Efficient Encoder-Free Fourier-based 3D Large Multimodal Model

Guofeng Mei[1] Wei Lin[2] Luigi Riz[1] Yujiao Wu[3] Yiming Wang[1] Fabio Poiesi[1]
[1]Fondazione Bruno Kessler, Italy [2]JKU Linz, Austria [3]CSIRO, Australia
{gmei,luriz,ywang,poiesi}@fbk.eu, wlin2021at@gmail.com, yujiao.wu@csiro.au

Abstract

Large Multimodal Models (LMMs) that process 3D data typically rely on heavy, pre-trained visual encoders to extract geometric features. While recent 2D LMMs have begun to eliminate such encoders for efficiency and scalability, extending this paradigm to 3D remains challenging due to the unordered and large-scale nature of point clouds. This leaves a critical unanswered question: How can we design an LMM that tokenizes unordered 3D data effectively and efficiently without a cumbersome encoder? We propose Fase3D, the first efficient encoder-free Fourier-based 3D scene LMM. Fase3D tackles the challenges of scalability and permutation invariance with a novel tokenizer that combines point cloud serialization and the Fast Fourier Transform (FFT) to approximate self-attention. This design enables an effective and computationally minimal architecture, built upon three key innovations: First, we represent large scenes compactly via structured superpoints. Second, our space-filling curve serialization followed by an FFT enables efficient global context modeling and graph-based token merging. Lastly, our Fourier-augmented LoRA adapters inject global frequency-aware interactions into LLM backbones at a negligible cost. Fase3D achieves performance comparable to encoder-based 3D LMMs while being significantly more efficient in computation and parameters. Project website: https://tev-fbk.github.io/Fase3D.

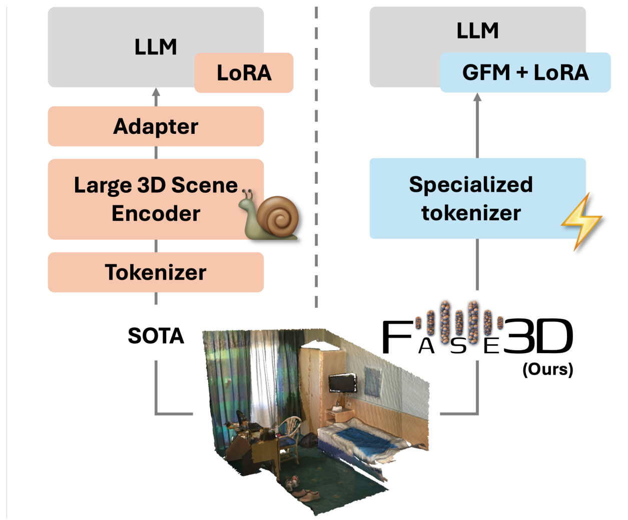

Figure 1. Fase3D's contribution overview. Mainstream 3D LMMs are based on computationally-heavy scene encoders to extract geometric features before alignment with the LLM. In contrast, our method (Fase3D) employs a lightweight Fourier-based tokenizer to process raw point clouds directly and introduces Fourier-augmented LoRA adapters, which infuse global frequency-aware context into the LLM without additional computational overhead.

1. Introduction

A typical practice in 3D Large Multimodal Models (LMMs) involves using pre-trained vision encoders (*e.g.*, CLIP [42] or Sparse 3D U-Net [12]) to extract high-level visual semantics, which are then mapped into the language model's embedding space [26, 35]. These encoders are effective, but impose substantial computational overhead and limit input flexibility. To improve scalability and efficiency, recent works on 2D LMMs have explored vision encoder-free architectures, such as EVE/EVEv2 [13, 14] and Mono-InternVL [32]. Although constructing these models is challenging due to the lack of large-scale vision pre-training, specialized modules, such as visual experts [32] or modality-aware components [14], have enabled encoder-free designs to approach the performance of encoder-based counterparts. In the 3D domain, however, encoder-free scene LMMs remain largely unexplored. Using multi-view 2D workarounds is insufficient, as sacrificing the illumination invariance of 3D sensors degrades performance in low-light environments [55]. Moreover, directly transferring 2D encoder-free architectures to native 3D data is non-trivial. Unlike regular pixel grids, point clouds are unordered and irregular, requiring specialized permutation-invariant operators [27, 34, 39] or serialization techniques [36, 50]. This gap raises a critical question: *how can we design encoder-free 3D LMMs that operate directly on point clouds while remaining both effective and efficient?* Designing such models demands addressing unique 3D challenges. Since vision encoder-free architectures lack traditional pre-training, they must incorporate explicit inductive bias to robustly tokenize the inherent unordered nature of point clouds. This specialized tokenizer must have minimal learnable parameters for efficiency. Furthermore, given the arbitrary length and massive scale of point clouds, the overall LMM architecture must be computationally and memory efficient.

In this paper, we introduce an efficient, encoder-free, Fourier-based 3D scene LMM, named Fase3D, that effectively processes scene-level point clouds. Fase3D introduces a novel interaction by viewing token processing as a synthesis between the spatial and frequency domains. This dual-domain aggregation effectively captures global and local semantic and geometric information while inherently reducing complexity by approximating self-attention. Its core mechanisms are point cloud serialization [50] and the Fast Fourier Transform (FFT), which together yield a highly effective and efficient tokenizer. FFT is a powerful operator that can approximate self-attention and aggregate global context while being computationally efficient [17]. We leverage frequency domain processing across several stages of our pipeline. We first apply FFT to precomputed, serialization-based geometry superpoints to generate context-aware candidate tokens. We then introduce a sparse graph-based token merging formulation that adaptively reduces the token set, significantly lowering GPU cost. Lastly, we propose a novel strategy for training the LLM by augmenting LoRA layers [19] in the frequency domain of low-pass filtered input tokens. We evaluate our method on 3D dense captioning and question answering. With substantially fewer activated visual parameters, Fase3D achieves comparable performance to state-of-the-art approaches, like LL3DA [6] and PerLA [35], on ScanQA [2], SQA3D [33], ScanRefer [5], and Nr3D [1].
To summarize, our main contributions are:

- We present *Fase3D*, the first scene-level encoder-free 3D LMM that eliminates dedicated 3D encoders and instead integrates superpoint tokenization, positional encoding, and FFT-based augmentation into a standard LLM.
- We propose an *FFT context enhancer* that leverages space-filling curves (SFCs) to enable efficient frequency-domain mixing and compact token merging.
- We introduce an efficient *sparse kNN superpoint-graph* construction based on space-filling curve ranking, providing a structured yet lightweight representation.
- We design a *Fourier-augmented LoRA adapter* that enriches the LLM's internal layers with global context modeling at negligible computational and parameter cost, preserving the monolithic philosophy.

2. Related Work

Encoder-based 3D LMMs. Early work on 3D vision and language understanding relies on specialized geometric feature extractors to bridge spatial structure and semantics. For Visual Question Answering (VQA), ScanQA [2] proposed a baseline by pairing point-cloud features with text via dedicated encoders. Subsequent methods focused on cross-modal fusion and unified tasks: 3D-LLM [18] introduced pre-training to strengthen cross-attention among point clouds, images, and text. Chatscene [20] embeds segmented 3D objects into LLM-interpretable tokens. LL3DA [6] and PerLA [35] unified captioning, QA, and grounding by coupling a point encoder with Q-Former adapters for alignment. To tackle large scene understanding, methods like LSceneLLM [54] and MICAS [46] were proposed, using techniques such as adaptive region selection, scene magnification, and multi-grained sampling for improved detail capture and efficient grounding. Other approaches, such as DAC [49], offer a simpler CLIP+MLLM recipe for open-set 3D object retrieval. In parallel, many works extend the popular 2D LLaVA [30] architecture: 3D-LLaVA [12] integrates 3D encoders aligned via instruction tuning for open-vocabulary QA and grounding. Similarly, methods like LISA [23] and SceneLLM [16], LLaVA-3D [55], and SceneVerse [22] lift 2D priors (like mask proposals) into the 3D domain for holistic scene understanding. While these methods establish strong baselines, their dependence on computationally expensive encoders constrains input resolution and scalability. Moreover, their resulting feature embeddings often remain semantically misaligned with the reasoning capabilities of LLMs [47]. Most existing systems still require dedicated 3D encoders or projection modules to process geometric information effectively.
Encoder-free LMMs. Recent advances in monolithic 2D architectures, which integrate perception and reasoning within a single decoder-only Transformer, have inspired a shift toward encoder-free Large Multimodal Models (LMMs). Examples include SOLO [7], Fuyu-8B [4], EVE/EVEv2 [13, 14], and Mono-InternVL [32]. These models eliminate modality-specific vision backbones by mapping visual inputs directly into the LLM's token space through lightweight projections. Extending this paradigm to 3D data is non-trivial: point clouds are large, sparse, and unordered, making naïve serialization computationally expensive and permutation-variant, which disrupts instance coherence and global context modeling. Early explorations demonstrate the feasibility of encoder-light designs for object-level reasoning under instruction tuning. ENEL [47] adopts a lightweight hierarchical tokenization strategy to reduce reliance on heavy vision Transformers in ShapeLLM [40] and PointLLM [51]. However, these approaches still struggle to scale to full scenes and capture long-range, cross-instance dependencies. Our Fase3D directly addresses these limitations by making encoder-free 3D modeling practical for scene-level reasoning, while tackling the challenges of token ordering, scalability, and global context integration.

3. Fase3D

Fase3D is a vision encoder-free LMM that relies on a specialized tokenizer to abstract input point clouds into a set of tokens for a decoder-only LLM [52]. Fase3D is designed

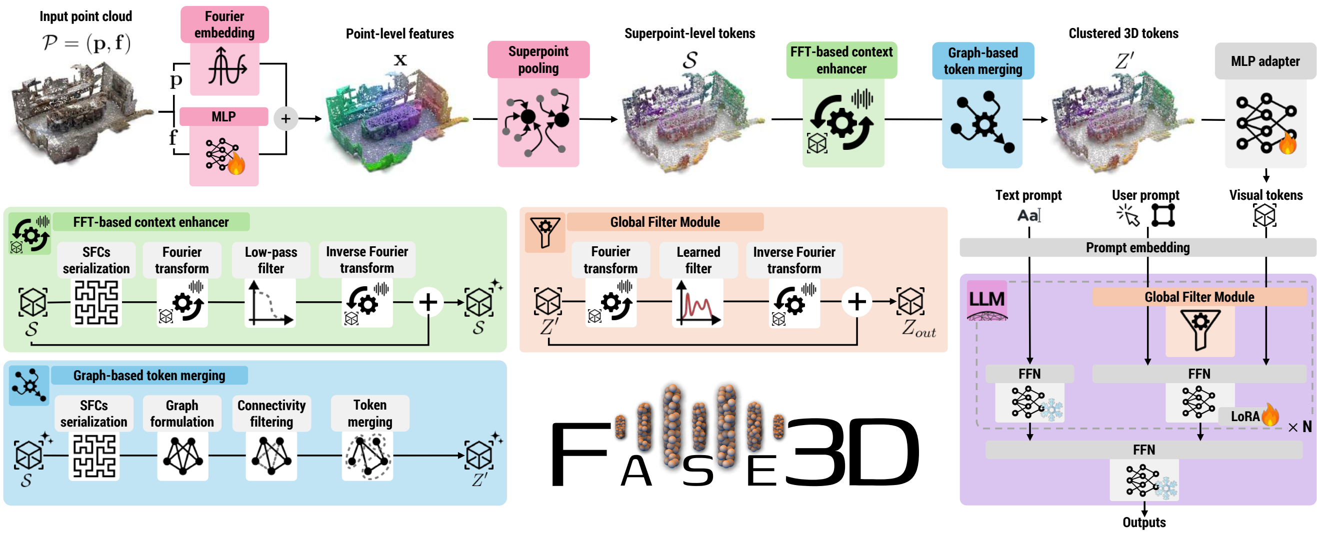

Figure 2. The Fase3D pipeline. A lightweight tokenizer (●) produces M superpoint tokens, which are refined by an FFT-based context enhancer (●). A graph is then constructed, and a token-merging block (●) compresses the tokens into T compact 3D tokens ($T < M$). Finally, an LLM (●) with an FFT-based global filter (●) processes these tokens together with textual and user prompts.

to be parameter- and compute-efficient. It progressively reduces the input token count while enhancing their semantic and spatial information (Fig. 2). We first compute point-level features with a lightweight multi-layer perceptron (MLP) that capture only local context and partition the point cloud into M superpoints via geometric clustering [24] (§3.1). We then average-pool the features within each superpoint to obtain candidate 3D tokens. We serialize the superpoints and employ the Fast Fourier Transform (FFT) on the sequence to further encapsulate contextual information. This FFT-based context enhancer applies frequency weighting to capture context (§3.2). We introduce a graph-based token merging strategy to further aggregate semantic and spatial information into T clusters ($T{<}M$), yielding the final T visual tokens for the 3D scene (§3.3). These tokens, together with the text and user prompts, are fed into a prompt embedding (§3.4) whose outputs serve as inputs to the LLM. Lastly, we enrich global context by training Fourier-augmented LoRA adapters on the weighted tokens in the frequency domain (§3.5).

3.1. Superpoint-based token initialization

To reduce the number of input tokens for the LLM, we employ geometric clustering and produce superpoints [35]. We use average pooling to aggregate information from neighboring points into a corresponding superpoint embedding [12, 43]. Specifically, let $\mathcal{P}{=}\{(\mathbf{p}_i, \mathbf{f}_i)\}_{i=1}^{N}$ be the input point cloud, where $\mathbf{p}_i \in \mathbb{R}^3$ are the 3D coordinates and $\mathbf{f}_i \in \mathbb{R}^{\mathbf{c}_{in}}$ are additional features (*e.g.* color and normal). $\mathcal{P}$ is partitioned into M superpoints $\mathcal{Q}$ centered at $\mathbf{C}{=}\{\mathbf{c}_i\}_{i=1}^{M}$ via geometric clustering [24], where $\mathbf{c}_i \in \mathbb{R}^3$ are the centers. For each superpoint, we compute a token of dimension d. Specifically, we first tokenize $\mathcal{P}$ into embeddings $\mathbf{X}^{(0)} \in \mathbb{R}^{N\times d}$. For each point $\mathbf{p}_i$, we project $\mathbf{f}_i$ into a d-dimensional feature via a shallow learnable multilayer perceptron (MLP), yielding the token $\mathbf{x}_{feat}^{(0)} \in \mathbb{R}^d$. In parallel, we encode $\mathbf{p}_i$ using a non-parametric Fourier feature embedding of varying frequencies, obtaining $\mathbf{x}_{coor}^{(0)} \in \mathbb{R}^{N\times d}$. We obtain the point-level token $\mathbf{x}^{(0)}{=}\mathbf{x}_{feat}^{(0)} + \mathbf{x}_{coor}^{(0)}$. Lastly, we derive the superpoint-level tokens $\mathbf{S} \in \mathbb{R}^{M\times d}$ by average pooling their associated point-level tokens:

$$\mathbf{S} = \mathrm{SptPool}(\mathbf{X}^{(0)}, \mathcal{Q}) \in \mathbb{R}^{M\times d}, \tag{1}$$

where *SptPool* stands for superpoint based average pooling.

3.2. Fourier-based context enhancer

Since the initialized superpoint tokens $\mathbf{S}$ only capture local information, we designed a lightweight token enhancer module that injects global context by operating in the frequency domain [17]. Prior "frequency" methods typically use either (i) voxel/grid-based 3D FFT, which is costly at scene scale, or (ii) Graph Fourier Transform (*e.g.*, PointGST [28]) that requires explicit graph construction and Laplacian operations ($O(M^2)$). In contrast, we first serialize the superpoints [50] into a 1D sequence, enabling the application of FFT-based processing. The FFT operates with complexity $O(M \log M)$ over M tokens and adaptively reweights spectral components to enable context mixing and capture global layout information (*e.g.*, object groupings). The inverse FFT (iFFT) then produces a spatially varying global context field that is fused back into the original tokens via residual addition. This design enriches each token with both local and global context, promoting long-range reasoning. To mitigate ordering bias from a single sequence, we adopt *multi-curve serialization* with varied axis orderings to diversify 1D adjacencies.

Token serialization. We serialize tokens to apply FFT. Specifically, we map the coordinates $\mathbf{C}$ into a *locality-preserving* 1D sequence using *SFCs* [44, 50], centering on four representatives (denoted as $\pi = \{\pi_i\}_i$): the z-order curve, transpose z-order curve, Hilbert curve and the transpose Hilbert curve. We can reorder the superpoint tokens $\mathbf{S}$ into a 1D sequence $\mathbf{S}[\pi_i]$ where 3D locality is preserved.

FFT-based token enhancer. We perform spectral mixing over the serialized token sequence using a real-valued frequency transform to aggregate contextual information [17]. We describe the module using FFT/iFFT notation; in implementation, it can be realized with either rFFT-based operations or a DCT-style transform for improved efficiency on real-valued inputs. Let $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote the 1D FFT and its inverse (iFFT), respectively, applied along the token axis. Additional details are provided in the *Supp. Mat.* For each traversal π_i, we apply a frequency transform to the sorted tokens and modulate informative frequency components by

$$\mathbf{S}'(\pi_i) = \mathcal{F}^{-1}(\mathcal{F}(\mathbf{S}(\pi_i)) \odot \mathbf{G}_v)\,, \tag{2}$$

where $\mathbf{G}_v$ is a learnable non-negative frequency-domain gate, and $\odot$ denotes element-wise multiplication. To obtain position-aware mixing, we apply (2) to overlapping windows of length $L_w{=}128$ and stride $L_s{=}L_w/2$ on $\mathbf{S}(\pi_i)$, and reconstruct by overlap–add with squared-Hann weights [38]. This yields localized spectral aggregation while maintaining a complexity of $O(L_w \log L_w)$ per window. We process all curve traversals, restore the original token order via inverse permutations, and fuse them by uniform averaging:

$$\tilde{\mathbf{S}} = \frac{1}{|\pi|} \sum_{\pi_i} \mathbf{S}'(\pi_i)\,. \tag{3}$$

We fuse the enhancement with a residual as $\mathbf{S} \leftarrow \mathbf{S} + \tilde{\mathbf{S}}$.

3.3. Graph-based token merging

To improve computational efficiency, we reduce the number of tokens by merging superpoints into a compact set of informative tokens that better align with object-level structures. Specifically, we aggregate the initial superpoints $\mathbf{S}$ into T tokens using a lightweight module with only a few learnable parameters, and then feed them into the LLM. Furthermore, we optionally perform spectral clustering on the superpoint graph to generate 3D masks for dense captioning. This graph-based formulation removes the need for an explicit detection stage, as commonly adopted in existing 3D LMMs [12, 35], which rely on learned mask proposals (*e.g.*, Mask3D [45]). In contrast, our method leverages purely geometry-driven superpoints without any learned mask generation. Despite its simplicity, it achieves strong empirical performance.

We model superpoints and their relationships as a graph, where superpoint tokens serve as nodes and their spatial relationships define edges, promoting semantically and spatially coherent representations. We construct a sparse superpoint graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ at the point cloud $\mathcal{P}$ level. This graph is computed once and used as a *topological prior* for the merging stage. The vertices $\mathcal{V}$ correspond to the M superpoints, while the edges $\mathcal{E}$ encode their geometric relationships.

Neighbor searching via window voting. Delaunay triangulation is common for point-cloud graph construction, but can be computationally expensive for large-scale point sets [43]. Instead, we connect superpoints via a point-level window-voting scheme along 1D orderings induced by SFCs. Analogous to our superpoint serialization, we serialize all points into four SFC traversals, which avoids explicit radius or k-NN queries. For each curve, we scan the sorted index list and sample anchor positions p with stride s_r. Let i be the point index at the center of window $\mathcal{W}(p)$, and let j be the point index for any other position $q \in \mathcal{W}(p)$. Let s_i and s_j denote the superpoint indices points i and j are assigned to, respectively. If both points belong to valid superpoints and these superpoints are different ($s_i{\neq}s_j$), we cast a vote for the edge (s_i, s_j) as $v_{s_i,s_j}{\leftarrow}v_{s_i,s_j}{+}1$. Aggregating these votes across all curves yields a sparse set of superpoint pairs with integer vote counts, which defines the graph adjacency. Graph construction complexity is analyzed in the *Supp. Mat.*

Graph-based token merging. We introduce a token merging module to further reduce the number of tokens from M to T ($T < M$). The module compresses the superpoint features $\mathbf{S}$ by jointly exploiting the global context provided by the curve-based serialization order π and the local connectivity encoded in the superpoint graph $\mathcal{G}$, yielding a compact token set that preserves both long-range context and local geometric structure. To this end, we adopt a lightweight point-seeded graph pooling module that aggregates superpoint features over sparse local neighborhoods on the superpoint graph. Concretely, we first sample T anchor points $\{\mathbf{a}_t\}_{t=1}^{T}$ from the input point cloud by farthest point sampling (FPS), and map them to their corresponding superpoints as initial seeds $\{s_t\}_{t=1}^{T}$. To improve coverage, duplicated or invalid seeds are removed by graph-aware non-maximum suppression, and the suppressed slots are replaced with uncovered valid superpoints. For each seed superpoint s_t, we define a sparse local support set as $\mathcal{N}_t = \{s_t\} \cup \mathcal{N}(s_t)$, where $\mathcal{N}(s_t)$ denotes the 1-hop neighbors of s_t on the superpoint graph. For each superpoint $i \in \mathcal{N}_t$, we use the graph connectivity weight as the pooling prior. Specifically, the edge weight encodes the affinity between neighboring superpoints, computed from their feature similarity and further modulated by the multi-curve voting strength. We define $\tilde{w}_{it} = 1$ if $i = s_t$, and $\tilde{w}_{it} = w^{\text{graph}}_{s_t i}$ otherwise, where $w^{\text{graph}}_{s_t i}$ denotes the edge weight between the seed superpoint s_t and its neighbor i on the superpoint graph. The normalized pooling weight is then given by $w_{it} = \frac{\tilde{w}_{it}}{\sum_{j \in \mathcal{N}_t} \tilde{w}_{jt} + \epsilon}$, where $\epsilon = 10^{-5}$ is a small constant to avoid division by zero. Based on these weights, the pooled token feature is computed as

$\mathbf{z}_t^{\text{pool}} = \sum_{i \in \mathcal{N}_t} w_{it} \mathbf{s}_i$, where $\mathbf{s}_i$ denotes the feature of superpoint i. We keep the anchor positions fixed and only pool local superpoint features, *i.e.*, $\mathbf{c}'_t = \mathbf{a}_t$. In this way, the token position is determined by point-level spatial coverage, while the token content is read out from the local superpoint graph neighborhood. After local normalization and projection, we obtain the final merged token representation $\mathbf{z}_t = \phi(\mathbf{z}_t^{\text{pool}})$, where $\phi(\cdot)$ denotes a lightweight feature transformation. Finally, since LLM inputs are inherently sequential and rotary positional embeddings depend on token order, we serialize the merged tokens $\mathbf{Z}' = \{\mathbf{z}_t\}_{t=1}^T$ together with their fixed coordinates $\mathbf{C}' = \{\mathbf{c}'_t\}_{t=1}^T$ along a Hilbert curve, yielding a locality-preserving ordered token sequence for the LLM.

3.4. Prompt embedding

In tasks such as dense object captioning or center-object question answering, the language instruction may include explicit coordinates, instances, or bounding boxes. To handle such inputs, we introduce a *3D coordinate token* that enables the model to incorporate spatial cues into its reasoning process. Concretely, we encode the input coordinates (or box/instance centers) together with their k-nearest neighbors using a Fourier positional encoding layer [6]. The resulting coordinate tokens are concatenated with the 3D patch tokens and text tokens before being fed into the LLM. This design enables coordinate-aware 3D perception and reasoning.

3.5. Fourier-augmented LoRA adapter for LLM

While LoRA provides a parameter-efficient method for adapting the linear feed-forward network (FFN) layers, the representations $\mathbf{Z}' \in \mathcal{R}^{T \times D}$ fed into them are generated by the frozen, pre-trained backbone. These representations are not explicitly optimized for the downstream task, which can limit the potential of the LoRA updates. We propose to enhance these representations by introducing a lightweight *Global Filter Module (GFM)*. The goal is to enrich the token features with globally-mixed information, thus providing a more robust and adaptive input to the LoRA-adapted layers. Our GFM is inspired by the efficiency of Fourier transforms for global mixing [25, 41]. It operates on each token $\mathbf{z} \in \mathcal{R}^D$ within the sequence $\mathbf{Z}'$. Specifically, we first project the feature vector into the frequency domain, apply a learnable filter, and then project back to compute the mixed representation as $\mathbf{z}_{\text{mixed}}$=iFFT(FFT($\mathbf{z}$) $\odot$ $\mathbf{G}_t$), where FFT and iFFT are applied along the channel dimension D. $\mathbf{G}_t \in \mathcal{R}^D$ is a learnable parameter vector that acts as the filter. The final enhanced feature $\mathbf{z}_{\text{out}}$ fed into the LoRA-adapted layer is then computed as an averaged residual connection as $\mathbf{z}_{\text{out}} = (\mathbf{z} + \mathbf{z}_{\text{mixed}})/2$. This blending of the original and filtered representations provides a rich, globally-aware input to the FFN while maintaining a stable learning dynamic. This module remains highly efficient, introducing only D learnable parameters (the filter $\mathbf{G}_t$). We adopt a multi-head formulation with N_h heads to reduce the effective computational cost, resulting in $O\left(\frac{D}{N_h} \log \frac{D}{N_h}\right)$ complexity.

3.6. Training

We adopt a two-stage training strategy inspired by [12, 35]: First, we perform general 3D instruction tuning, then we specialize the model on downstream tasks.

Language modeling loss. For 3D scene–text pairs, we optimize caption generation with next-token cross-entropy, following the standard LLM setup:

$$\mathcal{L}_{\text{LM}} = -\frac{1}{\sum_t m_t} \sum_t m_t \log p_\theta\big(w_t \mid w_{<t}, \mathbf{Z}'\big), \quad (4)$$

where p_θ is the model's token distribution, $m_t = 1[t \geq t_0]$ masks out non-caption prefix/prompt tokens, and t_0 indexes the first caption token. Padding tokens are ignored via m_t.

Datasets. During training, we use the ScanNet v2 [10] portion from the 3DLLM dataset [18], which provides 1,201 training and 312 validation reconstructed indoor scenes. For language supervision, we combine four established sources: ScanQA [2] and SQA3D [33] for 3D question answering, and ScanRefer [5] together with Nr3D (ReferIt3D) [1] for 3D referring expression comprehension/localization. These components jointly define our benchmark, covering both QA and dense captioning. We follow the official train/val splits and report results on the validation set unless otherwise specified. Detailed statistics are in the *Supp. Mat.*

4. Experiments

Implementation details. Following [6, 8], we uniformly sample 50k points per scene. Point features are pooled into superpoint tokens, followed by clustering into 256 tokens. The number of heads is set to N_h=8. We use a frozen Qwen2.5-3B-Instruct [52] language model in *float16*. The LoRA configuration is rank r=768 with scaling α=768 for the first 8 layers. We use AdamW [31] with weight decay 0.1 and cosine decay from 10^{-4} to 10^{-6} over $\sim$100k iterations. We train with a batch size of 8 for seven days on up to four NVIDIA A100 64GB GPUs. For each task, we fine-tune only the parameters for $\sim$30k iterations.

Metrics. We follow the evaluation protocol [6, 35] to evaluate the quality of output responses. We use the abbreviations C, B-4, M and R for CIDEr [48], BLEU-4 [37], METEOR [3], and ROUGE-L [29], respectively. We report #Params, indicating the number of parameters activated for 3D scene tokenization, and FLOP as an efficiency measure.

4.1. Evaluation on 3D question answering (3DQA)

3DQA involves answering free-form questions about a 3D environment, requiring the model to reason about objects, attributes, and relationships within the scene. We benchmark Fase3D on ScanQA [2] and also report competitive

Table 1. Question answering results on ScanQA [2] and SQA3D [33]. #Param/FLOP: number of activated parameters and Floating Point Operation count required for the encoding/tokenization stage. Best result is in **bold**. Second best result is underlined.

Method	LLMs	#Params ↓	FLOP ↓	ScanQA (val)				SQA3D (test)
				R↑	M↑	B-4↑	C↑	EM@1↑
Encoder-based 3D LMMs with point cloud as inputs								
LL3DA [6]	OPT-1.3B [53]	118.87M	40.21	37.31	15.88	13.53	76.79	-
PerLA [35]	OPT-1.3B [53]	119.76M	163.38	39.60	17.44	14.49	78.13	-
3D-LLaVA [12]	Vicuna-1.5-7B [9]	58.26M	37.75	43.10	18.40	17.10	**92.60**	**54.5**
Encoder-free 3D LMMs with point cloud as inputs								
Fase3D	Qwen2.5-3B [52]	**10.54M**	**2.04**	42.56	18.24	**17.12**	90.11	53.9
Fase3D	Vicuna-1.5-7B [9]	12.11M	2.09	**43.37**	**18.61**	16.87	91.74	54.3

performance on SQA3D [33]. Built on ScanNet, ScanQA contains ∼41.4k questions across 800 scenes that probe object recognition and 3D reasoning. SQA3D extends this setting with ∼20.4k situation descriptions covering 6.8k unique situations from 650 scenes, together with ∼33.4k associated questions, placing stronger emphasis on situated and embodied scene understanding. We also report results with Vicuna-1.5-7B [9] for a fair comparison with 3D-LLaVA. Tab. 1 summarizes the results on ScanQA (val) and SQA3D (test). Fase3D achieves performance comparable to 3D-LLaVA on both datasets, while significantly outperforming the other encoder-based baselines. Notably, these results are obtained with substantially fewer vision parameters (#Param: 10.54M–12.11M vs. 58.26M for 3D-LLaVA and ∼119M for LL3DA/PerLA) and a much lower FLOP count (∼2.0 vs. 37.75 for 3D-LLaVA, 40.21 for LL3DA, and 163.38 for PerLA). Fig. 3 compares LL3DA [6], PerLA [35], and Fase3D. In the bathroom scene (left), all methods correctly answer "*What is above the bathroom counter?*" (mirror). For "*What type of dispenser is above the counter?*", Fase3D focuses on the smaller *soap* dispenser, while baselines answer *towel*; all are plausible, though ours is more semantically specific. Yet, all fail on the fine-grained color of the toilet paper rolls, likely due to low texture fidelity and illumination. In the dining scene (right), Fase3D identifies attributes and relationships: it answers "*2 brown chairs*" to "*What chairs are closest to the plant?*", correctly identifies the *round table*, and recovers the chair-fabric colors "*red and black*", whereas the baselines miss at least one of these.

4.2. Evaluation on 3D dense captioning

3D dense captioning is object-centric and conditioned on regions. The model localizes object instances and generates fine-grained, attribute-rich descriptions grounded at 3D coordinates. To evaluate Fase3D, we use two proposal variants: with external segmenter Mask3D [45] and without an external segmenter, using only our graph-based token-clustering proposals. For the variant with our graph-based token merging, we apply spectral clustering on the constructed superpoint graph $\mathcal{G}$ to obtain 48 clusters, which are then treated as proposal instances for evaluation. We condition the generator with tokens derived from each proposal's 3D center, and evaluate on ScanRefer [5] and Nr3D [1]. Following prior work [6, 18], we report m@kIoU, where $m \in \{\text{C, B-4, M, R}\}$ and k is the IoU threshold. For fair comparison, we list models trained with standard per-word cross-entropy and without extra 3D scene pretraining. Tab. 2 shows that Fase3D achieves results comparable to 3D-LLaVA in the same setup with the external segmenter (*Mask3D*) on ScanRefer. In the *spectral clustering* setting, Fase3D slightly underperforms 3D-LLaVa, but still maintains similar performance to PerLA. On Nr3D, Fase3D maintains comparable performance to PerLA with the *Mask3D* and *spectral clustering* variants.

Fig. 4 compares LL3DA, PerLA and Fase3D on ScanRefer [5]. In the bedroom scene on the left, Fase3D delivers the most accurate caption for the object in the magenta bounding box, correctly identifying it as a *pillow* positioned on the "*left side of the bed*". LL3DA produces a confused description mixing *left* and *right*, and PerLA fails to generate any output. In the same scan, all methods correctly recognize the object within the cyan bounding box as a *radiator*, but only Fase3D accurately describes its color. PerLA emphasizes its shape, while LL3DA omits descriptive details. In the second scene (right), all approaches correctly identify the *lamp* in the yellow bounding box despite its small size but misinterpret its spatial relation to the bed. For the blue bounding box, Fase3D achieves higher semantic accuracy by identifying the object as a *wooden stool*, while LL3DA and PerLA describe it less precisely as a *small table* and a *rectangular coffee table*, respectively.

4.3. Ablation Studies

We assess: (i) *patch embedding* alternatives (raw point tokens vs. superpoint pooling with or without an FFT-based context enhancer), (ii) *LoRA with Fourier residuals* design, and (iii) *different LLM* backbones. Unless otherwise specified, all models are trained from scratch on ScanQA, under identical optimization and data settings. We report validation results and vary a single factor at a time, keeping all other components fixed to the default configuration. See

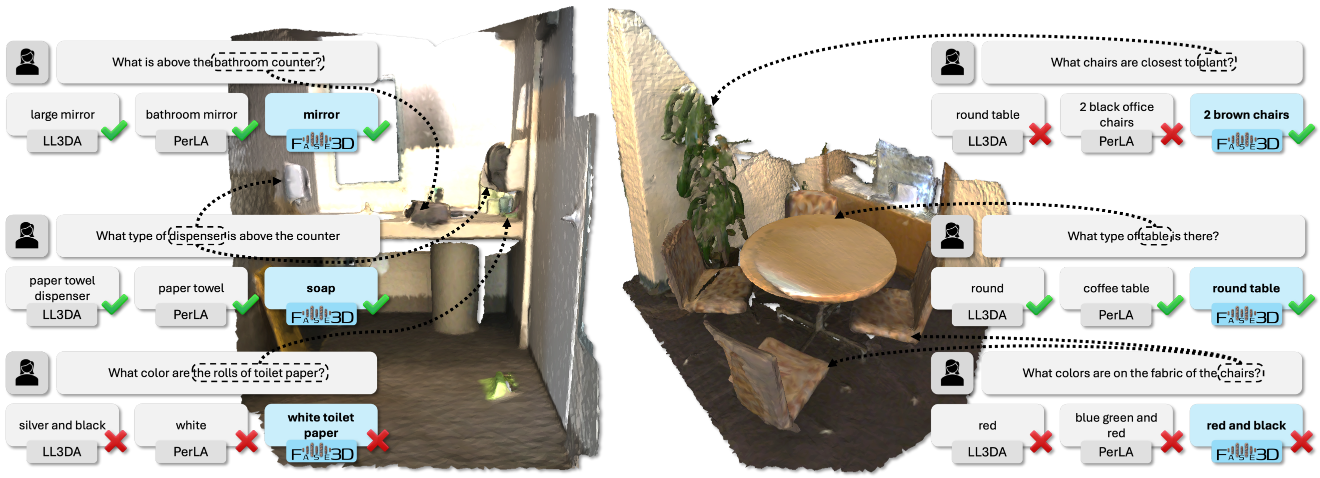

Figure 3. Qualitative results and comparisons between Fase3D, PerLA [35], and LL3DA [6] on the ScanQA [2] dataset.

Table 2. Dense captioning results on ScanRefer and Nr3D. #Param/FLOP: number of activated parameters and Floating Point Operation count required for the encoding/tokenization stage. Best result in **bold**. Second best result is <u>underlined</u>.

Method	Segmenter	#Params ↓	FLOP ↓	ScanRefer				Nr3D			
				C@0.5↑	B-4@0.5↑	M@0.5↑	R@0.5↑	C@0.5↑	B-4@0.5↑	M@0.5↑	R@0.5↑
Encoder-based 3D LMMs											
LL3DA [6]	✓	118.87M	80.43	65.19	36.79	25.97	55.06	51.18	28.75	25.91	56.61
PerLA [35]	✓	119.76M	326.76	69.41	38.02	**29.07**	56.80	**55.06**	**31.24**	**28.52**	**59.13**
3D-LLaVA [12]	✓	<u>58.26M</u>	<u>37.75</u>	**78.80**	<u>36.90</u>	27.10	**57.70**	-	-	-	-
Encoder-free 3D LMMs											
Fase3D	✓	**10.54M**	**2.04**	<u>78.14</u>	**41.34**	<u>27.92</u>	<u>57.63</u>	<u>54.91</u>	<u>30.24</u>	<u>26.48</u>	<u>57.14</u>
Fase3D	✗			70.72	37.83	26.81	56.37	52.89	29.31	26.14	56.41

supplementary material for additional ablation studies.

Patch embedding choices. We examine how 3D inputs are embedded into tokens for the language head (Qwen2.5-3B). Our pipeline first projects points through a lightweight MLP to point features, then aggregates them via superpoint pooling into superpoint tokens. Tab. 3 ablates three factors: using downsampled raw point tokens (*Point*), adding superpoint pooling (*Superpoint*), and adding our lightweight FFT-based context enhancer. Superpoint pooling shortens the token sequence by roughly one order of magnitude while improving semantic coherence, yielding +3.66 CIDEr over point-only tokens (76.04 → 79.70). The FFT-based enhancer alone provides +6.93 CIDEr (76.04 → 82.97). Combining both delivers the strongest ablation result (+10.87 CIDEr; 86.91 total). Training with only raw point tokens is also slower and less stable due to quadratic self-attention. The bottom row reports our *full model with additional pretraining*, further lifting all metrics on the ScanQA val set.

LoRA with Fourier residual. We compare (i) single-branch LoRA (vision-only or text-only), (ii) shared LoRA on both branches (sLoRA), (iii) decoupled LoRA per branch (dLoRA), and (iv) dLoRA augmented with a Fourier residual branch (+FFT). On ScanQA (val) with Qwen2.5-3B [52],

Table 3. Ablation study of vision embedding modules. Point (downsampled raw point tokens), Superpoint (superpoint pooling), FFT (lightweight FFT-based context enhancer).

Module			ScanQA (Validation)			
Point	Superpoint	FFT	R↑	M↑	B-4↑	C↑
✓			37.03	15.43	13.14	76.04
✓	✓		37.18	16.38	13.96	79.70
✓		✓	39.56	17.03	15.11	82.97
✓	✓	✓	41.64	17.80	16.70	86.91
Full model with pretraining			*42.56*	*18.24*	*18.02*	*90.11*

adding a Fourier residual to the vision branch yields the best parameter-efficient results: dLoRA+FFT (vision) improves over dLoRA by +4.38 CIDEr, +1.61 BLEU-4, +0.57 METEOR, and +1.68 ROUGE-L (Tab. 4). However, applying the Fourier residual to both branches reduces the gains on four metrics differently, suggesting frequency-domain cues are most beneficial on the visual pathway. Full end-to-end fine-tuning achieves the highest absolute numbers, but at substantially greater compute and memory; dLoRA+FFT approaches that performance with a fraction of trainable parameters, verifying the effectiveness of our design.

Different LLMs. We further evaluate Fase3D with different language backbones, including OPT-1.3B [53] and

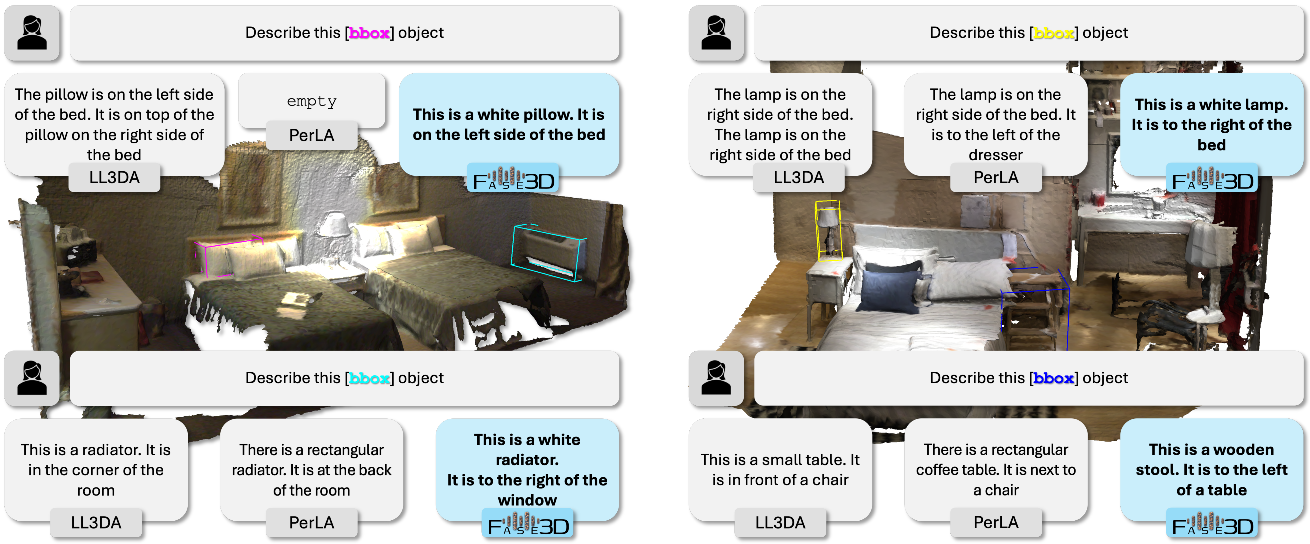

Figure 4. Qualitative comparison between Fase3D, PerLA [35], and LL3DA [6] on the ScanRefer [5] dataset.

Table 4. Ablation study of LoRA placement and Fourier residual on ScanQA (Validation) [2]. sLoRA: the branches share the *same* LoRA modules. dLoRA: the branches use *separate* (decoupled) LoRA modules. "+FFT" adds a parallel Fourier residual branch.

Module		ScanQA (Validation)			
vision	text	R↑	M↑	B-4↑	C↑
LoRA	-	36.98	15.37	13.35	76.45
-	LoRA	36.49	15.40	13.19	76.21
sLoRA	sLoRA	37.34	16.04	12.83	78.24
dLoRA	dLoRA	39.96	17.23	15.09	82.53
dLoRA+FFT	dLoRA	41.64	17.80	16.70	86.91
dLoRA+FFT	dLoRA+FFT	37.54	17.63	15.39	83.64
Full model with pretraining		42.56	18.24	18.02	90.11

Table 5. Question answering results with different LLMs on ScanQA [2]. #Param/FLOP(G) denote the activated parameters and encoding/tokenization FLOPs. Best results in **bold**.

Method	Enc.	#Params ↓	FLOP ↓	ScanQA (val) R↑	M↑	B-4↑	C↑
OPT-1.3B [53] LLM							
LL3DA [6]	✓	118.87M	40.21	37.31	15.88	13.53	76.79
PerLA [35]	✓	119.76M	163.38	39.60	17.44	14.49	78.13
Fase3D	✗	**9.30M**	**2.01**	**40.34**	**17.63**	**15.96**	**86.24**
Qwen2.5-3B [52] LLM							
LL3DA [6]	✓	118.87M	40.21	37.24	16.01	14.91	79.18
PerLA [35]	✓	119.76M	163.38	39.91	16.08	15.53	81.42
Fase3D	✗	**10.54M**	**2.04**	**42.56**	**18.24**	**17.12**	**90.11**

Qwen2.5-3B [52]. Here, *Enc.* denotes variants using a pretrained 3D encoder, while ✗ indicates our encoder-free design that replaces the encoder with a lightweight MLP. As shown in Tab. 5, Fase3D matches or improves performance while drastically reducing 3D front-end cost. With OPT-1.3B on ScanQA, our encoder-free variant uses only 9.30M parameters and 2.01G FLOPs for encoding/tokenization, versus 118.87M/40.21G for LL3DA and 119.76M/163.38G for PerLA. Despite this reduction, it improves CIDEr from 78.13 to 86.24 (+8.11) and achieves the best ROUGE-L, METEOR, and BLEU-4. With Qwen2.5-3B, the trend remains: our model still uses only 10.54M parameters and 2.04G FLOPs, yet attains the best ScanQA validation performance (R 42.56, M 18.24, B-4 17.12, C 90.11). Overall, Fase3D generalizes consistently across LLM backbones, while its encoder-free design matches or even outperforms encoder-based baselines at an order-of-magnitude lower computational cost.

5. Conclusions

We presented Fase3D, an encoder-free, Fourier-based 3D LMM that addresses the twin challenges of scalability and permutation invariance for point clouds. By compactly representing scenes as structured superpoints, serializing them via SFCs, and applying an FFT-based context enhancer, our tokenizer efficiently approximates self-attention while preserving global context. To further reduce the token count, we merge tokens using a sparse graph constructed through curve window-based voting. In the language model head, Fourier-augmented LoRA injects frequency-aware interactions at negligible overhead, enabling strong performance without a heavy geometric backbone. Across experiments and ablations, Fase3D matches or exceeds encoder-based 3D LMMs while substantially reducing computation and parameters. Fase3D inherits the limitations of serialization-based approaches such as PTv3 [50], which may underperform on non-Euclidean long-range relationships in highly cluttered scenes. Future work includes pretraining on larger and more diverse 3D corpora, adaptive or learned serialization, and integration with additional modalities such as RGB images.

Acknowledgments

This work was supported by the PNRR FAIR – Future Artificial Intelligence Research project (PE00000013), funded by the European Union – NextGenerationEU, and by the Ministero delle Imprese e del Made in Italy under the IPCEI Cloud initiative (DM 27 giugno 2022 – IPCEI-CL-0000007). We also acknowledge ISCRA for granting access to the LEONARDO supercomputer, owned by the EuroHPC Joint Undertaking and hosted by CINECA (Italy).

References

[1] Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas Guibas. Referit3d: Neural listeners for fine-grained 3d object identification in real-world scenes. In *ECCV*, pages 422–440. Springer, 2020. 2, 5, 6

[2] Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. Scanqa: 3d question answering for spatial scene understanding. In *CVPR*, 2022. 2, 5, 6, 7, 8, 1

[3] Satanjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *ACL Workshop*, pages 65–72, 2005. 5, 1

[4] Rohan Bavishi, Erich Elsen, Curtis Hawthorne, Maxwell Nye, Augustus Odena, Arushi Somani, and Sağnak Taşırlar. Introducing our multimodal models, 2023. 2

[5] Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. Scanrefer: 3d object localization in rgb-d scans using natural language. In *ECCV*, 2020. 2, 5, 6, 8

[6] Sijin Chen, Xin Chen, Chi Zhang, Mingsheng Li, Gang Yu, Hao Fei, Hongyuan Zhu, Jiayuan Fan, and Tao Chen. Ll3da: Visual interactive instruction tuning for omni-3d understanding reasoning and planning. In *CVPR*, 2024. 2, 5, 6, 7, 8

[7] Yangyi Chen, Xingyao Wang, Hao Peng, and Heng Ji. Solo: A single transformer for scalable vision-language modeling. *TMLR*, 2024. 2

[8] Zhenyu Chen, Ali Gholami, Matthias Nießner, and Angel X Chang. Scan2cap: Context-aware dense captioning in rgb-d scans. In *CVPR*, pages 3193–3203, 2021. 5

[9] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90% chatgpt quality, 2023. 6

[10] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 5

[11] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *ACL*, 2019. 5

[12] Jiajun Deng, Tianyu He, Li Jiang, Tianyu Wang, Feras Dayoub, and Ian Reid. 3d-llava: Towards generalist 3d lmms with omni superpoint transformer. In *CVPR*, pages 3772–3782, 2025. 1, 2, 3, 4, 5, 6, 7

[13] Haiwen Diao, Yufeng Cui, Xiaotong Li, Yueze Wang, Huchuan Lu, and Xinlong Wang. Unveiling encoder-free vision-language models. *NeurIPS*, 37:52545–52567, 2024. 1, 2

[14] Haiwen Diao, Xiaotong Li, Yufeng Cui, Yueze Wang, Haoge Deng, Ting Pan, Wenxuan Wang, Huchuan Lu, and Xinlong Wang. Evev2: Improved baselines for encoder-free vision-language models. *ICCV*, 2025. 1, 2

[15] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 59(2):167–181, 2004. 5

[16] Rao Fu, Jingyu Liu, Xilun Chen, Yixin Nie, and Wenhan Xiong. Scene-llm: Extending language model for 3d visual reasoning. In *WACV*, pages 2195–2206. IEEE, 2025. 2

[17] Ziwei He, Meng Yang, Minwei Feng, Jingcheng Yin, Xinbing Wang, Jingwen Leng, and Zhouhan Lin. Fourier transformer: Fast long range modeling by removing sequence redundancy with fft operator. In *ACL*, 2023. 2, 3, 4

[18] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *NeurIPS*, 2023. 2, 5, 6

[19] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022. 2

[20] Haifeng Huang, Yilun Chen, Zehan Wang, Rongjie Huang, Runsen Xu, Tai Wang, Luping Liu, Xize Cheng, Yang Zhao, Jiangmiao Pang, et al. Chat-scene: Bridging 3d scene and large language models with object identifiers. *NeurIPS*, 37: 113991–114017, 2024. 2

[21] Le Hui, Linghua Tang, Yaqi Shen, Jin Xie, and Jian Yang. Learning superpoint graph cut for 3d instance segmentation. *NeurIPS*, 2022. 5

[22] Baoxiong Jia, Yixin Chen, Huangyue Yu, Yan Wang, Xuesong Niu, Tengyu Liu, Qing Li, and Siyuan Huang. Sceneverse: Scaling 3d vision-language learning for grounded scene understanding. *ECCV*, 2024. 2

[23] Xin Lai, Tian-Zhu Xiang, Shijie Wang, et al. LISA: Reasoning segmentation via large language model. In *CVPR*, pages 14389–14399, 2024. 2

[24] Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *CVPR*, pages 4558–4567, 2018. 3

[25] James Lee-Thorp, Joshua Ainslie, Ilya Eckstein, and Santiago Ontanon. Fnet: Mixing tokens with fourier transforms. In *ACL*, 2022. 5

[26] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024. 1

[27] Jinlong Li, Cristiano Saltori, Fabio Poiesi, and Nicu Sebe. Cross-modal and uncertainty-aware agglomeration for open-vocabulary 3d scene understanding. In *CVPR*, pages 19390–19400, 2025. 1

[28] Dingkang Liang, Tianrui Feng, Xin Zhou, Yumeng Zhang, Zhikang Zou, and Xiang Bai. Parameter-efficient fine-tuning

in spectral domain for point cloud learning. *IEEE TPAMI*, 2025. 3

[29] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004. 5, 1

[30] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *NeurIPS*, 2024. 2

[31] Ilya Loshchilov, Frank Hutter, et al. Decoupled weight decay regularization. *ICLR*, 5, 2017. 5

[32] Gen Luo, Xue Yang, Wenhan Dou, Zhaokai Wang, Jiawen Liu, Jifeng Dai, Yu Qiao, and Xizhou Zhu. Mono-internvl: Pushing the boundaries of monolithic multimodal large language models with endogenous visual pre-training. In *CVPR*, pages 24960–24971, 2025. 1, 2

[33] Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. Sqa3d: Situated question answering in 3d scenes. In *ICLR*, 2023. 2, 5, 6

[34] Guofeng Mei, Cristiano Saltori, Elisa Ricci, Nicu Sebe, Qiang Wu, Jian Zhang, and Fabio Poiesi. Unsupervised point cloud representation learning by clustering and neural rendering. *IJCV*, 132(8):3251–3269, 2024. 1

[35] Guofeng Mei, Wei Lin, Luigi Riz, Yujiao Wu, Fabio Poiesi, and Yiming Wang. Perla: Perceptive 3d language assistant. In *CVPR*, pages 14369–14379, 2025. 1, 2, 3, 4, 5, 6, 7, 8

[36] Guofeng Mei, Luigi Riz, Yiming Wang, and Fabio Poiesi. Vocabulary-free 3d instance segmentation with vision and language assistant. In *3DV*, 2025. 1

[37] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002. 5, 1

[38] Henry Pfister. Discrete-time signal processing. *Lecture Note, pfister. ee. duke. edu/courses/ece485/dtsp. pdf*, 2017. 4, 2

[39] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *NeurIPS*, 30, 2017. 1

[40] Zekun Qi, Runpei Dong, Shaochen Zhang, Haoran Geng, Chunrui Han, Zheng Ge, Li Yi, and Kaisheng Ma. Shapellm: Universal 3d object understanding for embodied interaction. In *ECCV*, pages 214–238. Springer, 2024. 2

[41] Zequn Qin, Pengyi Zhang, Fei Wu, and Xi Li. Fcanet: Frequency channel attention networks. In *ICCV*, pages 783–792, 2021. 5

[42] A. Radford, J. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1

[43] Damien Robert, Hugo Raguet, and Loic Landrieu. Efficient 3d semantic segmentation with superpoint transformer. In *ICCV*, pages 17195–17204, 2023. 3, 4

[44] Hans Sagan. *Space-filling curves*. Springer Science & Business Media, 2012. 4

[45] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. Mask3d: Mask transformer for 3d semantic instance segmentation. In *ICRA*, 2023. 4, 6, 5

[46] Feifei Shao, Ping Liu, Zhao Wang, Yawei Luo, Hongwei Wang, and Jun Xiao. Micas: Multi-grained in-context adaptive sampling for 3d point cloud processing. In *CVPR*, pages 6616–6626, 2025. 2

[47] Yiwen Tang, Zoey Guo, Zhuhao Wang, Ray Zhang, Qizhi Chen, Junli Liu, Delin Qu, Zhigang Wang, Dong Wang, Xuelong Li, et al. Exploring the potential of encoder-free architectures in 3d lmms. *ICLR*, 2026. 2

[48] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, pages 4566–4575, 2015. 5, 1

[49] Zhichuan Wang, Yang Zhou, Zhe Liu, Rui Yu, Song Bai, Yulong Wang, Xinwei He, and Xiang Bai. Describe, adapt and combine: Empowering clip encoders for open-set 3d object retrieval. In *ICCV*, pages 21026–21036, 2025. 2

[50] Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler faster stronger. In *CVPR*, pages 4840–4851, 2024. 1, 2, 3, 4, 8

[51] Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. In *ECCV*, pages 131–147. Springer, 2024. 2

[52] An Yang et al. Qwen2.5 technical report. *arXiv:2412.15115*, 2024. 2, 5, 6, 7, 8

[53] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022. 6, 7, 8

[54] Hongyan Zhi, Peihao Chen, Junyan Li, Shuailei Ma, Xinyu Sun, Tianhang Xiang, Yinjie Lei, Mingkui Tan, and Chuang Gan. Lscenellm: Enhancing large 3d scene understanding using adaptive visual preferences. In *CVPR*, pages 3761–3771, 2025. 2

[55] Chenming Zhu, Tai Wang, Wenwei Zhang, Jiangmiao Pang, and Xihui Liu. Llava-3d: A simple yet effective pathway to empowering lmms with 3d-awareness. *ICCV*, 2025. 1, 2