [27] language encoder. The visual encoder is

Table 6: Effect of initialization and pretraining on viewpoint grounding (ScanRefer). 2D priors benefit pointmap learning.

Initialization	w/o Pretrain		w/ Pretrain	
	R@1	R@5	R@1	R@5
Random	4.32	17.5	7.53	24.38
FG-CLIP [52]	**5.03**	**21.1**	**37.62**	**71.53**

Table 7: Ablations of image-pointmap (PM) fusion on viewpoint grounding (ScanRefer). Init. denotes initialization.

Case	R@1	R@5
UniScene3D	**38.6**	**72.5**
Random Init. PM Patch Embed	38.0	71.8
w/o Positional Encoding	37.2	71.4
Concat + Projection	36.5	70.0

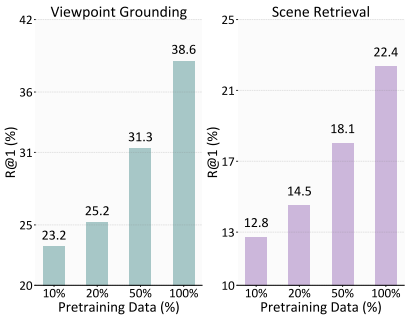

Fig. 5: Effect of pretraining data scale on viewpoint grounding and scene retrieval (ScanRefer, R@1, $n = 5$). Performance improves consistently with more pretraining data.

Table 8: Ablation study of UniScene3D on viewpoint grounding, scene retrieval (ScanRefer), and scene type classification. We compare variants with individual components removed against the full model.

Case	Viewpoint Grounding						Scene Retrieval				Scene Type	
	ScanRefer		Nr3D		Sr3D		$n = 5$		$n = 10$		Classification	
	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5	R@1	R@5	0-shot	10-shot
UniScene3D	**38.6**	**72.5**	**25.2**	**64.0**	**23.6**	**62.8**	**22.4**	**48.6**	**33.4**	**62.9**	**70.7**	**83.7**
w/o image	37.6	71.7	23.7	62.7	23.1	62.0	21.2	45.7	31.2	60.5	67.2	83.6
w/o pointmap	28.8	66.5	21.3	58.3	18.5	54.9	21.8	48.0	32.0	61.9	68.9	82.6
w/o $\mathcal{L}_{\text{geo}}$	38.2	71.5	23.2	62.0	23.0	62.3	21.6	47.3	32.4	61.8	68.7	81.3
w/o $\mathcal{L}_{\text{ground}}$	18.5	56.3	15.6	49.7	11.5	41.5	21.8	47.6	30.6	63.0	68.5	82.4

frozen, while only the language encoder and QA head are fine-tuned. We report Exact Match at top-1 and top-10 (EM@1, EM@10) as evaluation metrics.

Results. Consistent with the findings on other downstream tasks, Table 5 shows that UniScene3D consistently outperforms all 3D encoder baselines across the three benchmarks. Notably, POMA-3D underperforms image-based methods on ScanQA but surpasses them on SQA3D and Hypo3D. This discrepancy may stem from the fact that ScanQA contains many questions involving color and texture cues. By incorporating image appearance information, UniScene3D effectively enhances pointmap-based representations on this benchmark.

4.7 Ablation Study

2D Foundation Priors Benefit Pointmap Learning. Our work hypothesizes that the grid structure of pointmaps enables effective transfer of pretrained 2D priors. To validate this, we construct a pointmap-only baseline using the same ViT backbone as FG-CLIP, initialized with either random weights or FG-CLIP pretrained weights. On viewpoint grounding (ScanRefer), the pretrained model achieves consistent improvement over random initialization, even without further

training on 3D scenes. After applying the full UniScene3D training strategy, the performance gap widens further. These results provide direct evidence that 2D foundation priors significantly improve pointmap representation learning, both as a standalone encoder and as a strong initialization for subsequent training.

Effect of RGB-Pointmap Representation. Table 8 compares UniScene3D with two ablated variants: a pointmap-only model (w/o image input) and an image-only model (w/o pointmap input), both removing the early token fusion mechanism and reducing the architecture to a standard ViT. UniScene3D consistently outperforms these baselines across all zero-shot and few-shot tasks, demonstrating that the proposed RGB-Pointmap representation effectively integrates complementary geometric and appearance cues to learn stronger unified 3D scene representations. Qualitative examples in Fig. 1 further illustrate this advantage: the pointmap-only variant fails to identify green seats due to missing color cues, while the image-only variant cannot recognize the longest seating area without geometric information. These results highlight the necessity of jointly modeling appearance and geometry for robust 3D scene understanding.

We further compare our early token fusion strategy with alternative design choices in Table 7. When the patch embedding is randomly initialized instead of using the pretrained image patch embedding layer, or when the positional encoding is removed, performance drops noticeably. This validates that preserving pretrained 2D weights is important for effective RGB-Pointmap learning. We also evaluate a simple fusion strategy that concatenates image and pointmap inputs, followed by a projection layer to match the encoder dimension. This strategy performs substantially worse than our design and even underperforms the w/o image setting in Table 8. This may be because the projection layer has to be trained from scratch, which harms the generalization of the fused features.

Effect of Training Strategy. Table 8 shows that both cross-view geometric alignment and grounded view alignment contribute to learning effective features in UniScene3D across downstream tasks. Cross-view geometric alignment enforces global geometric consistency and, even without explicit semantic alignment with text, still yields consistent improvements on cross-modal grounding and retrieval tasks. Grounded view alignment further enhances the model's grounding capability, producing particularly strong gains on the viewpoint grounding task. These results suggest that effective pointmap embeddings should capture both geometric structure and semantic relationships within a scene.

Number of Views Analysis. We further show that UniScene3D is robust to the number of input views in Fig. 4, consistently outperforming POMA-3D across all settings. Notably, using only 16 views already surpasses POMA-3D with 32 views, indicating that our pretraining does not bias the model toward a specific number of viewpoints and supports broader applicability.

Data Scaling Analysis. Fig. 5 shows that UniScene3D scales positively with more training scenes, with performance improving steadily as the dataset size increases. Training on the full dataset clearly outperforms using only half of the data. While UniScene3D benefits from pretrained 2D priors, it remains data-hungry. We expect further gains with larger-scale real-world pretraining data.

5 Limitation

Although UniScene3D learns effective 3D representations for a wide range of downstream tasks, it still has several limitations. First, UniScene3D is pretrained with a fixed input resolution of image–pointmap pairs and a ViT-B/16 backbone. Pretraining with varied input resolutions and model sizes would be necessary for UniScene3D to scale as a general-purpose 3D representation framework across diverse scenarios. Second, UniScene3D is trained on a limited number of scenes. Although it benefits from strong 2D foundation model priors, larger-scale pretraining on more diverse 3D scenes could further improve its effectiveness.

6 Conclusion

In this paper, we present UniScene3D, a 3D representation learning method that jointly models geometry and appearance from multi-view RGB-Pointmap. Leveraging the rich cross-view context information pointmaps, we introduce cross-view geometric alignment and grounded view alignment objectives for pretraining. Experiments show that UniScene3D achieves state-of-the-art performance across multiple 3D scene understanding benchmarks while remaining robust to varying numbers of input views. In future work, we plan to scale UniScene3D to larger model sizes, similar to 2D CLIP models, and explore its use as a general vision backbone for building 3D foundation models.

References

1. Achlioptas, P., Abdelreheem, A., Xia, F., Elhoseiny, M., Guibas, L.: Referit3d: Neural listeners for fine-grained 3d object identification in real-world scenes. In: European conference on computer vision. pp. 422–440. Springer (2020) 9
2. Arnaud, S., McVay, P., Martin, A., Majumdar, A., Jatavallabhula, K.M., Thomas, P., Partsey, R., Dugas, D., Gejji, A., Sax, A., et al.: Locate 3d: Real-world object localization via self-supervised learning in 3d. arXiv preprint arXiv:2504.14151 (2025) 3
3. Azuma, D., Miyanishi, T., Kurita, S., Kawanabe, M.: Scanqa: 3d question answering for spatial scene understanding. In: proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 19129–19139 (2022) 2, 4, 9
4. Baruch, G., Chen, Z., Dehghan, A., Dimry, T., Feigin, Y., Fu, P., Gebauer, T., Joffe, B., Kurz, D., Schwartz, A., et al.: Arkitscenes: A diverse real-world dataset for 3d indoor scene understanding using mobile rgb-d data. arXiv preprint arXiv:2111.08897 (2021) 4, 8
5. Bi, Z., Wang, L.: Advances in 3d data acquisition and processing for industrial applications. Robotics and Computer-Integrated Manufacturing **26**(5), 403–413 (2010) 4
6. Chen, D.Z., Chang, A.X., Nießner, M.: Scanrefer: 3d object localization in rgb-d scans using natural language. In: European conference on computer vision. pp. 202–221. Springer (2020) 4, 9

7. Chen, J., Barath, D., Armeni, I., Pollefeys, M., Blum, H.: "where am i?" scene retrieval with language. In: European Conference on Computer Vision. pp. 201–220. Springer (2024) 3
8. Chen, R., Liu, Y., Kong, L., Zhu, X., Ma, Y., Li, Y., Hou, Y., Qiao, Y., Wang, W.: Clip2scene: Towards label-efficient 3d scene understanding by clip. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7020–7030 (2023) 2, 3
9. Chen, Z., Gholami, A., Nießner, M., Chang, A.X.: Scan2cap: Context-aware dense captioning in rgb-d scans. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3193–3203 (2021) 4
10. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5828–5839 (2017) 4, 8
11. Deitke, M., VanderBilt, E., Herrasti, A., Weihs, L., Ehsani, K., Salvador, J., Han, W., Kolve, E., Kembhavi, A., Mottaghi, R.: Procthor: Large-scale embodied ai using procedural generation. Advances in Neural Information Processing Systems **35**, 5982–5994 (2022) 4
12. Ding, R., Yang, J., Xue, C., Zhang, W., Bai, S., Qi, X.: Pla: Language-driven open-vocabulary 3d scene understanding. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7010–7019 (2023) 3
13. Dong, J., Wu, T., Qian, R., Wang, J.: Simc3d: A simple contrastive 3d pretraining framework using rgb images. arXiv preprint arXiv:2412.05274 (2024) 2
14. Duan, J., Yu, S., Tan, H.L., Zhu, H., Tan, C.: A survey of embodied ai: From simulators to research tasks. IEEE Transactions on Emerging Topics in Computational Intelligence **6**(2), 230–244 (2022) 2
15. Fan, G., Qi, Z., Shi, W., Ma, K.: Point-gcc: Universal self-supervised 3d scene pre-training via geometry-color contrast. In: Proceedings of the 32nd ACM International Conference on Multimedia. pp. 4709–4718 (2024) 2
16. Fan, H., Su, H., Guibas, L.J.: A point set generation network for 3d object reconstruction from a single image. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 605–613 (2017) 6
17. Fang, A., Jose, A.M., Jain, A., Schmidt, L., Toshev, A., Shankar, V.: Data filtering networks. arXiv preprint arXiv:2309.17425 (2023) 3, 4, 9, 11, 12
18. Feng, T., Wang, W., Quan, R., Yang, Y.: Shape2scene: 3d scene representation learning through pre-training on shape data. In: European conference on computer vision. pp. 73–91. Springer (2024) 4
19. Gao, Y., Wang, Z., Zheng, W.S., Xie, C., Zhou, Y.: Mixcon3d: Synergizing multi-view and cross-modal contrastive learning for enhancing 3d representation (2023) 3
20. Guo, Z., Zhang, R., Zhu, X., Tang, Y., Ma, X., Han, J., Chen, K., Gao, P., Li, X., Li, H., et al.: Point-bind & point-llm: Aligning point cloud with multi-modality for 3d understanding, generation, and instruction following. arXiv preprint arXiv:2309.00615 (2023) 3
21. He, Y., Yu, H., Liu, X., Yang, Z., Sun, W., Anwar, S., Mian, A.: Deep learning based 3d segmentation: A survey. arXiv preprint arXiv:2103.05423 (2021) 2
22. Hou, J., Graham, B., Nießner, M., Xie, S.: Exploring data-efficient 3d scene understanding with contrastive scene contexts. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 15587–15597 (2021) 4
23. Huang, R., Luo, W., Mao, Y., Mikolajczyk, K.: From none to all: Self-supervised 3d reconstruction via novel view synthesis. arXiv preprint arXiv:2603.27455 (2026) 4

24. Huang, T., Dong, B., Yang, Y., Huang, X., Lau, R.W., Ouyang, W., Zuo, W.: Clip2point: Transfer clip to point cloud classification with image-depth pre-training. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 22157–22167 (2023) 3
25. Jia, B., Chen, Y., Yu, H., Wang, Y., Niu, X., Liu, T., Li, Q., Huang, S.: Sceneverse: Scaling 3d vision-language learning for grounded scene understanding. In: European Conference on Computer Vision. pp. 289–310. Springer (2024) 3, 4, 7, 8, 9, 11, 12
26. Jiao, S., Dong, H., Yin, Y., Jie, Z., Qian, Y., Zhao, Y., Shi, H., Wei, Y.: Clip-gs: Unifying vision-language representation with 3d gaussian splatting. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4670–4680 (2025) 3
27. Koukounas, A., Mastrapas, G., Eslami, S., Wang, B., Akram, M.K., Günther, M., Mohr, I., Sturua, S., Wang, N., Xiao, H.: jina-clip-v2: Multilingual multimodal embeddings for text and images. arXiv preprint arXiv:2412.08802 (2024) 12
28. Lee, H.H., Zhang, Y., Chang, A.X.: Duoduo clip: Efficient 3d understanding with multi-view images. arXiv preprint arXiv:2406.11579 (2024) 2, 3
29. Li, W., Zhao, Y., Qin, M., Liu, Y., Cai, Y., Gan, C., Pfister, H.: Langsplatv2: High-dimensional 3d language gaussian splatting with 450+ fps. arXiv preprint arXiv:2507.07136 (2025) 3
30. Li, Y., Ma, Q., Yang, R., Li, H., Ma, M., Ren, B., Popovic, N., Sebe, N., Konukoglu, E., Gevers, T., et al.: Scenesplat: Gaussian splatting-based scene understanding with vision-language pretraining. arXiv preprint arXiv:2503.18052 (2025) 3
31. Li, Z., Yu, H., Ding, Y., Li, Y., He, Y., Akhtar, N.: Embodied intelligence for 3d understanding: A survey on 3d scene question answering. arXiv preprint arXiv:2502.00342 (2025) 2
32. Liao, G., Li, J., Bao, Z., Ye, X., Li, Q., Liu, K.: Clip-gs: Clip-informed gaussian splatting for view-consistent 3d indoor semantic understanding. ACM Transactions on Multimedia Computing, Communications and Applications **21**(8), 1–24 (2025) 3
33. Liu, D.C., Nocedal, J.: On the limited memory bfgs method for large scale optimization. Mathematical programming **45**(1), 503–528 (1989) 12
34. Liu, M., Shi, R., Kuang, K., Zhu, Y., Li, X., Han, S., Cai, H., Porikli, F., Su, H.: Openshape: Scaling up 3d shape representation towards open-world understanding. Advances in neural information processing systems **36**, 44860–44879 (2023) 2, 3, 4
35. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017) 9
36. Ma, X., Yong, S., Zheng, Z., Li, Q., Liang, Y., Zhu, S.C., Huang, S.: Sqa3d: Situated question answering in 3d scenes. arXiv preprint arXiv:2210.07474 (2022) 4, 9
37. Mao, Y., Jing, J., Mikolajczyk, K.: Opendlign: Open-world point cloud understanding with depth-aligned images. Advances in Neural Information Processing Systems **37**, 101144–101167 (2024) 2, 3, 4
38. Mao, Y., Luo, W., Huang, R., Jing, J., Mikolajczyk, K.: Poma-3d: The point map way to 3d scene understanding. arXiv preprint arXiv:2511.16567 (2025) 2, 4, 7, 8, 9, 11, 12
39. Mao, Y., Luo, W., Jing, J., Qiu, A., Mikolajczyk, K.: Hypo3d: Exploring hypothetical reasoning in 3d. arXiv preprint arXiv:2502.00954 (2025) 4, 9
40. Mao, Y., Zhang, Y., Jiang, H., Chang, A., Savva, M.: Multiscan: Scalable rgbd scanning for 3d environments with articulated objects. Advances in neural information processing systems **35**, 9058–9071 (2022) 4

41. Nguyen, K., Hassan, G.M., Mian, A.: Occlusion-aware text-image-point cloud pretraining for open-world 3d object recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16965–16975 (2025) 3
42. Qin, M., Li, W., Zhou, J., Wang, H., Pfister, H.: Langsplat: 3d language gaussian splatting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20051–20060 (2024) 3
43. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PmLR (2021) 2, 3, 4, 8
44. Ramakrishnan, S.K., Gokaslan, A., Wijmans, E., Maksymets, O., Clegg, A., Turner, J., Undersander, E., Galuba, W., Westbury, A., Chang, A.X., et al.: Habitat-matterport 3d dataset (hm3d): 1000 large-scale 3d environments for embodied ai. arXiv preprint arXiv:2109.08238 (2021) 4
45. Sarkar, S.D., Miksik, O., Pollefeys, M., Barath, D., Armeni, I.: Crossover: 3d scene cross-modal alignment. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8985–8994 (2025) 3
46. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4104–4113 (2016) 4
47. Sun, Q., Fang, Y., Wu, L., Wang, X., Cao, Y.: Eva-clip: Improved training techniques for clip at scale. arXiv preprint arXiv:2303.15389 (2023) 3, 4
48. Tchapmi, L., Choy, C., Armeni, I., Gwak, J., Savarese, S.: Segcloud: Semantic segmentation of 3d point clouds. In: 2017 international conference on 3D vision (3DV). pp. 537–547. IEEE (2017) 2
49. Tschannen, M., Gritsenko, A., Wang, X., Naeem, M.F., Alabdulmohsin, I., Parthasarathy, N., Evans, T., Beyer, L., Xia, Y., Mustafa, B., et al.: Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. arXiv preprint arXiv:2502.14786 (2025) 2, 3, 4, 8, 9, 11, 12
50. Wald, J., Avetisyan, A., Navab, N., Tombari, F., Nießner, M.: Rio: 3d object instance re-localization in changing indoor environments. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7658–7667 (2019) 4, 8
51. Wang, X., Ma, W., Li, Z., Kortylewski, A., Yuille, A.L.: 3d-aware visual question answering about parts, poses and occlusions. Advances in Neural Information Processing Systems **36**, 58717–58735 (2023) 2
52. Xie, C., Wang, B., Kong, F., Li, J., Liang, D., Zhang, G., Leng, D., Yin, Y.: Fg-clip: Fine-grained visual and textual alignment. arXiv preprint arXiv:2505.05071 (2025) 5, 7, 9, 11, 12, 13
53. Xue, L., Gao, M., Xing, C., Martín-Martín, R., Wu, J., Xiong, C., Xu, R., Niebles, J.C., Savarese, S.: Ulip: Learning a unified representation of language, images, and point clouds for 3d understanding. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1179–1189 (2023) 2, 3, 4
54. Xue, L., Yu, N., Zhang, S., Panagopoulou, A., Li, J., Martín-Martín, R., Wu, J., Xiong, C., Xu, R., Niebles, J.C., et al.: Ulip-2: Towards scalable multimodal pretraining for 3d understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 27091–27101 (2024) 3

55. Yang, J., Ding, R., Deng, W., Wang, Z., Qi, X.: Regionplc: Regional point-language contrastive learning for open-world 3d scene understanding. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 19823–19832 (2024) 3
56. Yao, Y., Luo, Z., Li, S., Zhang, J., Ren, Y., Zhou, L., Fang, T., Quan, L.: Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1790–1799 (2020) 4
57. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: Scannet++: A high-fidelity dataset of 3d indoor scenes. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12–22 (2023) 4
58. Zeng, Y., Jiang, C., Mao, J., Han, J., Ye, C., Huang, Q., Yeung, D.Y., Yang, Z., Liang, X., Xu, H.: 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. pp. 15244–15253 (2023) 3
59. Zhai, X., Mustafa, B., Kolesnikov, A., Beyer, L.: Sigmoid loss for language image pre-training. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 11975–11986 (2023) 4
60. Zhang, R., Guo, Z., Zhang, W., Li, K., Miao, X., Cui, B., Qiao, Y., Gao, P., Li, H.: Pointclip: Point cloud understanding by clip. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8552–8562 (2022) 3
61. Zheng, D., Huang, S., Wang, L.: Video-3d llm: Learning position-aware video representation for 3d scene understanding. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 8995–9006 (2025) 9
62. Zheng, J., Zhang, J., Li, J., Tang, R., Gao, S., Zhou, Z.: Structured3d: A large photo-realistic dataset for structured 3d modeling. In: European Conference on Computer Vision. pp. 519–535. Springer (2020) 4
63. Zhou, J., Wang, J., Ma, B., Liu, Y.S., Huang, T., Wang, X.: Uni3d: Exploring unified 3d representation at scale. arXiv preprint arXiv:2310.06773 (2023) 2, 3, 4, 9, 11, 12
64. Zhou, Z., Wang, P., Liang, Z., Bai, H., Zhang, R.: Cross-modal 3d representation with multi-view images and point clouds. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 3728–3739 (2025) 2
65. Zhu, X., Zhang, R., He, B., Guo, Z., Zeng, Z., Qin, Z., Zhang, S., Gao, P.: Pointclip v2: Prompting clip and gpt for powerful 3d open-world learning. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 2639–2650 (2023) 3, 4
66. Zhu, Z., Ma, X., Chen, Y., Deng, Z., Huang, S., Li, Q.: 3d-vista: Pre-trained transformer for 3d vision and text alignment. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2911–2921 (2023) 3, 9, 11, 12