GeoPACT: Geometry-Preserving Token Compression for Spatial Reasoning in VLMs

Anonymous authors
Paper under double-blind review

Abstract

Vision-language models excel at semantic recognition and visual question answering, yet remain unreliable at spatial understanding: eg. metric distance, object size, relative direction, and multi-view scene geometry. This limitation becomes critical for embodied agents, where visual context grows continuously while memory and latency budgets remain constrained. Although geometry-aware encoders provide rich spatial representations, their dense patch features are costly to process and poorly aligned with language-model embeddings. Compressing them aggressively is therefore essential—but conventional language-only training can discard precisely the local and metric information needed for spatial reasoning.

In this paper, we introduce a general training scheme that explicitly preserves spatial information during compression. During projector pretraining, lightweight auxiliary heads reconstruct dense segmentation and depth targets from the compressed tokens, encouraging them to retain object-level and geometric cues. The heads are removed before downstream fine-tuning, adding no inference cost. We validate this approach with two complementary encoders: DUNE, which provides fine-grained semantic features, and MapAnything, which captures metric multi-view geometry. We then introduce GeoPACT, a lightweight dual-encoder VLM that compresses these representations into 20 tokens per view and 200 scene-level tokens—only 840 visual tokens for 32 views—and couples them with a 0.8B language model. Experiments on 2D and 3D spatial-reasoning benchmarks show consistent improvements, while ablations and frozen-token probes confirm that segmentation and depth information remain more accessible after compression. GeoPACT demonstrates that strong visual compression can preserve geometry, enabling scalable spatial reasoning for embodied VLMs.

1 Introduction

Vision-language models (VLMs) have achieved strong performance on semantic recognition (Chen et al., 2022; Peng et al., 2023) and visual question answering (Alayrac et al., 2022a; Li et al., 2022), yet they remain unreliable at spatial reasoning. Estimating metric distance, object size, relative direction, or the configuration of a multi-view scene requires more than recognizing which objects are present: the model must preserve fine-grained geometric information and make it accessible to the language model. This capability is particularly important for embodied agents which must repeatedly interpret the surroundings and reason about spatial relationships under limited computational and latency budget.

A natural approach to achieve this is to augment a VLM with geometry-aware visual encoders. Recent 3D reconstruction models such as MapAnything (Keetha et al., 2026) produce representations that capture metric 3D structure across multiple views, while universal encoders such as DUNE (Sariyildiz et al., 2025) inherit semantic and geometric knowledge from heterogeneous 2D and 3D teachers. However, these representations are dense and not directly aligned with the embedding space of a language model. Since each frame yields up to thousands of patch tokens, the visual context grows with every additional frame, making dense features impractical for multi-view or streaming inputs, where memory and inference cost become the bottleneck. Aggressive compression is therefore necessary for a lightweight model, but it also creates an information bottleneck: a projector trained only through a language objective may retain visually salient appearance cues while failing to preserve the local and metric structure required for spatial reasoning.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

This raises the central question of our work: *can the compression module itself be trained to preserve spatial structure?* We propose a learning scheme that directly supervises the compact visual representation during projector pretraining. Lightweight prediction heads reconstruct dense segmentation and depth targets generated by perception decoders already available in the visual pipeline. These auxiliary objectives encourage the compressed tokens to retain object-level and geometric information rather than optimizing only for caption generation. Notably, the heads are removed before downstream fine-tuning, making the proposed learning scheme applicable without changing the deployed architecture, token budget, or inference cost.

We choose to assess the effectiveness of auxiliary heads by training VLMs instantiated with following vision encoders: Map Anything and DUNE. These two encoders provide complementary scene understanding, with MapAnything providing global, metrically grounded multi-view scene understanding, and DUNE providing rich semantic features inherited from its' various teachers for each frame in the scene independently. Perciever resampler then compresses DUNE frame dense features into 20 tokens, and Map Anything scene features into 200 tokens, yielding 840 visual tokens for a 32-view input.

The resulting model uses Qwen3.5-0.8B language backbone[1], allowing us to study whether spatial information provided by DUNE and/or MapAnything can survive strong compression in a lightweight setting relevant to embodied applications.

The main contributions of this work are as following

(1) We propose general training scheme that applies *auxiliary dense segmentation and depth supervision directly to compressed visual tokens, encouraging the visual-language interface to preserve spatial structure*. Through thorough ablation analysis, we show that the proposed supervision makes spatial information more accessible after compression and consistently improves 2D and 3D spatial understanding.

(2) We introduce GeoPACT, a *lightweight dual-encoder VLM* that compresses complementary per-view and metric multi-view features into a fixed visual-token budget for enhanced spatial understanding. With only 0.8B parameters, GeoPACT achieves overall *strong results at light latency* cost compared to much larger (7B) spatially specialized VLMs.

2 Related work

Visual token compression in VLMs. Modern vision-language models bridge a frozen or lightly-tuned visual encoder to a language model through an interface module that maps hundreds of patch features into a compact token budget. Perceiver-style resamplers (Jaegle et al., 2021; Alayrac et al., 2022b) achieve this with a fixed set of learnable queries and cross-attention, while alternative designs use a Q-Former (Li et al., 2023) or simple MLP projectors (Liu et al., 2023). More recent work considers compression as an adaptive process rather than a fixed one-shot projection. As such, Hu et al. (2026) proposes a tail-token dropping strategy making the query tokens form a Matryoshka structure (Kusupati et al., 2022). TokenPacker (Li et al., 2024) condenses high-resolution features through coarse-to-fine region-to-point aggregation, Shang et al. (2026) select informative visual tokens and merges redundant ones, and Yang et al. (2026) exploit redundancy in vision-encoder representations to retain a compact set of informative tokens. Chen et al. (2024b) propose a versatile plug-and-play method designed to optimize computational efficiency by learning adaptive attention patterns in early layers and pruning visual tokens in subsequent ones. Compression is essential for multi-image and multi-view inputs, where the raw patch sequence scales as $N_{\text{views}} \times N_{\text{patches}}$ and quickly exceeds the language model's context. The interface is therefore an information bottleneck: whatever the queries do not attend to is unavailable to the language model downstream. Prior works have largely treated this module as a passive conduit, trained only by the downstream task language loss. We instead treat it as an object of supervision in its own right, and ask what the compressed tokens are made to retain.

Spatial and 3D understanding in VLMs. Despite strong 2D semantics, VLMs remain unreliable at metric spatial reasoning: estimating absolute distance, physical object size, and relative direction from images (Yang et al., 2025a; Tong et al., 2024). To overcome these limitations, in the last few

[1]We discard vision encoder for Qwen3.5-0.8B.

years many VLM models have shifted toward spatial intelligence by integrating 3D-aware priors. These models fall into several architectural paradigms such as training-free prompting (Wei et al., 2022; Liao et al., 2024; Mitra et al., 2023; Yang et al., 2023; Liu et al., 2025a; Yang et al., 2025e; Liu et al., 2024; Lei et al., 2024; Li et al., 2025b; Taguchi et al., 2025); customized training designs (Wang et al., 2025b; Liu et al., 2025c; Cho et al., 2024; Li et al., 2025a) or specific training methodologies such as reinforcement learning (Shen et al., 2025; Ouyang et al., 2025) and task-specific policy optimization (Wu et al., 2025; Ma et al., 2025; Wang et al., 2025a) or distillation techniques (Covert et al., 2025; Wang et al., 2023; Bigverdi et al., 2024).

Recent works (Chen et al., 2025; Cha et al., 2024) argue that connectors between visual and textual modalities play a critical role in retaining visual spatial cues for spatial tasks. However, a dominant response to the problem has been data-centric: SpatialVLM (Chen et al., 2024a), InternSpatial (Deng et al., 2025), and SPAR (Zhang et al., 2025) synthesize large-scale spatially-grounded supervision, while SSR (Liu et al., 2025b) scale structured spatial reasoning data and iterative data-model loops. These works show that targeted spatial data improves spatial competence, but remain at the 7B scale and leave the visual interface unchanged. Fan et al. (2026) instead fuses geometry and appearance features, making it architecturally closest to ours, but it does not consider token compression. In contrast, our sub-1B LLM based GeoPACT combining supervised token compression with explicit metric-3D encoding is competitive with specialized systems under a unified evaluation.

Distilling dense perception into VLMs. Closest to out work is a small set of methods that supervise the VLM's visual signal from dedicated perception expert rather than with language alone. OLA-VLM (Jain et al., 2025) distills features from off-the-shelf depth, segmentation and generation encoders into the intermediate layers of the language model during pretraining, using a resampler-based tokens together with a "seek" token that persist in the input sequence at inference. Chain-Of-Visual-Thought (Qin et al., 2025) instead treats compact visual tokens as elements of the language model's autoregressive reasoning chain: within a budget of tokens it distills depth, segmentation from self-supervised experts. Our approach differs along three axes. First, the auxiliary supervision acts on the visual compression module between the encoders and the language model, rather then on language model's hidden states (Jain et al., 2025) or generated reasoning tokens (Qin et al., 2025); the objective therefore shapes the compressed visual interface itself. Second, the head is used only during pre-training and is discarded afterwards, so the deployed model is architecturally identical to its unsupervised counterpart and incurs no additional tokens and no additional inference cost. Third, our approach does not require external annotated data during training: our auxiliary heads rely on the dense depth and segmentation predictions of the pretrained encoders' own perception heads, whereas OLA-VLM and CoVT rely on separate pretrained specialist encoders.

GeoPACT (Figure 1) is a compact vision-language model for multi-view 3D spatial reasoning, built around three ideas: (i) geometry-aware vision encoder which combines a universal 2D perception encoder with an explicit metric-3D scene encoder, (ii) Perceiver-based compression of image and scene tokens into small, fixed token budgets, and (iii) auxiliary pseudo-ground-truth distillation heads that supervise the compressed tokens during pretraining (discarded afterwards). The language backbone is Qwen3.5-0.8B (Qwen Team, 2026), which is natively multimodal but we bypass its vision pathway, replacing with 3D-aware features and use only the language modeling part of it[2].

3 METHOD

Problem setting. Given V views $\{I_1, \ldots, I_V\}$ of a scene and a natural-language prompt, the model generates a textual response. We target spatial questions involving object counting, physical size, metric distance, relative direction, and relationships between objects observed in different views. Answering these questions requires both localized semantic information and a globally consistent geometric representation: appearance features identify objects and their attributes, while metric scene features resolve scale, depth and cross-view relationships. Furthermore, the representation must remain compact enough to avoid overwhelming the LLM context as the number of views increases.

[2]Note that the choice of the backbone is arbitrary, and in principle our method could be applied to any off-the-shelf LLM, not necessarily multimodal.

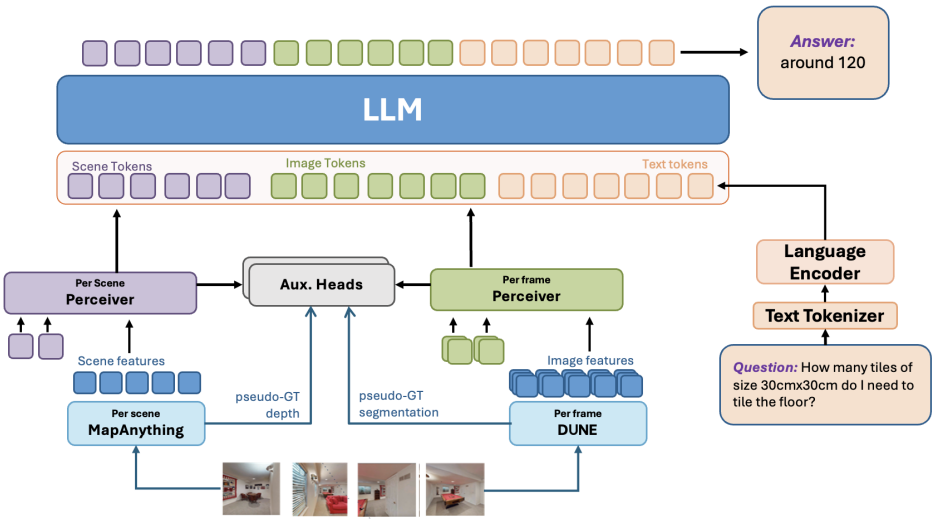

Figure 1: **GeoPACT architecture.** Frozen MapAnything and DUNE encoders feed a Scene Perceiver (200 tokens per scene) and a Frame Perceiver (20 tokens per view) respectively; scene, image, and text tokens are consumed by the LLM. Auxiliary heads are discarded at inference.

3.1 Metric Scene Branch

Spatial questions involving physical size, absolute distance, or relationships between objects observed in different views require a metrically consistent scene representation. We obtain this representation using a frozen MapAnything model (Keetha et al., 2026), a feed-forward multi-view reconstruction network that jointly processes a set of views and predicts metric 3D structure.

MapAnything processes each view at resolution 518×518 with patch size 14. We extract patch features from its final representation together with intermediate features from layers 7, 11, and 14 of its multi-view information-sharing transformer. These layers capture complementary levels of spatial context, and we combine them with a trainable multi-scale mixer. Each feature scale is independently normalized and linearly projected to a common width. Three zero-initialized AdaLN-conditioned residual blocks then progressively incorporate the intermediate features into the final representation. Zero initialization makes the mixer start as an identity mapping and learn how much to retain from each scale.

The resulting multi-view features remain dense and grow with the number of input views. A scene-level Perceiver Resampler compresses them into a fixed representation. It first projects the features to the LLM embedding dimension and then applies six Perceiver blocks, each consisting of cross-attention from learned latent queries to the visual features, latent self-attention, and a $4\times$-expansion GELU feed-forward network. The resampler uses eight attention heads and 200 learned queries, producing exactly 200 scene tokens irrespective of the number of input views.

This branch provides a compact, metrically grounded representation of scene scale, layout, and cross-view object relationships. However, compressing the entire scene into a single token set can remove localized semantic and appearance information required for object-centric questions.

3.2 Fine-grained Image Branch

We complement the global scene representation with per-view features from a frozen DUNE encoder (Sariyildiz et al., 2025). DUNE is a universal ViT distilled from heterogeneous 2D and 3D teachers, including DINOv2 (Oquab et al., 2024), MASt3R (Leroy et al., 2024), and Multi-HMR (Baradel et al., 2024). Its features capture fine-grained semantic and appearance information while inheriting spatial awareness from its 3D teachers (more details in Appendix D).

Each view is processed independently at resolution 672×672 with patch size 14. The resulting patch features are compressed by a second Perceiver Resampler with the same architecture as the scene resampler. This branch uses 20 learned queries, producing 20 tokens per view regardless of image

resolution. The image tokens, scene tokens, and text embeddings are concatenated and passed to the language model (see token-budget ablations in Appendix E).

The two branches provide complementary information: MapAnything captures global metric structure shared across views, while DUNE preserves localized semantic and appearance cues. Nevertheless, both branches rely on aggressive compression. A language objective alone does not explicitly require their Perceiver tokens to retain the spatial information produced by the frozen encoders.

3.3 Auxiliary Supervision of Compressed Tokens

Let X_b denote the dense features produced by branch $b \in \text{scene}, \text{image}$ and let $Z_b = R_b(X_b)$ denote the corresponding compressed tokens. The language model consumes Z_b, but next-token prediction provides only indirect supervision about which visual information the resampler should preserve. We therefore attach lightweight dense-prediction heads directly to Z_b during Stage 1.

Each auxiliary head uses a bank of learned spatial queries that cross-attend to the compressed tokens and reconstruct a dense 48×48 prediction map. Because the heads operate on the same tokens provided to the language model, satisfying the auxiliary objectives requires the resamplers to encode spatial information within their fixed token budgets.

Segmentation and depth supervision. We supervise the Perceiver tokens for both image and scene with the same dense targets. Semantic pseudo-labels are generated by the frozen DUNE segmentation decoder, while metric depth targets are produced by the frozen MapAnything depth decoder. For each view, an auxiliary head reconstructs these targets from the corresponding compressed representation: the 20 per-view image tokens for the image branch and the 200 global scene tokens for the scene branch. Segmentation supervision encourages both representations to retain object identity and boundaries, while the MapAnything depth targets provide metric scale and multi-view-consistent geometric structure. Because both teachers are already part of the visual pipeline, the targets require no additional perception models at training time.

Target formulation. We consider two ways of transferring the dense predictions. Soft-target distillation matches the complete teacher distribution using KL divergence. Hard pseudo-label supervision selects a discrete target at each output location and applies cross-entropy. We use hard pseudo-labels in the final configuration because they provide the strongest downstream results. We hypothesize that their sharper targets place greater emphasis on object boundaries and depth transitions, whereas soft distributions may be satisfied by broader spatial prediction patterns.

$$\mathcal{L}_{\text{Stage1}} = \mathcal{L}_{\text{LM}} + \sum_{b \in \mathcal{B}} \sum_{t \in \mathcal{T}_b} \mathcal{L}_t \left(g_{b,t}(Z_b), T_{b,t} \right). \tag{1}$$

where $\mathcal{L}_{\text{LM}}$ is the frozen LLM's next-token prediction loss, $g_{b,t}$ is the auxiliary head for branch b and task t, and $T_{b,t}$ is the corresponding frozen target. The active tasks are segmentation and metric depth for the two encoder branches. The heads are discarded after Stage 1 and not used at inference.

3.4 Two-stage training

We follow standard two-stage procedure used for VLM training:

Stage 1 — independent branch pretraining against a frozen LLM. The two branches are pretrained independently, each in its own Stage-1 run in which only that branch's tokens reach the LLM: the geometric run trains the multi-scale mixer and scene projector with no image branch present, and the image run trains the image projector. In both runs the entire LLM is frozen; its next-token loss on captions and QA back-propagates through the input embeddings into the trainable modules, shaping the compressed tokens to be "LLM-legible," while the auxiliary heads of Sec. 3.3 regularize them to remain dense-prediction-capable. Isolating the branches at this stage prevents the frozen LLM from satisfying the captioning objective through whichever stream is easier to read, which would leave the other branch's queries under-trained.

In this stage we align the compressed visual representations with the frozen LLM using a mixture of captioning and spatially grounded data. Captioning examples are drawn primarily from FineVision (Wiedmann et al., 2025) and our in-house multi-view dataset, while InternSpatial and

SPAR-7M-RGBD (Zhang et al., 2025) provide explicit spatial supervision and account for 30% of the mixture.

Stage 2 — joint fine-tuning with LoRA. Both branches are then combined, initialized from their independent Stage-1 checkpoints, and fine-tuned jointly: the two attention projectors and the geometric mixer remain trainable, both backbone encoders stay frozen, and the LLM is adapted with LoRA. This stage targets spatial reasoning and instruction following, therefore, we use 60.5% spatial QA, 19.5% VSI-Bench-style instruction data (Yang et al., 2025c), and 20% multi-view captioning. The retained captions preserve the alignment established during Stage 1, while open-ended and multiple-choice questions cover counting, size, distance, orientation, and cross-view relations.

Implementation details on model training and dataset processing are provided in Appendix A and C.

4 Experimental evaluation

Our evaluation aims to assess geometrical understanding acquired by GeoPACT, and identify which components of our model contribute to this understanding. More specifically, we explicitly evaluate: (1) the impact of auxiliary heads, across different types of geometry-related questions, (2) the complementarity of DUNE and MapAnything branches in overall scene spatial understanding, (3) overall performance and efficiency of GeoPACT in the context of various spatially specialized VLMs.

Benchmarks. We evaluate on four spatial-reasoning benchmarks. VSI-Bench (Yang et al., 2025a) evaluates multi-view reasoning over indoor scenes; we use its debiased split of 2,744 questions, covering object counting, absolute distance, object and room size, relative distance and direction, route planning, and appearance order. ReVSI (Zhang et al., 2026) re-annotates the scenes and regenerates the questions to reduce annotation errors and language biases. In both cases we follow the 32-frame evaluation protocol. SPAR-Bench (Zhang et al., 2025) contains 20 tasks spanning depth, distance, object relations, and spatial imagination; we report its overall average. CV-Bench (Yang et al., 2025c) evaluates single-image spatial understanding through 1,438 2D questions on counting and spatial relations and 1,200 3D questions on depth ordering and relative distance. Note that CV-Bench is most distant from our training data, and evaluates rather high-level spatial understanding rather than fine-grained geometry understanding. We use this benchmark as a way to evaluate generalization of our models.

Evaluation protocol. We evaluate all models through VLMEvalKit (Duan et al., 2024) under a common protocol. VSI-Bench and ReVSI use the same 32 sampled frames for every model, and all models receive the same benchmark instructions through their respective chat templates. We use greedy decoding and the official benchmark-specific answer extraction and scoring procedures. Complete generation settings, numerical-answer normalization, and LLM-judge configuration are provided in the Appendix section B.

Baselines. We compare against groups of model, all re-evaluated using the protocol described above. Qwen3.5-0.8B and Qwen3.5-2B (Qwen Team, 2026), the former shares the language backbone and differs from GeoPACT only in the visual interface, the latter allows to assess how far generalist scaling alone carries spatial performance. We also provide results on Qwen3.5-0.8B finetuned (Qwen3.5-0.8B-FT) on our data using solely the stage 2 as there is no need to align the visual representation. This baseline allows to assess the impact of Stage-2 data in the emergence of geometrical understanding. Furthermore we add the following SOTA methods: VLM3R-7B (Fan et al., 2026) derives implicit 3D and camera-pose tokens from monocular video with a geometry encoder and fuses them with appearance features, making it the closest architectural analogue at our geometric branch. RoboBrain2.0-7B (RoboBrainTeam et al., 2025) is an embodied spatial-reasoning model trained for downstream robotic use.

4.1 Effect of Auxiliary Supervision

Table 1 isolates auxiliary supervision by varying only the Stage-1 objectives. We note that auxiliary heads are beneficial in case of single vision encoder (DUNE, MapAnything), or when both vision encoders are combined.

Table 1: **Ablations.** All rows use the Stage-1/Stage-2 recipe of Tab. 3. *Seg*/*Depth*: auxiliary heads active in Stage 1, trained on hard pseudo-labels or soft targets with KL loss (denoted by ‡) obtained from the encoders' own decoders.

Branches	Seg	Depth	VSI-deb.	ReVSI	CVB-2D	CVB-3D
DUNE only			50.7	35.5	52.6	55.3
	✓		51.1	39.2	60.4	62.1
		✓	54.3	40.2	53.1	54.6
	✓‡	✓‡	50.8	38.4	61.3	63.0
	✓	✓	56.1	43.5	63.8	65.3
MapAnything only			53.2	42.6	40.0	60.5
	✓		54.5	43.0	45.2	60.7
		✓	56.5	45.7	43.6	58.3
	✓‡	✓‡	56.2	43.2	44.4	58.1
	✓	✓	58.8	47.2	45.8	61.9
DUNE + MapAnything			57.0	44.6	51.3	55.5
	✓		58.1	45.8	63.1	66.4
		✓	58.8	46.9	58.4	66.9
	✓‡	✓‡	54.3	44.3	69.6	69.3
	✓	✓	**62.1**	**48.4**	**72.8**	**70.2**

Table 2: **Per-task effect of auxiliary supervision.** Results for the full image and scene model, with and without segmentation and/or depth pseudo-GT supervision.

	VSI-Bench (debiased)								CV-Bench 2D		CV-Bench 3D	
Aux. heads	Count	AbsD	Size	Room	RelD	RelDir	Route	AppO.	Count	Rel.	Depth	Dist.
Qwen3.5-0.8B-FT	56.3	45.6	57.7	67.4	62.6	60.4	**44.7**	**80.8**	54.3	**88.2**	**84.8**	**73.3**
GeoPACT	**62.7**	**53.8**	**61.9**	**70.7**	**61.9**	**74.1**	40.4	75.8	**69.1**	74.5	76.2	64.1
w/o aux heads	53.2	42.0	55.5	69.2	51.1	70.2	35.5	71.1	48.9	55.1	54.3	56.6
Seg Head Only	59.1	44.0	53.4	65.2	54.0	71.4	42.7	74.9	67.1	59.1	67.1	65.7
Depth Head Only	59.2	46.4	55.7	69.1	54.5	71.9	42.1	71.4	50.1	61.2	71.3	62.5
Δ (GeoPACT - w/o aux heads)	+9.5	+11.8	+6.4	+1.5	+10.8	+3.9	+4.9	+4.7	+20.2	+19.4	+21.9	+7.5

Individual objectives. The benefits of segmentation and depth heads are complementary. Segmentation produces the larger improvement on CV-Bench 2D, whereas depth boosts more VSI-Bench-Debiased, ReVSI, and CV-Bench 3D. Combining both objectives gives the strongest overall results. The image-only model follows the same general trend, showing that the learning scheme remains effective without the metric scene branch.

Hard versus soft targets. The target formulation is also important. For the full model, soft-target distillation improves the CV-Bench scores but falls below the no-head baseline on VSI-Bench-Debiased and ReVSI. Hard pseudo-labels perform better on all four benchmarks. We hypothesize that their sharper targets place greater emphasis on object boundaries and depth transitions, whereas matching the complete teacher distribution might provide noisier learning signal.

Per-task effects. Table 2 breaks down the effect of joint segmentation and depth supervision by task. Overall, on VSI-Bench-Debiased, the largest gains occur in absolute distance, relative distance, and object counting, which improve by 11.8, 10.8, and 9.5 points. On CV-Bench, depth ordering improves by 21.9 points, while 2D counting and spatial-relation reasoning improve by 20.2 and 19.4 points. The benefits therefore concentrate on questions requiring localized object information, metric estimates, or depth relationships to survive compression.

We see that Segmentation head mostly benefits tasks requiring object localization (eg. counting, appearance order and relative direction), while it's benefit on metric-estimate tasks (eg. Object Size, Room size, Absolute Distance) is less clear. On the other hand, Depth head brings larger improvements on metric tasks, and is therefore complements Segmentation head efficiently.

Figure 2 shows segmentation and depth predictions reconstructed from the compressed tokens, while Figure 3 visualizes the Perceiver's cross-attention with and without auxiliary supervision. The re-



Figure 2: **Auxiliary heads recover dense structure from compressed tokens.** Segmentation (columns 2 and 3) and depth (columns 4 and 5) reconstructed from the 20 image (left) and 200 scene tokens (right); both branches retain object boundaries and depth ordering despite the compression.

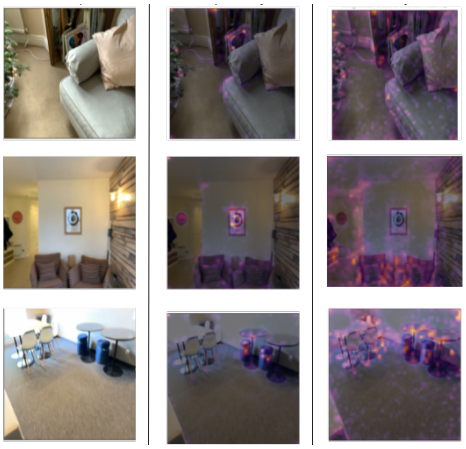

Figure 3: **Perceiver cross-attention summed over latent queries.** Captioning alone (second column) concentrates on a few salient regions; the auxiliary head (third column) shows how the attention spread across surfaces and boundaries.

constructed outputs retain coherent object boundaries and depth ordering, and the attention maps show a broader distribution across objects, surfaces, and boundaries after auxiliary training. These qualitative observations are consistent with the frozen-token probe and downstream improvements.

4.2 Complementary Dual-Encoder Architecture

Each group of models in the table 1 share the 0.8B language backbone, the two-stage training recipe, and auxiliary supervision, and differ only in the visual encoder and compression module. The DUNE-only model represents a 32-view scene with 640 tokens (20 per view), while the MapAnything-only model compresses the whole scene into just 200 tokens. Despite this 3.2× smaller budget, MapAnything performs better on the two multi-view benchmarks, improving over DUNE by 2.7 points on VSI-Bench-Debiased and 3.7 on ReVSI. Conversely, DUNE is stronger on single-image questions, with gains of 18.0 points on CV-Bench 2D and 3.4 on CV-Bench 3D. These results reflect the different information each encoder captures: MapAnything provides a compact, globally consistent metric representation of the scene, whereas DUNE retains the localized semantic and appearance cues needed for object-centric questions. Combining the DUNE image branch with the MapAnything scene branch produces 640 per-view tokens and 200 scene tokens, for a total of 840 visual tokens in the 32-view setting. The combined model improves over the stronger individual branch on every benchmark: by 3.3 points on VSI-Bench-Debiased, 1.2 on ReVSI, 9.0 on CV-Bench 2D, and 4.9 on CV-Bench 3D. The gain is therefore not explained solely by selecting one stronger encoder. Instead, the global metric representation from MapAnything and the localized per-view representation from DUNE provide complementary information.

4.3 Comparison with Existing Models

Table 3 compares GeoPACT with general-purpose and spatially specialized VLMs.

To fairly compare with exiting methods, RoboBrain (RoboBrainTeam et al., 2025) or VLM3R (Fan et al., 2026), we took the provided checkpoints and evaluated the models ourselves under the exact same VLMEvalKit evaluation protocol as GeoPACT which might explain the difference with published results for these models. As Table 3 show in spite of being larger, they are outperformed by GeoPACT on VSI-Bench-Debiased, ReVSI, CV-Bench 2D, and SPAR, while VLM3R and RoboBrain2.0 obtain higher CV-Bench 3D scores. These comparisons contextualize the effectiveness of the compact architecture; the principal evidence for our learning scheme remains the controlled with-versus-without-supervision comparison above.

Table 3: **Comparison to other baselines.** All models re-evaluated under one same VLMEvalKit protocol (which might explain the difference with published results). #Tok.: visual tokens fed to the LLM; Lat.: per-example inference latency (A100 80GB, bf16, batch size 1). GeoPACT remain competitive with 8–15× fewer visual tokens while being 2–4.6× faster.

Model	Params	VSI-deb.	ReVSI	CVB-2D	CVB-3D	SPAR	Avg.	#Tok.	Lat. (s)
Qwen3.5-0.8B	0.8B	29.0	26.8	62.1	64.7	58.1	48.1	9600	1.1
Qwen3.5-2B	2B	35.2	34.0	70.6	74.1	61.3	55.0	9600	1.4
Qwen3.5-0.8B-FT	0.8B	59.5	48.8	71.2	79.1	57.1	63.1	9600	1.1
RoboBrain2.0-7B	7B	33.9	39.4	70.1	74.5	42.4	52.1	12512	5.1
VLM3R-7B	7B	53.3	42.9	70.3	71.2	42.2	55.9	6720	2.2
GeoPACT (ours)	0.8B	**62.1**	**48.4**	**72.8**	70.2	**62.1**	**63.7**	**840**	**1.1**

We also compare the results with Qwen3.5-0.8B-FT, which uses the same language backbone and Stage-2 data while retaining Qwen's native visual interface. It allows to disentagle the effect of Stage-2 data created for bette spatial understanding, from the effect of native geometry understanding in vision encoders.

The two models obtain similar five-benchmark average (63.1 vs. 63.7) and have comparable end-to-end latency. GeoPACT performs better on VSI-Bench-Debiased (62.1 vs. 59.5) , and SPAR (62.1 vs. 57.1), but substantially worse on CV-Bench 3D (79.1 vs. 70.2). The results on CV-Bench 2D (72.8 vs. 71.2), and ReVSI (48.8 vs. 48.4) are comparable. Table 1 breaks down the some of the results across different categories. Thus, we see that most gains on VSI-Bench-Debiased are due to the metric-estimation tasks, while tasks requiring more semantic scene understanding (eg. Appeareance order, Route Planning) are much stronger in Qwen3.5-0.8B-FT model. This could be explained by the fact that initial Qwen-3.5-0.8B vision encoder representations are much better aligned with its' language space compared to DUNE or MapAnything encoders. We believe that the current gap could be reduced by more extensive stage 1 training.

The primary efficiency advantage of GeoPACT is its visual context: it uses 840 visual tokens rather than 9,600, an 11.4× reduction, rather than lower end-to-end latency. The latency gains can be increased with the LLM model size, or amount of frames model receives as an input.

Inference efficiency. The auxiliary heads are removed after Stage 1 and add no inference cost. For 32 views, GeoPACT uses 840 visual tokens, compared with 6,720 for VLM3R and 12,512 for RoboBrain2.0. Its 1.1 s end-to-end latency matches Qwen3.5-0.8B because MapAnything dominates the vision-encoding cost, but it remains approximately 2× faster than VLM3R and 4.6× faster than RoboBrain2.0. A complete runtime breakdown is provided in Appendix F.

5 Conclusion

We have shown in this paper that spatial reasoning can fail before information even reaches the language model: at the visual compression interface. When trained only through a language objective, this interface preserves salient semantic cues but discards much of the local and metric structure encoded by the vision backbone. Directly supervising compressed tokens with segmentation and depth objectives changes what the compressor retains, consistently improving spatial reasoning. Frozen-token probes and attention analysis confirm that the gains arise from more accessible spatial information and broader scene coverage. The approach requires neither human annotations nor additional inference cost: supervision is obtained from perception decoders already present in the pipeline, and the auxiliary heads are removed after pretraining. The compression interface is not merely an efficiency mechanism—it determines what the language model is allowed to see. Training it to preserve geometry offers a simple and practical path toward more spatially capable VLMs.

Limitations. Our resource-constrained training is intended to validate the proposed supervision scheme rather than establish its performance ceiling. The auxiliary targets may inherit errors from their frozen decoders and increase training cost, although they add no inference overhead. We also leave evaluation in closed-loop embodied tasks to future work.

AI Use Disclosure

We used generative AI tools to assist with writing and improving software code that the authors fully understood, reviewed, and tested; to polish the language and readability of the manuscript. We also used generative AI to support data annotation, specifically scene and object captioning, as described in detail in Appendix Section C. Finally, we used AI-assisted search tools to identify potentially relevant gaps in related work and ensure that important references were not overlooked; all citations were incorporated manually.

All AI-assisted outputs were reviewed and validated by the authors, who take full responsibility for the final content of the paper, including its text, code, annotations, figures, and scientific claims.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning, 2022a. URL https://arxiv.org/abs/2204.14198. 1

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a Visual Language Model for Few-Shot Learning. In *NeurIPS*, 2022b. 2

Fabien Baradel, Matthieu Armando, Salma Galaaoui, Romain Brégier, Philippe Weinzaepfel, Grégory Rogez, and Thomas Lucas. Multi-HMR: Multi-Person Whole-Body Human Mesh Recovery in a Single Shot. In *ECCV*, 2024. 4

Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, and Elad Shulman. ARKitScenes: A Diverse Real-World Dataset for 3D Indoor Scene Understanding Using Mobile RGB-D Data. In *NeurIPS*, 2021. 1

Mahtab Bigverdi, Zelun Luo, Cheng-Yu Hsieh, Ethan Shen, Dongping Chen, Linda G. Shapiro, and Ranjay Krishna. Perception tokens enhance visual reasoning in multimodal language models. arXiv:2412.03548, 2024. 3

Junbum Cha, Wooyoung Kang, Jonghwan Mun, and Byungseok Roh. Honeybee: Locality-Enhanced Projector for Multimodal LLM. In *CVPR*, 2024. 3

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brain Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14455–14465, June 2024a. 3

Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. An Image Is Worth 1/2 Tokens After Layer 2: Plug-and-Play Inference Acceleration for Large Vision-Language Models. In *ECCV*, 2024b. 2

Shiqi Chen, Tongyao Zhu, Ruochen Zhou, Jinghan Zhang, Siyang Gao, Juan Carlos Niebles, Mor Geva, Junxian He, Jiajun Wu, and Manling Li. Why Is Spatial Reasoning Hard for Vlms? an Attention Mechanism Perspective on Focus Areas. In *ICML*, 2025. 3

Xi Chen, Xiao Wang, Soravit Changpinyo, Anthony J Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022. 1

Jang Hyun Cho, Boris Ivanovic, Yulong Cao, Edward Schmerling, Yue Wang, Xinshuo Weng, Boyi Li, Yurong You, Philipp Krähenbühl, Yan Wang, and Marco Pavone. Language-Image Models with 3D Understanding. arXiv:2405.03685, 2024. 3

Ian Connick Covert, Tony Sun, James Zou, and Tatsunori Hashimoto. Locality Alignment Improves Vision-Language Models. In *ICLR*, 2025. 3

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. 1

Nianchen Deng, Lixin Gu, Shenglong Ye, Yinan He, Zhe Chen, Songze Li, Haomin Wang, Xingguang Wei, Tianshuo Yang, Min Dou, Tong He, Wenqi Shao, Kaipeng Zhang, Yi Wang, Botian Shi, Yanting Zhang, Jifeng Dai, Yu Qiao, Hongjie Zhang, and Wenhai Wang. Internspatial: A comprehensive dataset for spatial reasoning in vision-language models, 2025. URL https://arxiv.org/abs/2506.18385. 3

Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 11198–11201, 2024. 6

Zhiwen Fan, Jian Zhang, Renjie Li, Junge Zhang, Runjin Chen, Hezhen Hu, Kevin Wang, Huaizhi Qu, Shijie Zhou, Dilin Wang, Zhicheng Yan, Hongyu Xu, Justin Theiss, Tianlong Chen, Jiachen Li, Zhengzhong Tu, Zhangyang Wang, and Rakesh Ranjan. Vlm-3r: Vision-language models augmented with instruction-aligned 3d reconstruction, 2026. URL https://arxiv.org/abs/2505.20279. 3, 6, 8, 1

GemmaTeam, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025. 1

Wenbo Hu, Zi-Yi Dou, Liunian Harold Li, Amita Kamath, Nanyun Peng, and Kai-Wei Chang. Matryoshka Query Transformer for Large Vision-Language Models. In *NeurIPS*, 2026. 2

Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 4651–4664. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/jaegle21a.html. 2

Jitesh Jain, Zhengyuan Yang, Humphrey Shi, Jianfeng Gao, and Jianwei Yang. Elevating Visual Perception in Multimodal LLMs with Visual Embedding Distillation. In *NeurIPS*, 2025. 3

Baoxiong Jia, Yixin Chen, Huangyue Yu, Yan Wang, Xuesong Niu, Tengyu Liu, Qing Li, and Siyuan Huang. SceneVerse: Scaling 3D Vision-Language Learning for Grounded Scene Understanding. In *ECCV*, 2024. 1

Nikhil Keetha, Norman Müller, Johannes Schönberger, Lorenzo Porzi, Yuchen Zhang, Tobias Fischer, Arno Knapitsch, Duncan Zauss, Ethan Weber, Nelson Antunes, Jonathon Luiten, Manuel Lopez-Antequera, Samuel Rota Bulò, Christian Richardt, Deva Ramanan, Sebastian Scherer, and Peter Kontschieder. MapAnything: Universal feed-forward metric 3D reconstruction. In *International Conference on 3D Vision (3DV)*. IEEE, 2026. 1, 4

Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham Kakade, Prateek Jain, and Ali Farhadi. Matryoshka Representation Learning. In *NeurIPS*, 2022. 2

Xuanyu Lei, Zonghan Yang, Xinrui Chen, Peng Li, and Yang Liu. Scaffolding Coordinates to Promote Vision-Language Coordination in Large Multi-Modal Models. arXiv:2402.12058, 2024. 3

Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding Image Matching in 3D with MASt3R. In *ECCV*, 2024. 4

Chengzu Li, Wenshan Wu, Huanyu Zhang, Yan Xia, Shaoguang Mao, Li Dong, Ivan Vuli'c, and Furu Wei. Imagine While Reasoning in Space: Multimodal Visualization-of-Thought. arXiv:2501.07542, 2025a. 3

Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. BLIP: Bootstrapping Language-Image Pre-Training for Unified Vision-Language Understanding and Generation. In *ICML*, 2022. 1

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models. In *ICML*, 2023. 2

Rong Li, Shijie Li, Lingdong Kong, Xulei Yang, and Junwei Liang. Seeground: See and Ground for Zero-Shot Openvocabulary 3D Visual Grounding. In *CVPR*, 2025b. 3

Wentong Li, Yuqian Yuan, Jian Liu, Dongqi Tang, Song Wang, Jie Qin, Jianke Zhu, and Lei Zhang. Token-Packer: Efficient Visual Projector for Multimodal LLM. arXiv:2407.02392, 2024. 2

Yuan-Hong Liao, Rafid Mahmood, Sanja Fidler, and David Acuna. Reasoning Paths with Reference Objects Elicit Quantitative Spatial Reasoning in Large Vision-Language Models. arXiv:2409.09788, 2024. 3

Benlin Liu, Yuhao Dong, Yiqin Wang, Zixian Ma, Yansong Tang, Luming Tang, Yongming Rao, Wei-Chiu Ma, and Ranjay Krishna. Coarse Correspondences Boost Spatial-Temporal Reasoning in Multimodal Language Model. arXiv:2408.00754, 2024. 3

Dingning Liu, Cheng Wang, Peng Gao, Renrui Zhang, Xinzhu Ma, Yuan Meng, and Zhihui Wang. 3DAxis-Prompt: Promoting the 3D Grounding and Reasoning in GPT-4o. arXiv:2503.13185, 2025a. 3

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual Instruction Tuning. In *NeurIPS*, 2023. 2

Yang Liu, Ming Ma, Xiaomin Yu, Pengxiang Ding, Han Zhao, Mingyang Sun, Siteng Huang, and Donglin Wang. SSR: Enhancing Depth Perception in Vision-Language Models via Rationale-Guided Spatial Reasoning. arXiv:2505.12448, 2025b. 3

Yuecheng Liu, Dafeng Chi, Shiguang Wu, Zhanguang Zhang, Yaochen Hu, Lingfeng Zhang, Yingxue Zhang, Shuang Wu, Tongtong Cao, Guowei Huang, Helong Huang, Guangjian Tian, Weichao Qiu, Xingyue Quan, Jianye Hao, and Yuzheng Zhuang. SpatialCoT: Advancing Spatial Reasoning Through Coordinate Alignment and Chain-of-Thought for Embodied Task Planning. arXiv:2501.10074, 2025c. 3

Wufei Ma, Yu-Cheng Chou, Qihao Liu, Xingrui Wang, Celso de Melo, Jianwen Xie, and Alan Yuille. SpatialReasoner: Towards Explicit and Generalizable 3D Spatial Reasoning. arXiv:2504.20024, 2025. 3

Yongsen Mao, Yiming Zhang, Hanxiao Jiang, Angel X. Chang, and Manolis Savva. MultiScan: Scalable RGBD Scanning for 3D Environments with Articulated Object. In *NeurIPS*, 2022. 1

Chancharik Mitra, Brandon Huang, Trevor Darrell, and Roei Herzig. Compositional Chain-of-Thought Prompting for Large Multimodal Models. arXiv:2311.17076, 2023. 3

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning Robust Visual Features without Supervision. *Transactions on Machine Learning Research* , 3(1), 2024. 4

Kun Ouyang, Yuanxin Liu, Haoning Wu, Yi Liu, Hao Zhou, Jie Zhou, Fandong Meng, and Xu Sun. SpaceR: Reinforcing MLLMs in Video Spatial Reasoning. arXiv:2504.01805, 2025. 3

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023. 1

Yiming Qin, Bomin Wei, Jiaxin Ge, Konstantinos Kallidromitis, Stephanie Fu, Trevor Darrell, and XuDong Wang. Chain-of-visual-thought: Teaching vlms to see and think better with continuous visual tokens, 2025. URL https://arxiv.org/abs/2511.19418. 3

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5. 3, 6

QwenTeam. Qwen3.5: Accelerating productivity with native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5. 2

RoboBrainTeam, Mingyu Cao, Huajie Tan, Yuheng Ji, Xiansheng Chen, Minglan Lin, Zhiyu Li, Zhou Cao, Pengwei Wang, Enshen Zhou, Yi Han, Yingbo Tang, Xiangqi Xu, Wei Guo, Yaoxu Lyu, Yijie Xu, Jiayu Shi, Mengfei Du, Cheng Chi, Mengdi Zhao, Xiaoshuai Hao, Junkai Zhao, Xiaojie Zhang, Shanyu Rong, Huaihai Lyu, Zhengliang Cai, Yankai Fu, Ning Chen, Bolun Zhang, Lingfeng Zhang, Shuyi Zhang, Dong Liu, Xi Feng, Songjing Wang, Xiaodan Liu, Yance Jiao, Mengsi Lyu, Zhuo Chen, Chenrui He, Yulong Ao, Xue Sun, Zheqi He, Jingshu Zheng, Xi Yang, Donghai Shi, Kunchang Xie, Bochao Zhang, Shaokai Nie, Chunlei Men, Yonghua Lin, Zhongyuan Wang, Tiejun Huang, and Shanghang Zhang. Robobrain 2.0 technical report, 2025. URL https://arxiv.org/abs/2507.02029. 6, 8

Mert Bulent Sariyildiz, Philippe Weinzaepfel, Thomas Lucas, Pau de Jorge, Diane Larlus, and Yannis Kalantidis. Dune: Distilling a universal encoder from heterogeneous 2d and 3d teachers, 2025. URL https://arxiv.org/abs/2503.14405. 1, 4

Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. LLaVA-PruMerge: Adaptive Token Reduction for Efficient Large Multimodal Models. arXiv:2403.15388, 2026. 2

Yifan Shen, Yuanzhe Liu, Jingyuan Zhu, Xu Cao, Xiaofeng Zhang, Yixiao He, Wenming Ye, James Matthew Rehg, and Ismini Lourentzou. Fine-Grained Preference Optimization Improves Spatial Reasoning in VLMs. arXiv:2506.21656, 2025. 3

Shun Taguchi, Hideki Deguchi, Takumi Hamazaki, and Hiroyuki Sakai. Spatialprompting: Keyframe-Driven Zero-Shot Spatial Reasoning with Off-the-Shelf Multimodal Large Language Models. arXiv:2505.04911, 2025. 3

Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Ziteng Wang, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: a Fully Open, Vision-Centric Exploration of Multimodal LLMs. In *NeurIPS*, 2024. 2

Johanna Wald, Armen Avetisyan, Nassir Navab, Federico Tombari, and Matthias Nießner. RIO: 3D Object Instance Re-Localization in Changing Indoor Environments. In *ICCV*, 2019. 1

Fudong Wang, Jiajia Liu, Jingdong Chen, Jun Zhou, Kaixiang Ji, Lixiang Ru, Qingpei Guo, Ruobing Zheng, Tianqi Li, et al. M2-Reasoning: Empowering MLLMs with Unified General and Spatial Reasoning. arXiv:2507.08306, 2025a. 3

Guangzhi Wang, Yixiao Ge, Xiaohan Ding, Mohan Kankanhalli, and Ying Shan. What Makes for Good Visual Tokenizers for Large Language Models. arXiv:2305.12223, 2023. 3

Haochen Wang, Anlin Zheng, Yucheng Zhao, Tiancai Wang, Zheng Ge, Xiangyu Zhang, and Zhaoxiang Zhang. Reconstructive Visual Instruction Tuning. arXiv:2410.09575, 2025b. 3

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-Of-thought Prompting Elicits Reasoning in Large Language Models. In *NeurIPS*, 2022. 3

Luis Wiedmann, Orr Zohar, Amir Mahla, Xiaohan Wang, Rui Li, Thibaud Frere, Leandro von Werra, Aritra Roy Gosthipaty, and Andres Marafioti. FineVision: Open Data Is All You Need. arXiv:2510.17269, 2025. 5, 1

Junfei Wu, Jian Guan, Kaituo Feng, Qiang Liu, Shu Wu, Liang Wang, Wei Wu, and Tieniu Tan. Reinforcing Spatial Reasoning in Vision-Language Models with Interwoven Thinking and Visual Drawing. arXiv:2506.09965, 2025. 3

Jihan Yang, Shusheng Yang, Anjali Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in Space: How Multimodal Large Language Models See, Remember and Recall Spaces. In *CVPR*, 2025a. 2, 6

Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Li Fei-Fei, and Sainingg Xie. Thinking in Space: How Multimodal Large Language Models See, Remember, and Recall Spaces. In *CVPR*, 2025b. 1

Lingfeng Yang, Yueze Wang, Xiang Li, Xinlong Wang, and Jian Yang. Fine-Grained Visual Prompting. In *NeurIPS*, 2023. 3

Senqiao Yang, Yukang Chen, Zhuotao Tian, Chengyao Wang, Jingyao Li, Bei Yu, and Jiaya Jia. VisionZip: Longer Is Better But Not Necessary in Vision Language Models. arXiv:2412.04467, 2026. 2

Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis Brown, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhan Wang, Danhao Lu, Rob Fergus, Yann LeCun, Li Fei-Fei, and Saining Xie. Cambrian-s: Towards spatial supersensing in video. *arXiv preprint arXiv:2511.04670*, 2025c. 6

Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis Brown, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhan Wang, Daohan Lu, Rob Fergus, Yann LeCun, Li Fei-Fei, and Saining Xie. Cambrian-S: Towards Spatial Supersensing in Video. arXiv:2511.04670, 2025d. 1

Yuncong Yang, Jiageng Liu, Zheyuan Zhang, Siyuan Zhou, Reuben Tan, Jianwei Yang, Yilun Du, and Chuang Gan. Mindjourney: Test-Time Scaling with World Models for Spatial Reasoning. arXiv:2507.12508, 2025e. 3

Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. ScanNet++: A High-Fidelity Dataset of 3D Indoor Scenes. In *ICCV*, 2023. 1

Jiahui Zhang, Yurui Chen, Yanpeng Zhou, Yueming Xu, Ze Huang, Jilin Mei, Junhui Chen, Yu-Jie Yuan, Xinyue Cai, Guowei Huang, Xingyue Quan, Hang Xu, and Li Zhang. From Flatland to Space: Teaching Vision-Language Models to Perceive and Reason in 3D. arXiv:2503.22976, 2025. 3, 6, 1

Yiming Zhang, Jiacheng Chen, Jiaqi Tan, Yongsen Mao, Wenhu Chen, and Angel X. Chang. Revsi: Rebuilding visual spatial intelligence evaluation for accurate assessment of vlm 3d reasoning. *arXiv preprint arXiv:2604.24300*, 2026. 6

Appendix

A Implementation Details.

Experimental setup. Both stages train with AdamW (β=[0.9, 0.95], weight decay 0.01, none on norms and biases), a single global learning rate of 10^{-4} under a warmup-cosine schedule ($\approx$400 warmup steps, floor $0.01\times$), and bf16 mixed precision. Input resolution is 672×672 for the image branch and 518×518 for the geometric branch (patch size 14 in both). Stage 2 training is performed using 8 Nvidia A100 GPUs for 100 epochs, while Stage 1 is a single epoch consisting of the entire Stage 1 dataset.

B Evaluation Protocol

We evaluate all models using VLMEvalKit under a shared inference and scoring protocol. Unless otherwise stated, we use the official benchmark prompts and answer-processing rules distributed with VLMEvalKit. For VSI-Bench and ReVSI, we sample 32 frames per scene using the same frame-selection procedure for every model. Single-image benchmarks are evaluated at each model's native input resolution.

Answer extraction. For multiple-choice questions, the model is instructed to return the selected option, and predictions are normalized using the benchmark-specific VLMEvalKit answer extractor. A prediction is counted as correct when the extracted option matches the reference answer. For numerical questions, units and formatting are normalized before evaluation, and performance is measured using mean relative accuracy following the official VSI-Bench protocol. Free-form answers are evaluated using the official benchmark rules described below.

LLM-based judging. For benchmarks requiring semantic evaluation of free-form predictions, we use the VLMEvalKit implementation with Gemma3-27B (GemmaTeam et al., 2025) as the judge. The judge receives the original question, the reference answer, and the model prediction through the unmodified VLMEvalKit evaluation template.

C Training Data and Dataset Construction.

FineVision 2D image captioning data. is a subset of FineVision (Wiedmann et al., 2025), large-scale open VLM training corpus, from which we retain 7M real images paired with captions or descriptions while excluding charts, mathematics, OCR, VQA, *etc.* .

VSI-590k and VLM-3R. These datasets, introduced by Yang et al. (2025d) and Fan et al. (2026) contain 590K and 340K video QA pairs respectively. Both follow the question structure of VSI-Bench (Yang et al., 2025b) and are commonly used for instruction fine-tuning on that benchmark.

SPAR-7M-RGBD. (Zhang et al., 2025) is a 33-task spatial suite built on ScanNet, ScanNet++, and Structured3D. We retain only the single- and multi-view splits of these three partitions, discarding the 10 video-based tasks.

Templated QA. Our templated spatial reasoning QA data is built on top of five 3D indoor scan datasets: ScanNet (Dai et al., 2017), ScanNet++ (Yeshwanth et al., 2023), ARKitScenes (Baruch et al., 2021), 3RScan (Wald et al., 2019), and MultiScan (Mao et al., 2022). Each question is generated from an instance-labeled 3D mesh together with video frames of the same scene. Except for ScanNet++, which already provides images aligned to its labeled mesh, we take the instance-labeled meshes from SceneVerse (Jia et al., 2024) and align the video frame poses to them. From the resulting collection of roughly 4.6K scenes we sample nearly 100K sub-scenes of 3–15 frames each, growing every sub-scene by iteratively adding frames that have sufficient overlap with at least one already selected frame.

For each sub-scene we then extract object-level 3D information from the mesh: position and size from a 3D oriented bounding box, per-view 2D bounding boxes, rendered segmentation masks, and

Table 4: **Cross-view patch correspondence, frozen encoders, no training.** For each patch in view A we ask whether its nearest neighbour in feature space is the patch containing the same 3D point in view B, established from ground-truth poses and depth with an occlusion check. PCK@k: accuracy within k patches of the ground-truth location. Pairs are binned by co-visibility; lower bins are wider baselines. All feature maps are resampled to a common 32×32 grid so the comparison is independent of native patch size. DUNE leads in every bin at every threshold despite being $3.5\times$ smaller than the ViT-L encoders, and its margin is largest at the widest baselines.

	SigLIP2-L 316M	DINOv3-B 85.7M	DINOv3-L 303M	DUNE **86.3M**
Co-visibility 0.2–0.4				
PCK@1	24.8	35.5	34.9	**42.9**
PCK@2	35.5	42.3	40.0	**59.6**
PCK@3	43.0	47.0	59.1	**67.3**
Co-visibility 0.4–0.6				
PCK@1	31.2	45.9	43.7	**51.4**
PCK@2	43.1	54.4	61.4	**68.0**
PCK@3	50.9	60.2	69.5	**74.9**
Co-visibility 0.6–0.8				
PCK@1	38.5	58.8	55.1	**62.6**
PCK@2	51.1	67.3	72.5	**78.5**
PCK@3	59.3	73.4	80.5	**84.7**

pairwise object distances measured between nearest points. We prompt a strong VLM (Qwen3.5 27B (QwenTeam, 2026)) with the frame set, marking the target object with 2D bounding boxes across views to obtain descriptive captions and referring expressions. A VLM judge is used to filter these annotations on object identifiability, bounding-box consistency, and presence of distractors. The pipeline yields approximately 1.5M object-level descriptions covering object appearance, attributes, and local context.

Scene captioning (3D). For each of the 100K sub-scenes extracted for Templated QA, we additionally prompt the same VLM (Qwen 3.5 27B) for a set of detailed scene descriptions, each targeting one of eight prescribed aspects: a general description; room geometry; relationships between objects; object orientations; spatial descriptions from imagined viewpoints; the functional or zonal organization of the room; salient objects; and object affordances, *i.e.*the actions the objects support.

D Encoder Selection.

Our recipe imposes two requirements that narrow the choice of backbone. The image branch must supply features that are consistent across views, since relations between objects seen in different frames cannot be resolved per-image, and it must expose a dense decoder whose predictions can serve as per-patch targets without any additional supervised training; otherwise the auxiliary supervision would need an external teacher, which is what the recipe is designed to avoid. The scene branch must produce metric output, because quantities like "about 1.5 meters" cannot be recovered from a scale-normalised reconstruction at any level of feature quality.

On a training-free cross-view correspondence test (Tab. 4), DUNE outperforms SigLIP2 and DINOv3 in every co-visibility bin despite being 3.5× smaller, with the largest margin at the widest baselines, and its dense segmentation decoder supplies the per-patch targets of Sec. 4.1. For the scene branch, scale-invariant reconstructors such as VGGT and π^3 cannot provide metric scale, whereas MapAnything predicts metric geometry in a single feed-forward pass over the whole view set, so it can be computed once per scene and reused across questions. Its metric-depth head also serves as the teacher for the scene branch's auxiliary supervision.

E Visual-Token Budget

In Table 5 we study how the number of learned Perceiver queries affects the trade-off between representation capacity and visual sequence length. These experiments use a reduced training mixture

Table 5: **Visual-token budget ablation.** VSI-Bench-Debiased accuracy under a reduced training mixture. Scene tokens are varied with the geometric branch, while per-view tokens are varied with the image branch.

Branch	Visual tokens	VSI-Debiased
Geometric	100 per scene	32.3
	200 per scene	**38.0**
	500 per scene	34.9
Image	**20 per view**	**35.9**
	50 per view	34.1

Table 6: **Inference cost**, averaged over 100 VSI-Bench-Debiased examples at 32 views; identical examples and hardware (A100 80GB, fp16, batch size 1). †DiViNE 111 ms + MapAnything 854 ms. ‡Small-Qwen LLM latency is dominated by variable generation length and has high variance (std ≈ mean). Auxiliary heads are discarded after Stage 1 and add no cost. GeoPACT feeds 8–15× fewer visual tokens than the other spatial VLMs; the generalist Qwen models are comparably fast only because they run a far smaller vision encoder while scoring far lower on spatial accuracy, whereas the specialized 7B systems (VLM-3R, RoboBrain) that approach our accuracy are 2–4.6× slower.

	GeoPACT	Qwen3.5-0.8B	Qwen3.5-2B	Qwen3.5-9B	VLM-3R-7B	RoboBrain-7B
Vision encoding (ms)	965.3†	114.9	260.8	356.8	1433.6	4021.3
Compression / proj. (ms)	48.5	1.4	2.2	3.4	12.1	6.3
LLM forward (ms)	78.0	970.7‡	907.1‡	2572.9	546.2	979.7
Total (ms)	**1094.7**	1136.6‡	1421.6‡	3033.1	2164.5	5078.8
Visual tokens	**840**	9600	9600	9600	6720	12512
LLM backbone	0.8B	0.8B	2B	9B	7B	7B

and are intended only to select the token budget; their absolute scores are therefore not directly comparable with the main results. Within each branch, we vary only the number of Perceiver queries while keeping the encoder, projector architecture, training schedule, and evaluation protocol fixed.

For the geometric branch, increasing the budget from 100 to 200 scene tokens improves VSI-Bench-Debiased from 32.3 to 38.0. Increasing it further to 500 tokens reduces performance to 34.9, despite the larger representation capacity. For the image branch, 20 tokens per view achieves 35.9, compared with 34.1 using 50 tokens per view, while reducing the per-view sequence length by 2.5×. We therefore use 200 scene tokens and 20 tokens per view in all main experiments.

These results indicate that performance is not monotonic in the number of visual tokens. A larger query bank does not necessarily produce a more useful representation for the language model: the additional tokens must also be trained to select, organize, and communicate relevant information through the compression interface. Under our training setting, the intermediate scene budget and smaller per-view budget provide the best balance between spatial information retention and sequence length.

F Inference cost Breakdown

We report a detailed breakdown of the inference cost in Table 6. All models are evaluated on the same 100 examples from VSI-Bench-Debiased using 32 input views, batch size 1, and a single NVIDIA A100 80GB GPU. Latency is divided into three components: vision encoding, visual-token compression or projection, and language-model inference. The auxiliary prediction heads are not included because they are discarded after Stage 1 and are never executed at inference.

GeoPACT uses only 840 visual tokens, compared with 6,720 for VLM3R and 12,512 for RoboBrain2.0. Its compressed visual sequence is therefore 8–15× shorter than those of the specialized 7B spatial models. This substantially reduces the language-model cost: despite using two visual encoders, GeoPACT requires only 78.0,ms for language-model inference. Its total latency is approximately 2× lower than VLM3R and 4.6× lower than RoboBrain2.0.

The main cost of GeoPACT lies in vision encoding rather than language-model inference. Of its 965.3,ms vision-encoding time, DUNE accounts for 111,ms and MapAnything for 854,ms. We retain the MapAnything branch because it supplies the metrically grounded scene representation responsible for substantial improvements on the multi-view spatial benchmarks. Moreover, MapAnything and the visual compression modules depend only on the scene, not on the question. Their outputs can therefore be cached when multiple questions are asked about the same scene, leaving only the language-model inference cost for each additional query.