3D-Aware VLMs with Implicit and Explicit Geometries

Wenhao Li[1], Xueying Jiang[1], Quanhao Qian[2,3], Deli Zhao[2,3], Ran Xu[2,3✉], Shijian Lu[1✉], Gongjie Zhang[4✉]

[1]Nanyang Technological University
[2]DAMO Academy, Alibaba Group [3]HuPan Lab [4]Alibaba Group

Abstract. Despite rapid progress, most existing vision-language models (VLMs) built from 2D visual inputs often struggle when handling various 3D tasks that require fine-grained spatial understanding and reasoning. To bridge this gap, we present VLM-IE3D, a unified framework that enhances the 3D spatial awareness of VLMs by equipping them with both implicit and explicit 3D geometries learned from RGB videos. Our VLM-IE3D introduces Implicit Geometry Tokens (IGTs) that capture high-level geometric priors from input videos, as well as complementary Explicit Geometry Tokens (EGTs) that encode detailed geometric structures from reconstructed 3D attributes. On top of that, VLM-IE3D comes with a 3D-aware adapter that effectively fuses the two types of geometric representations with 2D visual cues. This RGB-only design injects strong 3D inductive biases for fine-grained spatial understanding and reasoning without requiring any additional 3D inputs. Extensive experiments show that VLM-IE3D achieves superior performance consistently across various 3D tasks including 3D video detection, 3D visual grounding, 3D dense captioning, and spatial reasoning. Code and models are available at https://github.com/Vegetebird/VLM-IE3D.

Keywords: Vision-Language Models · Implicit and Explicit Geometries

1 Introduction

Vision-language models (VLMs) have demonstrated great potential in various spatial understanding and reasoning tasks such as embodied navigation [48, 51], vision-language-action (VLA) [23, 29], and 3D scene understanding [18, 46]. Recently, one promising initiative [11, 17, 40, 50] attempts to learn 3D representations directly from 2D visual observations instead of explicit 3D data (*e.g.*, point clouds), mitigating the dependency on specialized 3D sensors and enhancing VLM applicability with more accessible 2D RGB data in real-world scenarios. One prevalent approach endows VLMs with 3D spatial awareness by introducing a 3D geometry encoder [19, 36] to learn implicit 3D representations from video sequences. However, the learned implicit 3D representations often capture coarse and global spatial layouts of scenes, which are insufficient for handling various

✉ Corresponding Authors.

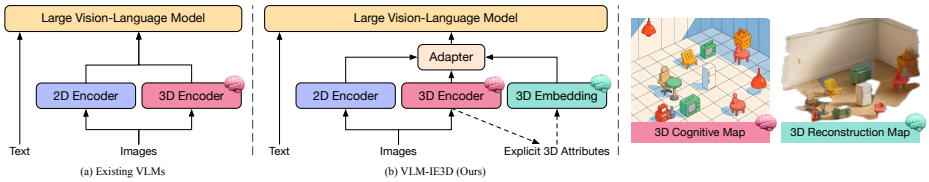

Fig. 1: **(a)** Existing VLMs acquire 3D awareness by learning solely implicit geometric representations that provide a coarse "3D cognitive map". **(b)** The proposed VLM-IE3D introduces explicit geometric representations from reconstructed 3D attributes, offering a "3D reconstruction map" that is equipped with fine-grained local geometries that are supplementary to the implicit representations. These two complementary representations are then fused with 2D visual cues, enhancing the 3D awareness of VLMs with both coarse global layouts and fine-grained local geometric details.

3D understanding tasks that require fine-grained scene structures and geometries for precise spatial localization and reasoning.

This limitation mirrors insights from cognitive science [21, 22], where humans form abstract "3D cognitive maps" for coarse scene awareness, often without geometric details. Similarly, the implicit representations from 3D geometry encoders in current VLMs provide high-level spatial priors. However, their compressed and latent nature makes it difficult for a language model to interpret quantitative geometric properties, even if such information is latently encoded. This leaves fine-grained geometric information inaccessible for precise reasoning tasks in existing VLMs.

In contrast, robotic systems typically rely on explicit 3D representations (*e.g.*, depth maps and point clouds) that form structured "3D reconstruction maps", making quantitative geometric data directly accessible for fine-grained spatial tasks such as navigation and localization. This structured and interpretable form of geometric representations is crucial for precise spatial understanding, yet remains largely unexploited in RGB-only VLM frameworks. Our work is motivated by bridging this gap: equipping VLMs with both the abstract spatial awareness of "3D cognitive maps" (global priors) and the fine-grained structured understanding of "3D reconstruction maps" (fine-grained structures) (see Fig. 1).

To address this gap, we propose VLM-IE3D, a unified framework that equips VLMs with both implicit and explicit 3D geometric representations extracted from RGB videos. Unlike prior works [11, 17, 40, 50] that solely rely on implicit 3D representations, VLM-IE3D introduces two complementary geometry-aware representations to intrinsically inject strong 3D inductive biases into VLMs: **(i)** Implicit Geometry Tokens (IGTs), generated by a 3D geometry encoder, capturing coarse and high-level spatial information for global scene understanding; **(ii)** Explicit Geometry Tokens (EGTs), encoded by a lightweight embedding module from reconstructed explicit 3D representations (*e.g.*, depth maps and point clouds), preserving detailed structural information for fine-grained spatial understanding. These two types of tokens are naturally complementary: IGTs provide high-level and coarse global geometric priors while EGTs capture specific and fine-grained local geometric structures. Furthermore, we develop a 3D-aware

adapter to effectively integrate these implicit and explicit 3D representations with 2D visual cues, producing unified 3D-aware visual embeddings that enable VLMs to perform comprehensive spatial understanding and reasoning at both holistic and fine-grained levels without requiring additional 3D inputs.

We conduct extensive experiments across diverse 3D understanding and reasoning tasks: 3D video object detection, 3D visual grounding, 3D dense captioning, and spatial reasoning. Results demonstrate that VLM-IE3D achieves state-of-the-art performance among RGB-only methods, validating the effectiveness of combining implicit and explicit geometric representations.

Our main contributions are summarized in three aspects. *First*, we present VLM-IE3D, a unified framework that leverages both implicit and explicit geometric representations to intrinsically inject strong 3D inductive biases into VLMs from purely RGB video sequences. *Second*, we introduce two complementary geometry-aware representations: IGTs encode high-level 3D priors and EGTs provide detailed 3D structures. We also design a 3D-aware adapter to effectively integrate these representations with 2D visual cues. *Third*, extensive experiments demonstrate that VLM-IE3D achieves competitive performance on multiple 3D scene understanding and spatial reasoning tasks without additional 3D inputs.

2 Related Work

Vision-Language Models. Vision-Language Models (VLMs) [1,27,28,30] have achieved remarkable success in integrating vision and language. Early works such as CLIP [32] employ contrastive learning to align image and text representations, while subsequent models like BLIP [24] enhance CLIP with larger datasets and improved training strategies. Recent developments [25,26,44] have extended VLMs from static images to dynamic video. For example, Qwen2.5-VL [2] strengthens temporal reasoning through dynamic resolution mechanisms and absolute time encoding. Despite being pre-trained on large-scale image–text corpora, these models are not explicitly optimized to capture 3D geometric information, thereby limiting their 3D capabilities.

3D Vision-Language Models with 3D Inputs. Recent progress in VLMs has extended their capabilities from 2D to 3D scene understanding by leveraging explicit 3D data as inputs (*e.g.*, depth maps and point clouds) [9,12,31,35]. For example, 3D-LLM [13] injects 3D scene representations, obtained by aggregating multi-view 2D features, into a large language model for 3D reasoning. PointLLM [41] combines a point encoder with a large language model to fuse geometric, appearance, and linguistic information for point cloud understanding. Video-3D LLM [49] backprojects each pixel in depth maps into 3D coordinates and injects these spatial cues into video features. Despite their progress, these methods rely on additional explicit 3D inputs. However, obtaining high-quality 3D data requires specialized sensors that are expensive and difficult to deploy, while in most real-world scenarios only 2D data (*i.e.*, images or videos) is available.

3D Vision-Language Models with Only 2D Inputs. To overcome the dependency on explicit 3D data, recent works [11,20,40,43,50] have shifted towards

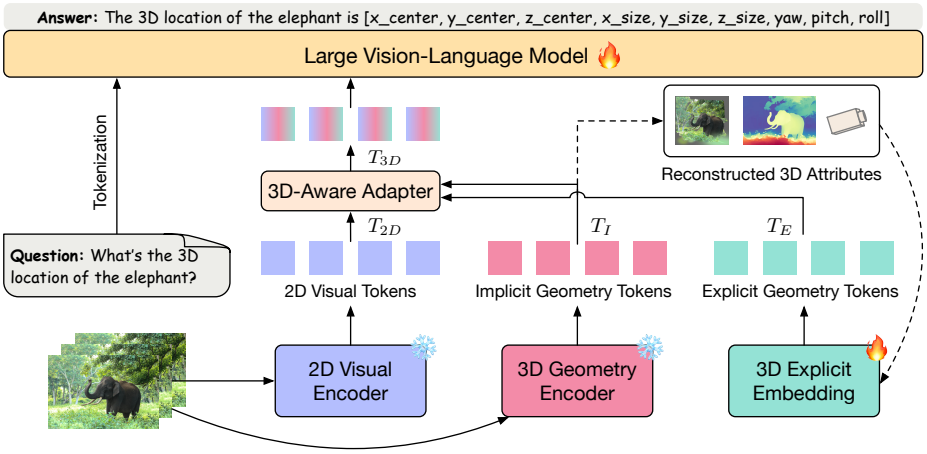

Fig. 2: Overview of the VLM-IE3D framework. VLM-IE3D directly processes RGB frames without explicit 3D inputs, using three streams: (1) a 2D visual encoder that extracts 2D visual tokens; (2) a 3D geometry encoder that generates Implicit Geometry Tokens (IGTs) for high-level 3D information; (3) a 3D explicit embedding that converts the reconstructed 3D explicit representations (*e.g.*, depth maps and point clouds) into Explicit Geometry Tokens (EGTs) encoding fine-grained structural details. Finally, a 3D-aware adapter fuses these complementary tokens into unified and geometry-rich features for VLMs.

understanding 3D scenes solely from RGB videos. These methods typically employ 3D geometry encoders [19, 36], which are pre-trained on large-scale 2D-RGB and 3D paired data, to extract implicit 3D representations to obtain 3D geometric priors and inject them into VLMs to enhance their 3D capabilities. These implicit representations can encode the global 3D structural relationships of scenes, such as the overall arrangement of furniture in a room. In this work, we argue that this paradigm does not fully exploit the potential of 3D geometry encoders to enhance VLMs, as such implicit representations often encode coarse global spatial priors, which struggle with fine-grained 3D understanding tasks. In contrast, our VLM-IE3D introduces a new paradigm that incorporates global 3D spatial priors and detailed 3D structural information into VLMs by jointly leveraging implicit and explicit 3D geometric representations.

3 Method

The overview of our proposed VLM-IE3D is illustrated in Figure 2. Given an RGB video sequence $\{I^i\}_{i=1}^{f}$ and a natural language query Q, we first extract 2D visual tokens $T_{2D} \in \mathbb{R}^{f \times n \times c}$ using a 2D visual encoder. Here, $I^i \in \mathbb{R}^{h \times w \times 3}$, f is the sequence length, $n = \left\lfloor \frac{h}{p} \right\rfloor \times \left\lfloor \frac{w}{p} \right\rfloor$, p is the patch size, and c is the channel dimension. The video sequence is also processed by a 3D geometry encoder [19] to produce Implicit Geometry Tokens (IGTs) $T_I \in \mathbb{R}^{f \times n \times c}$ along with corresponding

3D attributes. The reconstructed 3D attributes are further transformed by a 3D explicit embedding module to generate Explicit Geometry Tokens (EGTs) $T_E \in \mathbb{R}^{f \times n \times c}$. Subsequently, the three types of tokens are fused via a 3D-aware adapter, yielding enhanced 3D-aware visual tokens T_{3D}. Finally, the pre-trained VLM backbone [2] takes T_{3D} and Q as input to generate the final response.

3.1 Implicit and Explicit Geometry Tokens

We observe that the existing spatial VLMs [11, 17, 40, 50] rely solely on implicit 3D representations extracted by a 3D geometry encoder (see Figure 1 (a)), while overlooking the potential of explicit 3D representations. To address this limitation, we introduce Implicit Geometry Tokens (IGTs) and Explicit Geometry Tokens (EGTs). These two types of representations are inherently complementary: IGTs provide high-level and global implicit 3D spatial priors, while EGTs offer specific and fine-grained explicit 3D structural details, enabling more comprehensive 3D understanding and reasoning. In the following, we give details about the proposed IGTs and EGTs.

Implicit Geometry Tokens. 3D geometry encoders [19, 33, 36, 37], pre-trained on a diverse set of 3D geometric tasks (*e.g.*, depth estimation and point reconstruction), have demonstrated strong capabilities in learning and representing 3D geometric information from 2D images. These models encode scenes into token sequences to enable 3D reconstruction and typically comprise three key components: (1) an image encoder that extracts per-frame features, (2) a fusion decoder that employs alternating frame-wise and global self-attention mechanisms to facilitate cross-frame interaction, and (3) task-specific prediction heads for estimating 3D attributes such as depth maps.

Inspired by these advances, we employ AnySplat [19] as our 3D geometry encoder and utilize the implicit 3D representations from the outputs of its fusion decoder as our IGTs $T_I \in \mathbb{R}^{f \times n \times c}$. IGTs encode high-level 3D spatial priors with a global understanding scope of scenes in the form of implicit latent vectors, such as the overall layout of a scene (*e.g.*, the typical structure of a bedroom) and the spatial relationships between objects (*e.g.*, "a TV is mounted on the wall, and a sofa is placed in front of the TV"). By learning such representations, our method achieves strong generalization, enabling it to quickly adapt to 3D structural modeling of different scenes without explicit 3D inputs.

Explicit Geometry Tokens. Despite the strengths of IGTs in learning global scene representations, their compressed and latent form makes it challenging for language models to interpret quantitative geometric properties, thereby limiting their effectiveness in quantitative spatial understanding and reasoning. To address this, we introduce Explicit Geometry Tokens (EGTs) that encode detailed 3D structural information with clear geometric interpretability from reconstructed explicit 3D attributes.

Specifically, we obtain depth maps, camera poses, and 3D Gaussian splats directly from the corresponding prediction heads of the 3D geometry encoder [19]. The point maps are derived by back-projecting reconstructed depth maps into 3D coordinates using reconstructed camera poses. The well-defined 3D attributes

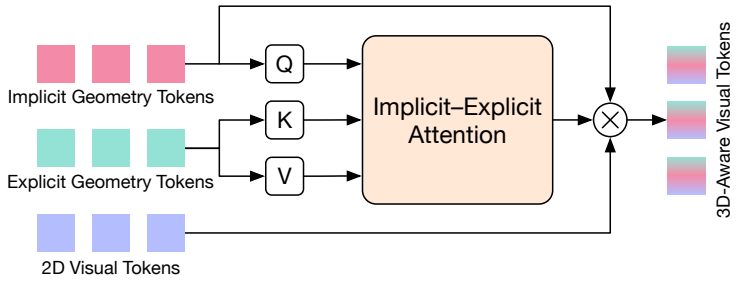

Fig. 3: Illustration of our 3D-aware adapter architecture. It takes the implicit and explicit 3D geometry tokens along with 2D visual tokens as input to generate the final 3D-aware visual tokens. The token compression operation is omitted for clarity.

$X_E \in \mathbb{R}^{f\times h\times w\times c'}$ are then embedded into EGTs $T_E \in \mathbb{R}^{f\times n\times c}$ through a lightweight 3D explicit embedding module, where c' represents the channel dimension, which varies across different 3D attribute types. For efficiency and scalability, we avoid using heavy and complex 3D backbones (such as DepthAnything V2 [42]) and instead use a simple one-layer patch embedding followed by a two-layer MLP with minimal structural changes and training cost. EGTs thus serve as latent representations of explicit 3D structural data with spatial quantitative measurements, providing explicit geometric priors that complement the implicit priors captured by IGTs. These explicit priors focus on fine-grained 3D structural details and can provide VLMs with specific position information for better quantitative understanding and reasoning tasks.

3.2 3D-Aware Adapter

After the above process, we obtain three types of tokens: 2D visual tokens T_{2D}, IGTs T_I, and EGTs T_E. We then design a 3D-aware adapter to effectively integrate these three types of tokens, as shown in Figure 3. Specifically, following the spatial merging strategy of Qwen2.5-VL [2], we concatenate spatially adjacent 2×2 features within each type of tokens and feed them into a two-layer MLP, producing compressed tokens $\{\tilde{T}_{2D}, \tilde{T}_I, \tilde{T}_E\} \in \mathbb{R}^{f\times m\times c}$, where $m = \left\lfloor \frac{h}{2p} \right\rfloor \times \left\lfloor \frac{w}{2p} \right\rfloor$.

To bridge the gap between implicit and explicit 3D geometric representations, we develop an implicit–explicit attention (IEA) module to align and fuse these complementary representations. It performs interaction between IGTs and EGTs through a multi-head cross-attention (MCA) layer. Formally, the dot-product attention [34] in MCA is expressed as:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(QK^T/\sqrt{d}\right)V, \tag{1}$$

where queries $Q \in \mathbb{R}^{n_q\times d}$, keys $K \in \mathbb{R}^{n_k\times d}$, and values $V \in \mathbb{R}^{n_v\times d}$, with d denoting the feature dimension.

For each frame i, MCA takes the compressed IGTs $\tilde{T}_I^i$ as queries and the compressed EGTs $\tilde{T}_E^i$ as keys and values, followed by a residual connection:

$$\tilde{T}_{3D}^i = \tilde{T}_I^i + \text{MCA}(\tilde{T}_I^i, \tilde{T}_E^i, \tilde{T}_E^i), \tag{2}$$

where $\text{MCA}(\cdot)$ denotes the multi-head cross-attention function, and its inputs are queries, keys, and values.

Next, we fuse 3D tokens $\tilde{T}_{3D}$ with compressed 2D visual tokens $\tilde{T}_{2D}$ via element-wise addition:

$$T_{3D} = \tilde{T}_{2D} + \tilde{T}_{3D}. \tag{3}$$

Through the 3D-aware adapter, our VLM-IE3D generates a unified 3D-aware visual representation T_{3D}, which is input into the pre-trained VLM together with text embeddings for final prediction. This design enables the model to jointly leverage both implicit and explicit 3D spatial relationships while preserving 2D visual semantics.

4 Experiments

4.1 Implementation Details

We adopt Qwen2.5-VL-3B [2] as our VLM backbone and AnySplat [19] as our 3D geometry encoder. The 2D visual encoder is the same as that in Qwen2.5-VL [2]. We train the model for one epoch using the Adam optimizer with a warmup ratio of 0.03. The learning rate is gradually increased to 1e-5 and subsequently decayed to zero. The batch size is set to 1 per GPU. During training, the 2D visual encoder and 3D geometry encoder are frozen, while the VLM backbone and 3D explicit embedding remain trainable. All experiments are conducted on 8 H100 80G GPUs.

The input image is rescaled and cropped to 392×518. We set the patch size $p = 14$, the channel dimension $c = 2048$, and the maximum sequence length $f = 32$. The token number is $n = 1036$, and the compressed token number is $m = 252$. For explicit 3D representations, we explore three types of 3D attributes: depth maps ($c' = 1$), point maps ($c' = 3$), and 3D Gaussian splats ($c' = 86$). Since our method does not require ground truth 3D scene information, we follow common 3D geometry encoders [19,36] to adopt the first frame as the reference coordinate system for all tasks, except for 3D visual grounding where bounding boxes are represented in each frame's local coordinates. We use the depth map as the explicit 3D attribute in our EGTs, since it can be more readily obtained from RGB-D sensors for large-scale model training.

4.2 Comparison with State-of-the-Art Methods

To verify the effectiveness of VLM-IE3D, we conduct extensive experiments on 3D scene understanding and spatial reasoning tasks to comprehensively evaluate its 3D understanding and reasoning capabilities. For fair comparison, individual models are trained separately for the two types of tasks.

Table 1: Quantitative results on Scan2Cap.

Method	3D Scene Input	C@0.5↑	M@0.5↑
Scan2Cap [10]	✓	39.1	22.0
3DJCG [3]	✓	49.5	24.2
D3Net [5]	✓	62.6	25.7
Vote2Cap-DETR [8]	✓	61.8	26.2
LL3DA [7]	✓	65.2	26.0
Grounded 3D-LLM [9]	✓	70.2	-
LEO [15]	✓	72.4	27.9
Chat-Scene [14]	✓	77.1	-
LLaVA-3D [52]	✓	79.2	**30.2**
Video-3D LLM [49]	✓	**80.0**	28.5
Qwen2.5-VL-3B [2]	✗	58.0	26.9
VG LLM [50]	✗	78.6	28.6
VLM-IE3D (Ours)	✗	**80.4**	**28.8**

Results on 3D Scene Understanding. We train our model using multi-task learning on combined datasets and evaluate it on three fundamental 3D scene understanding tasks:

(i) *3D dense captioning* involves detecting 3D object proposals and generating descriptions based on object coordinates. We evaluate on the Scan2Cap [10] benchmark. Following previous works [49, 50, 52], we utilize Mask3D-detected object proposals extracted by LEO [15], and the model is then tasked with generating captions conditioned on object center coordinates.

(ii) *3D visual grounding* aims to locate the first frame where a target object appears and its 3D bounding box in camera coordinates. We use the ScanRefer [4] dataset, which comprises 36,665 object descriptions paired with axis-aligned bounding boxes across 562 indoor scans. Following [45, 50], we refine predictions by matching the predicted bounding box with pre-detected object proposals.

(iii) *3D video detection* predicts 3D bounding boxes for all visible objects across a sequence of consecutive frames. We use the dataset curated by [50] from EmbodiedScan [38], which contains sequences of consecutive frames with corresponding object annotations in indoor scenes. Each sample consists of four consecutive frames captured at a rate of 1 FPS. Following [50], we partition the data into 958 training and 243 evaluation scenes, randomly selecting 150 and 10 samples per scene for training and evaluation, respectively. We evaluate performance on 20 common object categories from daily life and report average precision, recall, and F1 score at an IoU threshold of 0.25, denoted as P_{25}, R_{25}, and $F1_{25}$, respectively.

Our VLM-IE3D demonstrates strong performance across all three 3D scene understanding tasks, consistently outperforming methods using only 2D visual input while achieving competitive results with 3D-scene-input approaches.

For 3D Dense Captioning (Scan2Cap), as shown in Table 1, VLM-IE3D achieves the best C@0.5 among all compared methods (80.4) and the best M@0.5 among 2D-visual-input methods (28.8). Compared to baselines, VLM-IE3D shows substantial improvements of 22.4 in C@0.5 over Qwen2.5-VL-3B [2], and 1.8 in

Table 2: Quantitative results on ScanRefer. The content in "()" indicates results with proposal refinement.

Method	3D Scene Input	Acc@0.25	Acc@0.50
ScanRefer [4]	✓	37.3	24.3
MVT [16]	✓	40.8	33.3
ViL3DRel [6]	✓	47.9	37.7
Grounded 3D-LLM [9]	✓	47.9	44.1
Chat-Scene [14]	✓	55.5	50.2
LLaVA-3D [52]	✓	54.1	42.4
Video-3D LLM [49]	✓	**58.1**	**51.7**
SPAR [45]	✗	31.9 (48.8)	12.4 (43.1)
Qwen2.5-VL-3B [2]	✗	34.0 (50.7)	10.6 (44.7)
VG LLM [50]	✗	36.4 (53.5)	11.8 (47.5)
VLM-IE3D (Ours)	✗	**43.2** (55.4)	**16.9** (48.9)

Table 3: Quantitative results on 3D video detection. Frames per second (FPS) was computed on a single H100 GPU.

Method	Param (B)	FPS	P_{25}	R_{25}	$F1_{25}$
Qwen2.5-VL-3B [2]	3.09	14	32.1	30.1	30.9
VG LLM [50]	3.13	7	41.7	35.7	38.2
VLM-IE3D (Ours)	3.23	6	**44.2**	**41.9**	**42.8**

C@0.5 over VG LLM [50], demonstrating that our IGTs and EGTs effectively enhance 3D spatial understanding capabilities.

Table 2 presents results on the 3D visual grounding task (ScanRefer). VLM-IE3D achieves 43.2% and 16.9% accuracy at the IoU thresholds of 0.25 and 0.50, respectively, outperforming all methods without 3D input. The improvements of 9.2 in Acc@0.25 over Qwen2.5-VL-3B [2] and 6.8 over VG LLM [50] demonstrate the superior object localization ability of our method. With proposal refinement, our method achieves 55.4% and 48.9% accuracy, narrowing the gap with 3D-scene-input methods. This indicates that while explicit 3D inputs provide advantages for precise localization, our implicit and explicit geometry encoding approach offers a competitive alternative.

Table 3 shows results on the 3D video detection task. VLM-IE3D achieves 44.2% P_{25}, 41.9% R_{25}, and 42.8% $F1_{25}$, representing improvements of 12.1, 11.8, and 11.9 over the baseline Qwen2.5-VL-3B [2], and 2.5, 6.2, and 4.6 over VG LLM [50]. These consistent gains across all metrics demonstrate that our proposed IGTs and EGTs effectively enhance the model's ability to detect and localize objects in 3D video scenes.

Efficiency and Complexity Analysis. Our method remains lightweight and efficient. As shown in Table 3, VG LLM requires 3.13B trainable parameters, whereas VLM-IE3D has 3.23B. Despite incorporating the 3D explicit embedding module, the 3D-aware adapter, and the 3D attribute reconstruction step for EGTs, VLM-IE3D introduces only 0.10B additional parameters (a marginal 3.2% increase). Furthermore, compared to the implicit baseline VG LLM, our inference speed drops only marginally by 1 FPS (from 7 FPS to 6 FPS on a single H100 GPU). This slight decrease in speed is a highly acceptable trade-off

Table 4: Quantitative comparison with state-of-the-art methods on VSI-Bench.

Models	Avg.	Obj. Count	Abs. Dist.	Obj. Size	Room Size	Rel. Dist.	Rel. Dir.	Route Plan	Appr. Order
		Numerical Answer				Multiple-Choice Answer			
Proprietary Models (API)									
GPT-4o	34.0	46.2	5.3	43.8	38.2	37.0	41.3	31.5	28.5
Gemini-1.5-Flash	42.1	49.8	30.8	53.5	54.4	37.7	41.0	31.5	37.8
Gemini-1.5-Pro	45.4	56.2	30.9	**64.1**	43.6	**51.3**	**46.3**	36.0	34.6
Open-source Models									
InternVL2-8B	34.6	23.1	28.7	48.2	39.8	36.7	30.7	29.9	39.6
InternVL2-40B	36.0	34.9	26.9	46.5	31.8	42.1	32.2	34.0	39.6
Qwen2.5-VL-3B	30.6	24.3	24.7	31.7	22.6	38.3	41.6	26.3	21.2
Qwen2.5-VL-72B	37.0	25.1	29.3	54.5	38.8	38.2	37.0	34.0	28.9
LongVA-7B	29.2	38.0	16.6	38.9	22.2	33.1	43.3	25.4	15.7
VILA-1.5-40B	31.2	22.4	24.8	48.7	22.7	40.5	25.7	31.5	32.9
VideoLLaMA3-7B	35.8	41.9	23.5	42.2	27.1	39.4	-	32.0	31.4
LLaVA-OneVision-72B	40.2	43.5	23.9	57.6	37.5	42.5	39.9	32.5	44.6
LLaVA-NeXT-Video-72B	40.9	48.9	22.8	57.4	35.3	42.4	36.7	35.0	**48.6**
Spatial-Enhanced Models									
SAT-LLaVA-Video-7B	-	-	-	-	47.3	41.1	37.1	36.1	40.4
SPAR-8B	41.1	-	-	-	-	-	-	-	-
RynnEC-7B	45.8	58.5	25.4	54.9	42.7	44.2	-	**38.7**	30.5
VG LLM-4B	47.3	66.0	37.8	55.2	**59.2**	44.6	45.6	33.5	36.4
VLM-IE3D-4B (Ours)	**47.6**	**67.5**	**38.5**	55.0	58.7	47.7	45.1	36.6	31.9

considering the consistent performance improvements (*e.g.*, +4.6 in $F1_{25}$ for 3D video detection compared to VG LLM). The lightweight nature of our explicit geometry embedding (which relies on a simple MLP rather than a heavy deep encoder) ensures that the computational overhead remains negligible, facilitating potential deployment in real-world scenarios.

Qualitative Results Analysis. Figure 4 and Figure 5 illustrate qualitative comparisons among Qwen2.5-VL [2], VG LLM [50], and VLM-IE3D on 3D visual grounding and 3D video detection, respectively. As shown in Figure 4, when given a complex natural language query referring to a target in a video, baseline models like Qwen2.5-VL and VG LLM often struggle to accurately identify the exact first frame where the object appears, and they fail to precisely localize it in 3D space, producing bounding boxes that deviate from actual object boundaries. In contrast, VLM-IE3D correctly grounds the target in the temporal sequence and produces much tighter, more precise 3D bounding boxes. Similarly, in the 3D video detection task (Figure 5), VLM-IE3D captures fine-grained local geometries better, significantly alleviating typical failure cases such as scale mismatch or floating predictions observed in the implicit-only baseline (VG LLM). These qualitative results intuitively validate that complementing high-level IGTs with fine-grained structural EGTs effectively improves both the spatio-temporal and 3D geometric awareness of VLMs.

Results on Spatial Reasoning. We follow the training protocol of [50] for a fair comparison, utilizing subsets from SPAR-7M [45] (234K samples, 3% of the full dataset) and the LLaVA-Hound split of LLaVA-Video-178K [47] (63K samples, 25% of the original data). We then evaluate VLM-IE3D on VSI-Bench, a spatial reasoning benchmark that assesses egocentric-allocentric transformation and relational reasoning capabilities.

Table 4 presents a comprehensive comparison of VLM-IE3D against state-of-the-art methods on VSI-Bench. Despite using only 4B parameters, our method achieves the best average performance of 47.6%, outperforming all baselines

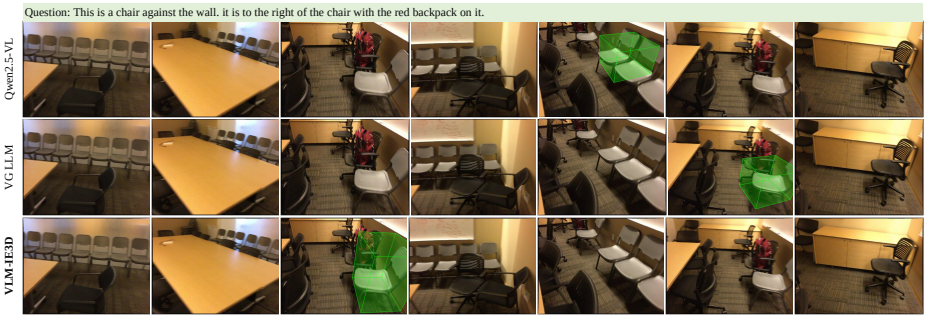

Fig. 4: Qualitative comparison of 3D visual grounding among Qwen2.5-VL, VG LLM, and our VLM-IE3D. This task aims to localize the first frame in the video that contains the object described in the text. Our VLM-IE3D achieves superior localization accuracy and produces more precise object bounding boxes.

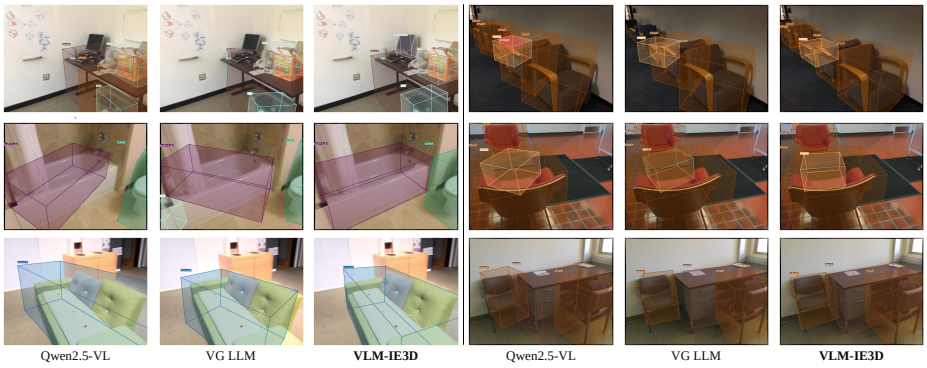

Fig. 5: Qualitative comparison of 3D video detection among Qwen2.5-VL, VG LLM, and our VLM-IE3D. Our method produces more accurate 3D bounding box predictions with superior shape alignment and spatial localization, precisely capturing object geometry.

including proprietary models like Gemini-1.5-Pro (45.4%), and open-source models with significantly larger parameter counts such as LLaVA-NeXT-Video-72B (40.9%). On numerical answer tasks, our model excels particularly in object counting (67.5%), outperforming the previous best spatial-enhanced model VG LLM by 1.5. For multiple-choice tasks, VLM-IE3D achieves competitive performance, particularly in tasks demanding high geometric perception where explicit structures are critical. For instance, on VSI-Bench (Relative Distance), we achieve a notable gain of +3.1 over VG LLM (47.7% vs. 44.6%). These improvements demonstrate that explicit representations effectively capture fine-grained information to resolve complex spatial ambiguities where purely implicit baselines (VG LLM) struggle, validating our motivation that combining implicit and explicit 3D representations provides complementary strengths for comprehensive spatial reasoning.

Table 5: Ablation study on implicit and explicit 3D feature representations.

Method	P_{25}	R_{25}	$F1_{25}$
Baseline [2]	32.1	30.1	30.9
+ EGTs	36.1	33.7	34.7
+ IGTs	41.8	39.7	40.5
+ IGTs + EGTs	44.2	41.9	42.8

Table 6: Ablation study on different fusion strategies for IGTs and EGTs.

Method	P_{25}	R_{25}	$F1_{25}$
VLM-IE3D, Concat	42.7	40.8	41.5
VLM-IE3D, Addition	43.4	41.9	42.4
VLM-IE3D, Weighted	42.3	40.4	41.2
VLM-IE3D, Proposed IEA	44.2	41.9	42.8

4.3 Ablation Study

To verify the impact of each component and design in the proposed model, we conduct extensive ablation experiments on 3D video detection.

Implicit and Explicit Geometry Tokens. We adopt Qwen2.5-VL-3B [2], which does not incorporate any 3D geometric information, as our baseline model and conduct ablation studies to investigate the effects of the proposed IGTs and EGTs. The results are summarized in Table 5. The results demonstrate that incorporating either implicit or explicit 3D representations substantially improves VLM performance. Specifically, equipping the baseline with EGTs improves $F1_{25}$ from 30.9 to 34.7, while incorporating IGTs yields a more significant improvement from 30.9 to 40.5. These gains can be attributed to the high-level 3D priors captured by IGTs and the detailed structural information provided by EGTs, validating our core insight. Notably, IGTs deliver larger improvements than EGTs, which is expected given that IGTs encode richer and more general 3D priors, offering more comprehensive geometric information. When both implicit and explicit 3D representations are integrated, our method achieves further performance gains, reaching an $F1_{25}$ of 42.8. This represents an improvement of 11.9 over the baseline (from 30.9 to 42.8), 8.1 over the baseline with EGTs alone (from 34.7 to 42.8), and 2.3 over the baseline with IGTs alone (from 40.5 to 42.8). These results demonstrate that implicit and explicit 3D representations are complementary, and their joint integration effectively enhances the 3D capabilities of VLMs, validating our motivation.

Fusion Strategy. To investigate the effect of different fusion mechanisms between implicit and explicit representations (*i.e.*, IGTs and EGTs), we conduct an ablation study as shown in Table 6. We ablate the following fusion strategies: **(i)** Concat: concatenation of features followed by a linear projection to align their dimensions. **(ii)** Addition: element-wise addition of the two feature maps. **(iii)** Weighted: element-wise addition with two learnable weights. **(iv)** Proposed IEA: our proposed IEA module, which dynamically exchanges information between IGTs and EGTs. Among these strategies, the proposed IEA yields the best

Table 7: Ablation study on different explicit 3D attributes.

Method	P_{25}	R_{25}	$F1_{25}$
Baseline [2]	32.1	30.1	30.9
+ IGTs	41.8	39.7	40.5
+ IGTs + EGTs (Point)	44.3	41.7	42.6
+ IGTs + EGTs (Depth)	44.2	41.9	42.8
+ IGTs + EGTs (Gaussian)	43.9	41.6	42.5

Table 8: Ablation study on different design choices of the 3D explicit embedding.

Method	P_{25}	R_{25}	$F1_{25}$
VLM-IE3D, None	41.8	39.7	40.5
VLM-IE3D, Pooling	43.6	41.0	42.0
VLM-IE3D, Positional Embedding	44.0	41.5	42.5
VLM-IE3D, Deep Encoder	38.5	34.0	35.9
VLM-IE3D	44.2	41.9	42.8

performance, indicating its effectiveness in integrating implicit and explicit representations. We also explore different fusion strategies for 2D and 3D token fusion as described in Eq. 3. Empirical results show that the direct and non-parametric addition operation is sufficient to achieve strong performance. Therefore, we choose the proposed IEA for fusing IGTs and EGTs and the addition operation for fusing 2D and 3D tokens.

Explicit 3D Attribute. We further investigate the impact of different explicit 3D attributes, including depth maps, point maps, and 3D Gaussians. As shown in Table 7, incorporating any of these explicit 3D representations consistently improves performance across all metrics, achieving highly comparable results ($F1_{25}$ scores ranging from 42.5% to 42.8%). This indicates that as long as the explicit representation provides physically meaningful spatial measurements, it can effectively compensate for the lack of fine-grained structures in IGTs. Consequently, we opt for depth maps in our default setting, as they are the most accessible and parameter-efficient representation among the three. These results validate that different explicit 3D representations can serve as robust geometric complements to IGTs, thereby consistently enhancing the spatial awareness of VLMs.

3D Explicit Embedding. Table 8 presents an ablation study examining different design choices for the 3D explicit embedding module in VLM-IE3D. We evaluate four alternative approaches for embedding explicit geometry: **(i)** None: removing the 3D explicit embedding and EGTs entirely. **(ii)** Pooling: replacing the one-layer patch embedding in our method with the average pooling operation. **(iii)** Positional Encoding: using average pooling with sinusoidal positional embedding [34]. **(iv)** Deep Encoder: treating depth maps as images and utilizing DepthAnything V2 [42] as a deep encoder. The results demonstrate the effectiveness of our simple and lightweight design for the 3D explicit embedding module, which introduces only 0.008B parameters (a negligible 0.25% of the 3.23B trainable parameters) while achieving significant performance gains. Surprisingly, we observe that a deep

Table 9: Ablation study on different 3D geometry encoders.

Method	P_{25}	R_{25}	$F1_{25}$
Baseline [2]	32.1	30.1	30.9
VLM-IE3D, π^3	43.6	41.2	42.1
VLM-IE3D, VGGT	43.1	40.9	41.7
VLM-IE3D, AnySplat	44.2	41.9	42.8
VLM-IE3D, AnySplat w. DepthAnything V2	43.8	41.5	42.5

encoder architecture is challenging to train and yields suboptimal performance. We hypothesize that complex deep models tend to over-process the inputs and extract high-level abstract semantic features. This essentially overlaps with the role of IGTs, causing feature redundancy. In contrast, explicit geometry requires a direct mapping of spatial coordinates. Our lightweight design accurately preserves these fine-grained details without over-abstraction, serving as an ideal complement to the abstraction-heavy IGTs.

3D Geometry Encoder. To evaluate the generalization capability of our method across different 3D geometry encoders, we conduct ablation studies using VGGT [36], AnySplat [19], and π^3 [39]. As shown in Table 9, all tested encoders consistently improve performance over the baseline model [2]. We also replace the depth maps from AnySplat with those reconstructed by DepthAnything V2 [42], yielding IGTs and EGTs from different source models. Notably, this variant also exhibits strong performance, demonstrating that IGTs and EGTs can be extracted from separate models. For efficiency and simplicity, we use a single 3D geometry encoder to obtain both IGTs and EGTs in our final framework. These results demonstrate the robust generalization capability of our VLM-IE3D framework and highlight its flexibility for seamless integration with diverse 3D geometry encoders, suggesting strong potential for leveraging future advancements of 3D geometry encoders.

5 Conclusion

This paper presents VLM-IE3D, a unified framework that enhances the 3D spatial awareness of VLMs by integrating both implicit and explicit 3D geometric representations from purely 2D visual inputs. Through the complementary design of IGTs and EGTs and the effective fusion of the 3D-aware adapter, our method injects strong 3D inductive biases into VLMs and achieves both holistic and fine-grained 3D understanding and reasoning without requiring 3D inputs. Experiments on various 3D scene understanding and spatial reasoning tasks validate the effectiveness of the proposed VLM-IE3D, showing superior performance over 2D-visual-input methods and competitive results with 3D-scene-input methods.

Acknowledgements. This study is funded by the Ministry of Education Singapore, under the Tier-2 project scheme with project number MOET2EP20123-0003.

References

1. Alayrac, J.B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al.: Flamingo: a visual language model for few-shot learning. In: NeurIPS. vol. 35, pp. 23716–23736 (2022)
2. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., Lin, J.: Qwen2.5-VL technical report. arXiv preprint arXiv:2502.13923 (2025)
3. Cai, D., Zhao, L., Zhang, J., Sheng, L., Xu, D.: 3DJCG: A unified framework for joint dense captioning and visual grounding on 3D point clouds. In: CVPR. pp. 16464–16473 (2022)
4. Chen, D.Z., Chang, A.X., Nießner, M.: ScanRefer: 3D object localization in RGB-D scans using natural language. In: ECCV. pp. 202–221 (2020)
5. Chen, D.Z., Wu, Q., Nießner, M., Chang, A.X.: D^3net: A unified speaker-listener architecture for 3D dense captioning and visual grounding. In: ECCV. pp. 487–505 (2022)
6. Chen, S., Guhur, P.L., Tapaswi, M., Schmid, C., Laptev, I.: Language conditioned spatial relation reasoning for 3D object grounding. In: NeurIPS. vol. 35, pp. 20522–20535 (2022)
7. Chen, S., Chen, X., Zhang, C., Li, M., Yu, G., Fei, H., Zhu, H., Fan, J., Chen, T.: LL3DA: Visual interactive instruction tuning for omni-3D understanding, reasoning, and planning. In: CVPR. pp. 26428–26438 (2024)
8. Chen, S., Zhu, H., Chen, X., Lei, Y., Yu, G., Chen, T.: End-to-end 3D dense captioning with Vote2Cap-DETR. In: CVPR. pp. 11124–11133 (2023)
9. Chen, Y., Yang, S., Huang, H., Wang, T., Xu, R., Lyu, R., Lin, D., Pang, J.: Grounded 3D-LLM with referent tokens. arXiv preprint arXiv:2405.10370 (2024)
10. Chen, Z., Gholami, A., Nießner, M., Chang, A.X.: Scan2Cap: Context-aware dense captioning in RGB-D scans. In: CVPR. pp. 3193–3203 (2021)
11. Fan, Z., Zhang, J., Li, R., Zhang, J., Chen, R., Hu, H., Wang, K., Wang, P., Qu, H., Zhou, S., et al.: VLM-3R: Vision-language models augmented with instruction-aligned 3D reconstruction. In: CVPR. pp. 31054–31065 (2026)
12. Fu, R., Liu, J., Chen, X., Nie, Y., Xiong, W.: Scene-LLM: Extending language model for 3D visual reasoning. In: WACV. pp. 2195–2206 (2025)
13. Hong, Y., Zhen, H., Chen, P., Zheng, S., Du, Y., Chen, Z., Gan, C.: 3D-LLM: Injecting the 3D world into large language models. In: NeurIPS. vol. 36, pp. 20482–20494 (2023)
14. Huang, H., Chen, Y., Wang, Z., Huang, R., Xu, R., Wang, T., Liu, L., Cheng, X., Zhao, Y., Pang, J., et al.: Chat-Scene: Bridging 3D scene and large language models with object identifiers. In: NeurIPS. vol. 37, pp. 113991–114017 (2024)
15. Huang, J., Yong, S., Ma, X., Linghu, X., Li, P., Wang, Y., Li, Q., Zhu, S.C., Jia, B., Huang, S.: An embodied generalist agent in 3D world. In: ICML (2023)
16. Huang, S., Chen, Y., Jia, J., Wang, L.: Multi-view transformer for 3D visual grounding. In: CVPR. pp. 15524–15533 (2022)
17. Huang, X., Wu, J., Xie, Q., Han, K.: MLLMs need 3D-aware representation supervision for scene understanding. arXiv preprint arXiv:2506.01946 (2025)
18. 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: ECCV. pp. 289–310 (2024)

19. Jiang, L., Mao, Y., Xu, L., Lu, T., Ren, K., Jin, Y., Xu, X., Yu, M., Pang, J., Zhao, F., et al.: AnySplat: Feed-forward 3D gaussian splatting from unconstrained views. ACM Transactions on Graphics **44**(6), 1–16 (2025)
20. Jiang, X., Li, W., Qian, Q., Zhao, D., Lu, S., Zhang, G., Xu, R.: Towards camera-robust 3D localization: Equation-anchored tool-use for MLLMs. arXiv preprint arXiv:2605.19528 (2026)
21. Johnson-Laird, P.N.: Mental models in cognitive science. Cognitive science **4**(1), 71–115 (1980)
22. Johnson-Laird, P.N.: Mental models: Towards a cognitive science of language, inference, and consciousness. No. 6, Harvard University Press (1983)
23. Kim, M.J., Pertsch, K., Karamcheti, S., Xiao, T., Balakrishna, A., Nair, S., Rafailov, R., Foster, E., Lam, G., Sanketi, P., et al.: OpenVLA: An open-source vision-language-action model. In: CoRL (2024)
24. Li, J., Li, D., Xiong, C., Hoi, S.: Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In: ICML. pp. 12888–12900 (2022)
25. Li, K., He, Y., Wang, Y., Li, Y., Wang, W., Luo, P., Wang, Y., Wang, L., Qiao, Y.: Videochat: Chat-centric video understanding. arXiv preprint arXiv:2305.06355 (2023)
26. Lin, B., Ye, Y., Zhu, B., Cui, J., Ning, M., Jin, P., Yuan, L.: Video-LLaVA: Learning united visual representation by alignment before projection. In: EMNLP. pp. 5971–5984 (2024)
27. Liu, H., Li, C., Li, Y., Lee, Y.J.: Improved baselines with visual instruction tuning. In: CVPR. pp. 26296–26306 (2024)
28. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. In: NeurIPS. vol. 36, pp. 34892–34916 (2023)
29. Ma, Y., Song, Z., Zhuang, Y., Hao, J., King, I.: A survey on vision–language–action models for embodied AI. IEEE Transactions on Neural Networks and Learning Systems (2026)
30. Peng, Z., Wang, W., Dong, L., Hao, Y., Huang, S., Ma, S., Ye, Q., Wei, F.: Grounding multimodal large language models to the world. In: ICLR. vol. 2024, pp. 51575–51598 (2024)
31. Qi, Z., Zhang, Z., Fang, Y., Wang, J., Zhao, H.: GPT4Scene: Understand 3D scenes from videos with vision-language models. In: ICLR (2026)
32. 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: ICML. pp. 8748–8763 (2021)
33. Tang, Z., Fan, Y., Wang, D., Xu, H., Ranjan, R., Schwing, A., Yan, Z.: MV-DUSt3R+: Single-stage scene reconstruction from sparse views in 2 seconds. In: CVPR. pp. 5283–5293 (2025)
34. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NeurIPS. pp. 5998–6008 (2017)
35. Wang, H., Zhao, Y., Wang, T., Fan, H., Zhang, X., Zhang, Z.: Ross3D: Reconstructive visual instruction tuning with 3D-awareness. In: ICCV. pp. 9275–9286 (2025)
36. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: VGGT: Visual geometry grounded transformer. In: CVPR. pp. 5294–5306 (2025)
37. Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., Revaud, J.: Dust3R: Geometric 3D vision made easy. In: CVPR. pp. 20697–20709 (2024)

38. Wang, T., Mao, X., Zhu, C., Xu, R., Lyu, R., Li, P., Chen, X., Zhang, W., Chen, K., Xue, T., et al.: EmbodiedScan: A holistic multi-modal 3D perception suite towards embodied AI. In: CVPR. pp. 19757–19767 (2024)
39. Wang, Y., Zhou, J., Zhu, H., Chang, W., Zhou, Y., Li, Z., Chen, J., Pang, J., Shen, C., He, T.: π^3: Scalable permutation-equivariant visual geometry learning. arXiv preprint arXiv:2507.13347 (2025)
40. Wu, D., Liu, F., Hung, Y.H., Duan, Y.: Spatial-MLLM: Boosting MLLM capabilities in visual-based spatial intelligence. In: NeurIPS. vol. 38, pp. 13569–13597 (2026)
41. Xu, R., Wang, X., Wang, T., Chen, Y., Pang, J., Lin, D.: PointLLM: Empowering large language models to understand point clouds. In: ECCV. pp. 131–147 (2024)
42. Yang, L., Kang, B., Huang, Z., Zhao, Z., Xu, X., Feng, J., Zhao, H.: Depth anything v2. In: NeurIPS. vol. 37, pp. 21875–21911 (2024)
43. Zhang, G., Li, W., Qian, Q., Wang, J., Zhao, D., Lu, S., Xu, R.: On the generalization capacities of MLLMs for spatial intelligence. In: ICLR (2026)
44. Zhang, H., Li, X., Bing, L.: Video-LLaMA: An instruction-tuned audio-visual language model for video understanding. In: EMNLP. pp. 543–553 (2023)
45. Zhang, J., Chen, Y., Xu, Y., Huang, Z., Mei, J., Chen, C., Zhou, Y., Yuan, Y.J., Cai, X., Huang, G., et al.: From flatland to space: Teaching vision-language models to perceive and reason in 3D. In: NeurIPS. vol. 38 (2026)
46. Zhang, T., He, S., Dai, T., Wang, Z., Chen, B., Xia, S.T.: Vision-language pre-training with object contrastive learning for 3D scene understanding. In: AAAI. vol. 38, pp. 7296–7304 (2024)
47. Zhang, Y., Wu, J., Li, W., Li, B., Ma, Z., Liu, Z., Li, C.: LLaVA-Video: Video instruction tuning with synthetic data. arXiv preprint arXiv:2410.02713 (2024)
48. Zhang, Y., Ma, Z., Li, J., Qiao, Y., Wang, Z., Chai, J., Wu, Q., Bansal, M., Kordjamshidi, P.: Vision-and-language navigation today and tomorrow: A survey in the era of foundation models. arXiv preprint arXiv:2407.07035 (2024)
49. Zheng, D., Huang, S., Wang, L.: Video-3D LLM: Learning position-aware video representation for 3D scene understanding. In: CVPR. pp. 8995–9006 (2025)
50. Zheng, D., Li, Y., Wang, L., et al.: Learning from videos for 3D world: Enhancing mllms with 3D vision geometry priors. In: NeurIPS. vol. 38, pp. 20560–20586 (2026)
51. Zhou, G., Hong, Y., Wu, Q.: NavGPT: Explicit reasoning in vision-and-language navigation with large language models. In: AAAI. vol. 38, pp. 7641–7649 (2024)
52. Zhu, C., Wang, T., Zhang, W., Pang, J., Liu, X.: LLaVA-3D: A simple yet effective pathway to empowering LMMs with 3D capabilities. In: ICCV. pp. 4295–4305 (2025)

Supplementary Material

This supplementary material covers the following details:
– The prompts for 3D scene understanding tasks (Sec. A).
– Additional visualization results (Sec. B).

A Example Prompts

The detailed prompts for the three 3D scene understanding tasks are provided in Table 10.

B Additional Visualization Results

We provide qualitative results in Figures 6 and 7, comparing VLM-IE3D with Qwen2.5-VL [2] and VG LLM [50] on 3D video detection and 3D visual grounding. VLM-IE3D achieves more accurate localization of objects given natural language queries and produces tighter bounding boxes with improved detection precision. These visual results confirm that incorporating implicit and explicit 3D geometric priors via IGTs and EGTs substantially enhances the 3D capabilities of VLMs, leading to more robust object grounding and detection performance.

Table 10. The prompts for 3D scene understanding tasks.

3D Visual Grounding

Question: Frame-0: <image> Frame-1: <image> Frame-2: <image> Frame-3: <image>···
Localize the first clear frame in the video showing the object described in the text.
Text: There is an upper cabinet in the room. It has a fridge to its right.

Answer:
```JSON
{"frame": 7, "bbox_3d": [-0.2, -0.07, 2.3, 1.23, 0.33, 0.84, -0.5, 1.53, -2.1]} ```
```

3D Dense Captioning

Question: <image><image><image><image>···
Carefully watch the video and describe the object located at [3.38, -0.19, 2.07] in detail.

Answer: The object is a cabinet on the right side of the room at the end of the walkway with a red chair against the wall. It is white and has a swivel chair sitting in front of it.

3D Video Object Detection

Question: <image><image><image><image>···
Detect the 3D bounding boxes in the camera coordinate system of the first frame.
Output a JSON list where each entry contains the object name in "label" and its 3D bounding box in "box_3d".
The 3D bounding box format should be [x_center, y_center, z_center, x_size, y_size, z_size, yaw, pitch, roll].

Answer:
```JSON
[
    {"label": "socket", "bbox_3d": [0.85, -0.54, 0.5, 0.01, 0.11, 0.11, 1.34, 0.74, -3.11]},
    {"label": "doorframe", "bbox_3d": [0.3, -0.62, 0.86, 0.1, 0.9, 2.08, 1.35, 0.74, -3.11]},
    ...
] ```
```

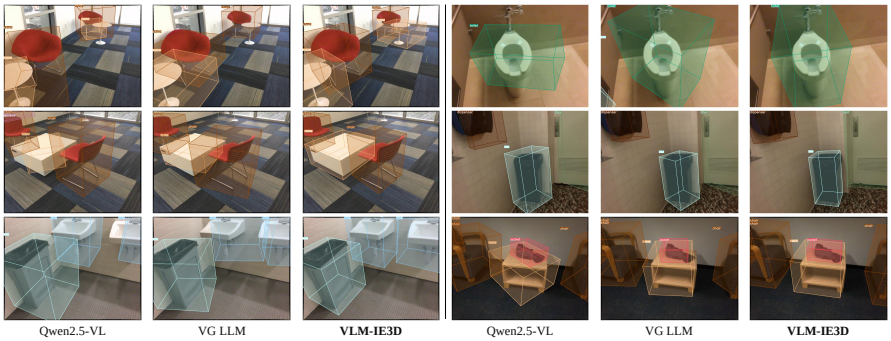

Fig. 6: Qualitative comparison of 3D video detection among Qwen2.5-VL, VG LLM, and our VLM-IE3D.

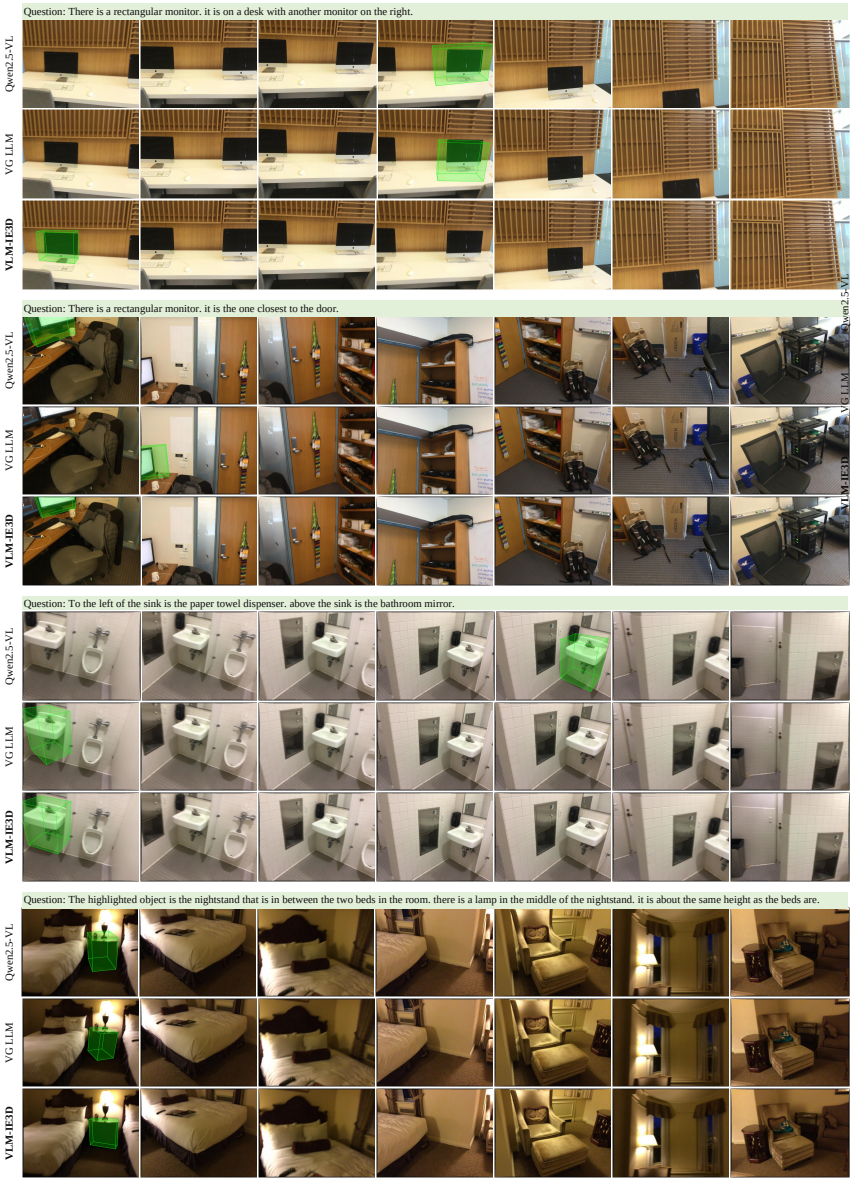

Fig. 7: Qualitative comparison of 3D visual grounding among Qwen2.5-VL, VG LLM, and our VLM-IE3D. This task aims to localize the first frame in the video that contains the object described in the text.