PanoWorld: Towards Spatial Supersensing in 360° Panorama World

Changpeng Wang[1], **Xin Lin**[2], **Junhan Liu**[1], **Yuheng Liu**[3], **Zhen Wang**[1], **Donglian Qi**[1], **Yunfeng Yan**[1], **Xi Chen**[4]

[1]Zhejiang University [2]University of California, San Diego
[3]University of California, Irvine [4]The University of Hong Kong

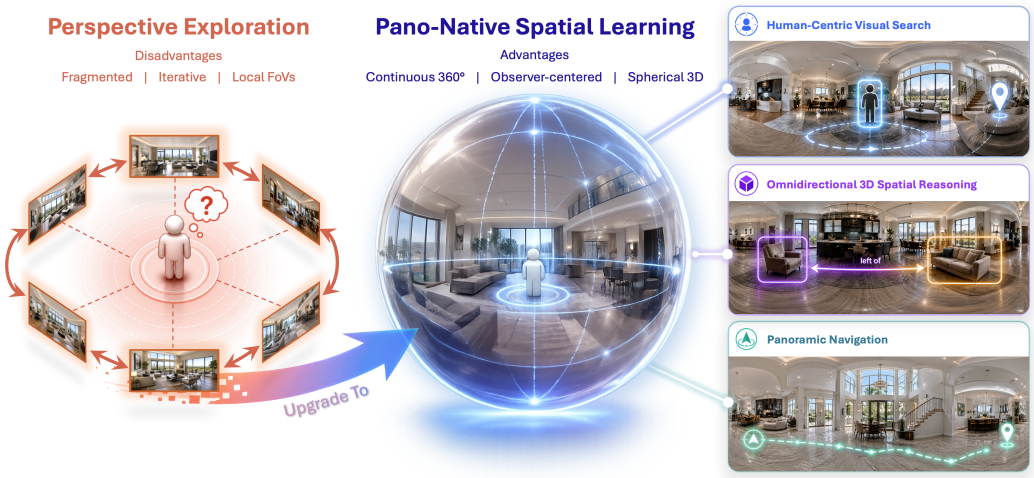


Figure 1: Existing MLLMs reason over fragmented local views, making it difficult to associate spatial cues in 360°. We introduce pano-native supersensing, which teaches VLMs to perceive and reason directly over 360° panoramas, providing a unified full-surround representation for downstream tasks such as human-centric visual search, omnidirectional 3D spatial reasoning, and panoramic navigation.

Abstract

Multimodal large language models (MLLMs) still struggle with spatial understanding under the dominant perspective-image paradigm, which inherits the narrow field of view of human-like perception. For navigation, robotic search, and 3D scene understanding, 360° panoramic sensing offers a form of supersensing by capturing the entire surrounding environment at once. However, existing MLLM pipelines typically decompose panoramas into multiple perspective views, leaving the spherical structure of equirectangular projection (ERP) largely implicit. In this paper, we study pano-native understanding, which requires an MLLM to reason over an ERP panorama as a continuous, observer-centered space. To this end, we first define the key abilities for pano-native understanding, including semantic anchoring, spherical localization, reference-frame transformation, and depth-aware 3D spatial reasoning. We then build a large-scale metadata construction pipeline that converts mixed-source ERP panoramas into geometry-aware, language-grounded, and depth-aware supervision, and instantiate these signals as capability-aligned instruction tuning data. On the model side, we introduce PanoWorld with Spherical Spatial Cross-Attention, which injects spherical geometry into the visual stream. We further construct PanoSpace-Bench, a diagnostic benchmark for evaluating ERP-native spatial reasoning. Experiments show that

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

PanoWorld substantially outperforms both proprietary and open-source baselines on PanoSpace-Bench, H*Bench, and R2R-CE Val-Unseen benchmarks. These results demonstrate that robust panoramic reasoning requires dedicated pano-native supervision and geometry-aware model adaptation. All source code and proposed data will be publicly released at `https://wcpcp.github.io/PanoWorld`.

1 Introduction

Recent multimodal large language models (MLLMs) have made substantial progress in perspective-image visual understanding, yet robust spatial reasoning remains challenging [39, 36, 48, 34]. A key limitation of this paradigm is that it inherits the limited instantaneous field of view of human-like perception, whereas tasks such as human-centric visual search, navigation, and immersive scene understanding benefit from full-surround environmental awareness. 360° panoramic sensing therefore offers a form of supersensing, expanding spatial perception from local views to the entire observer-centered environment. Despite this potential, current approaches often address full-surround reasoning through sequential perspective exploration, as highlighted by recent efforts [52], where a continuous panorama is decomposed into local perspective views to simulate human-like exploration of the surrounding 3D environment. This naturally raises the question of whether 360° spatial reasoning can be modeled more directly and efficiently from panoramic representations themselves, which encode globally consistent observer-centered scenes, including wrap-around continuity and viewpoint-dependent spatial relations.

However, directly transferring existing perspective models to panoramic understanding is challenging, since panoramic and perspective images exhibit fundamental representation gaps, including geometric distortion, non-uniform spatial sampling, and boundary discontinuities [24]. Although some existing approaches in other panoramic tasks utilize perspective-transfer pipelines through projection and stitching [3, 42, 12, 30, 62], recent progress suggests that panorama-specific models trained on large-scale panoramic data can achieve stronger performance [25, 12], which highlights the importance of large-scale panoramic supervision for learning pano-native representations. These observations suggest that panoramic MLLMs should likewise move beyond perspective transfer toward panorama-specific modeling. Moreover, to enable a unified MLLM capable of handling diverse spatial reasoning tasks, it is essential to systematically characterize the capabilities. However, existing efforts on panoramic MLLMs are typically organized around individual tasks or benchmarks [63, 57, 49, 10], and thus remain fragmented and incomplete, lacking both a systematic understanding of the required capabilities and a unified benchmark to evaluate them.

To address these limitations, we propose a unified pano-native spatial learning framework for MLLMs that learns an observer-centered representation of the 360° surrounding environment. We begin by introducing a capability-based formulation of panoramic spatial understanding, decomposing it into key components such as spherical localization, relative direction reasoning, viewpoint transformation, 3D spatial relations, and global scene topology. Building on this formulation, we develop a metadata-driven pipeline for scalable panoramic data construction and build a comprehensive benchmark that evaluates these capabilities beyond conventional VQA-style metrics. As summarized in Table 1, the resulting resource covers 570K ERP panoramas and provides a combination of depth-aware signals, entity-level metadata, scalable annotation, and verified graph supervision not jointly available in prior panoramic resources. We further introduce a pano-aware MLLM model **PanoWorld** with Spherical Spatial Cross-Attention, enabling the model to align visual features with the underlying geometry of panoramic inputs. Extensive experiments show that our approach achieves strong performance on the proposed benchmark and generalizes effectively to existing 360° reasoning benchmark H*Bench [52] and VLN benchmark R2R-CE Val-Unseen [20], substantially outperforming existing methods.

In summary, our main contributions are as follows:

- We systematically formulate panoramic spatial reasoning in MLLMs as a capability-structured problem, and derive a taxonomy including spherical localization, relative direction reasoning, viewpoint transformation, 3D spatial relations, and global scene topology.
- Based on this formulation, we develop a scalable metadata-driven pipeline for large-scale panoramic data construction, together with a comprehensive benchmark that systematically evaluates the defined spatial reasoning capabilities.
- We propose a pano-aware MLLM model **PanoWorld** with spherical spatial cross-attention for geometry-consistent panoramic understanding.

Table 1: Comparison with representative panoramic resources. We report the number of underlying panoramic images rather than only QA counts. ✓: available; ✗: unavailable; △: partially available.

Resource	Number	View	Scene	Depth / 3D	Entity metadata	Scalable annotation	QA / Captions	Verified graph
PanoCity [15]	120K	Pano	Outdoor	✓	✗	△	✗	✗
Pano-AVQA [53]	5.4K	Video	Mixed	△	△	✗	✓	✗
Dense360 [63]	160K	Pano	Mixed	✗	✓	✓	✓	△
OmniVQA [57]	1.2K	Pano	Indoor	✗	△	△	✓	✗
ODI-Bench [49]	2K	Pano	Mixed	✗	△	✗	✓	✗
CFpano [56]	2.7K	Multi-Pano	Mixed	✗	△	✗	✓	✗
PanoVQA [11]	44.6K	Pano	Outdoor	✗	△	△	✓	✗
OSR-Bench [10]	4.1K	Pano	Indoor	△	△	✗	✓	✗
PanoEnv [26]	595	Multi-Persp.	Mixed	✓	△	✗	✓	✗
PanoWorld	**570K**	**Pano**	**Mixed**	✓	✓	✓	✓	✓

- Extensive experiments demonstrate that the proposed model has achieved competitive performance on the proposed benchmark and effective transfer to existing 360° reasoning and VLN benchmarks, with substantial gains over prior methods.

2 Related Work

Pano-Native Panoramic Designing. To bridge gaps [24] between panoramic and perspective understanding, recent studies have emphasized the need for panorama-specific design. Existing efforts mainly address this problem from two perspectives: data and models. On the data side, recent works construct task-specific datasets and benchmarks for perception [25, 14, 60], and MLLM understanding [63, 57, 49, 26] On the model side, prior studies introduce pano-aware designs, including distortion map [25, 9, 51, 8] and spherical positional modeling [15, 22, 33, 27, 12, 11]. However, most existing efforts remain centered on specific tasks or designs, rather than a unified capability-level formulation of panoramic MLLM understanding.

Spatial Reasoning in Multimodal Large Language Models. Recent surveys identify spatial reasoning as a systematic bottleneck for large multimodal models, spanning spatial relations, 3D scene understanding, embodied interaction, and geometry-aware representation learning [29, 61, 54]. Following this perspective, spatial reasoning has become an important axis of multimodal evaluation, covering 2D/3D relations, depth order, relative distance, egocentric memory, and embodied question answering [36, 48, 32, 34, 50, 17, 16, 31, 6]. Recent methods therefore introduce spatial supervision or geometry-aware representations, such as depth-aware region features, 3D position embeddings, position-aware video representations, and structured 3D scene tokens [4, 6, 64, 59, 45]. These works demonstrate the value of spatial representation learning, but they primarily rely on perspective images, egocentric videos, multi-view observations, or explicit 3D geometry.

Recently, Thinking in 360 [52] studies human-centric visual search in immersive 360° environments. This view-based formulation is natural for embodied visual search, yet it treats the panorama primarily as a source of discrete views and leaves panorama geometry implicit. In contrast, we ask whether the ERP panorama itself can serve as the native spatial representation of the surrounding space. This motivates our pano-native framework, which injects spherical geometry into ERP visual tokens and trains MLLMs to reason directly over continuous, observer-centered panoramic space.

3 Method

Our goal is to enable MLLMs to understand panoramas as continuous, observer-centered 360° spaces. We first introduce the geometric preliminaries and task settings in Sec. 3.1. We then present a capability taxonomy for pano-native understanding in Sec. 3.2. Based on the formulation, we describe our large-scale metadata construction pipeline in Sec. 3.3. Finally, we introduce our pano-aware MLLM model in Sec. 3.4.

3.1 Preliminary and Task Settings

Unlike perspective images defined on a planar image grid, panoramic images are commonly represented in equirectangular projection (ERP), where each pixel corresponds to a spherical direction

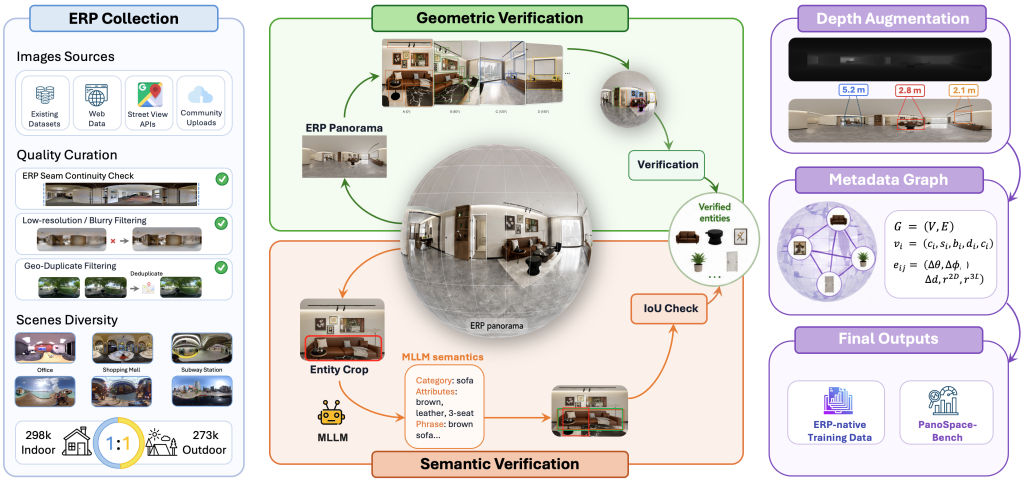


Figure 2: Verifiable metadata construction pipeline. We collect mixed-source ERP panoramas, perform perspective-view detection followed by ERP reprojection and cross-view geometric verification, and enrich verified entities with semantic metadata through MLLM annotation and description-guided referring re-detection. Depth cues are then associated with each entity to build a structured metadata graph, from which both training data and PanoSpace-Bench are derived.

parameterized by yaw and pitch. For an ERP pixel (u, v) with width W and height H, its yaw and pitch are

$$\lambda = 2\pi \left(\frac{u}{W} - \frac{1}{2} \right), \qquad \phi = \pi \left(\frac{1}{2} - \frac{v}{H} \right), \tag{1}$$

where $\lambda \in [-\pi, \pi)$ and $\phi \in [-\frac{\pi}{2}, \frac{\pi}{2}]$. The corresponding unit ray on the sphere is

$$\mathbf{r}(\lambda, \phi) = \begin{bmatrix} \cos\phi \sin\lambda \\ \sin\phi \\ \cos\phi \cos\lambda \end{bmatrix}, \tag{2}$$

which gives the viewing direction of that ERP location.

Given an ERP panorama $I \in \mathbb{R}^{H \times W \times 3}$ and a text query q, we study pano-native understanding for multimodal large language models by learning a multimodal function

$$y = f_\theta(I, q), \tag{3}$$

where y may denote an answer, a direction, a spatial relation, or a grounded target. Different from standard visual question answering on perspective images, this setting requires the model to reason over I as a continuous observer-centered spherical space, including seam continuity, viewpoint reorientation, and relations among entities distributed across the full panorama.

3.2 Capability Taxonomy for Pano-native Understanding

We decompose pano-native understanding into four capability families that together define the core requirements for reasoning over ERP panoramas. This taxonomy serves as the foundation for both supervision design and benchmark construction.

Semantic anchoring. The model must ground language to visual entities in ERP panoramas, covering object identity, attributes, scene contents, and global scene-topology semantics such as environment and layout structure. This forms the semantic basis for subsequent spatial reasoning.

Spherical grounding. The model must localize entities on the observer-centered viewing sphere, where directions are parameterized by yaw and pitch, rather than only on a planar image grid. This ranges from coarse directional localization to fine-grained BFOV-style angular grounding.

Reference-frame transformation. The model must reason about how spatial relations change under observer rotation or object-conditioned reorientation, including angular relations on the sphere and seam-aware wrap-around continuity.

Depth-aware 3D spatial reasoning. The model must connect spherical observations to surrounding 3D structure, including depth, relative distance, and viewer-centered relations such as left/right, front/behind, and above/below.

Together, these four families define pano-native understanding from *what* is present, to *where* it lies on the sphere, to *how* its relation changes under reference-frame transformation, and finally to *how* it is organized in 3D space around the observer. Table 11 summarizes the resulting task operators under each family, which instantiate this taxonomy as structured supervision. The next subsection describes how we construct the verified panorama metadata that supports them.

3.3 Large-scale Dataset Collection and Verifiable Metadata Construction

To support the capability taxonomy above, we require supervision that is both large-scale and verifiable. As shown in Figure 2, we construct a large ERP corpus and derive from it geometry-aware, semantic, and depth-aware metadata, which are finally unified as a structured metadata graph. Please refer to Appendix A for more details.

ERP collection from mixed sources. We build a large-scale ERP corpus from mixed sources, including existing panoramic datasets, web data, street-view APIs, and community-contributed uploads, as illustrated in Figure 2. The source composition and scene breakdown are summarized in Table 9. We then apply a quality-curation stage to remove invalid or low-quality samples, including ERP seam discontinuity checking, low-resolution and blur filtering, and geo-duplicate removal. Finally, we promote scene diversity by balancing indoor and outdoor panoramas and covering a broad range of environments, such as offices, shopping malls, subway stations, streets, public spaces, and natural scenes. The resulting corpus contains about 570K high-quality ERP panoramas with an approximately balanced indoor/outdoor ratio.

Geometry-aware detection metadata. Direct detection on ERP is unreliable as object shapes are distorted near high latitudes and may be split by the left-right seam. We therefore project each panorama into a set of overlapping perspective views and apply an off-the-shelf open-vocabulary detector to obtain candidate boxes. The detections are then reprojected to the ERP coordinate system and merged across views. As shown in Figure 2, we further apply **geometric verification**, including confidence thresholding, IoU-based duplicate suppression, and cross-view consistency checking, to remove unstable proposals caused by projection artifacts, seam splitting, or single-view detector failures. This process produces panorama-level entity candidates with reliable spherical locations and box extents.

Language-grounded semantic metadata. For each retained candidate, we select the most informative local crop or perspective view and prompt a multimodal language model to generate semantic annotations, including object category, attributes, descriptions, and a discriminative referring phrase. We then perform a crop-centered description–re-detection **semantic verification** step as shown in Figure 2, in which the generated phrase is fed to a referring/open-vocabulary detector to localize the same target again. Candidates whose re-detected boxes do not sufficiently overlap with the original proposals are discarded. This step improves semantic precision and filters out mismatches between language and detection.

Depth-aware spatial metadata. We further associate each verified entity with depth information. When aligned depth is available from the source data, we use it directly; otherwise, we estimate pseudo-depth with a panoramic depth model [25]. Depth values are aggregated over the ERP support region of each entity to estimate observer distance and derive depth-aware spatial cues.

Metadata graph construction. Combining semantics, angular location, box extent, and depth, we represent each panorama as a structured metadata graph

$$\mathcal{G} = (\mathcal{V}, \mathcal{E}), \tag{4}$$

where each node $v_i \in \mathcal{V}$ is a verified entity

$$v_i = (s_i, a_i, b_i, d_i, c_i)\,, \tag{5}$$

with semantics s_i, attributes a_i, angular footprint $b_i = (\theta_i, \phi_i, \Delta\theta_i, \Delta\phi_i)$, observer distance d_i, and local visual context c_i. Each edge $e_{ij} \in \mathcal{E}$ stores pairwise spherical and 3D relations:

$$e_{ij} = \left(\Delta\theta_{ij}, \Delta\phi_{ij}, \Delta d_{ij}, r_{ij}^{2D}, r_{ij}^{3D}\right). \tag{6}$$

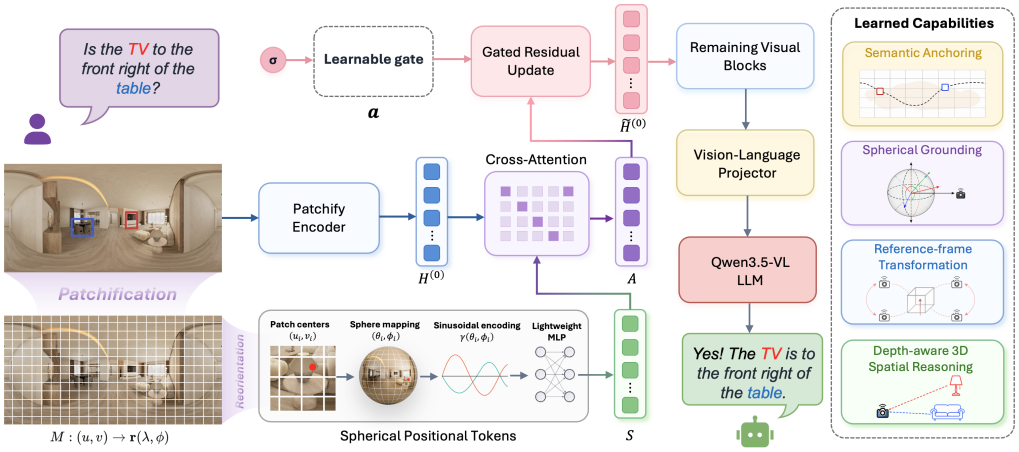


Figure 3: The architecture of PanoWorld. After patch embedding, visual tokens $H^{(0)}$ query spherical spatial tokens S derived from ERP patch centers, producing a geometry-aware signal that is fused through a gated residual update. The enhanced tokens are then fed into the remaining pretrained visual encoder, enabling pano-aware spatial reasoning while preserving the original backbone.

where $\Delta\theta_{ij}$ and $\Delta\phi_{ij}$ are spherical angular offsets, Δd_{ij} is the relative depth difference, and r_{ij}^{2D} and r_{ij}^{3D} denote discretized spherical and viewer-centered 3D relations, respectively. All downstream training tasks are instantiated from this graph, which serves as the structured interface between raw ERP data and capability-aligned supervision. We next describe the pano-aware model adaptation that learns from this supervision.

3.4 Pano-aware MLLM Adaptation

The model is illustrated in Figure 3. We adopt Qwen3.5-VL as the backbone and extend it with a pano-aware module that injects spherical geometry into the visual stream. Since the native visual encoder operates on a planar raster, it does not explicitly account for the spherical structure of ERP images, where the same pixel displacement may correspond to different angular changes at different latitudes and the left and right image borders are adjacent in the real scene. To address this mismatch, we introduce **Spherical Spatial Cross-Attention (SSCA)**, a pano-aware adapter inserted immediately after patch embedding.

Spherical spatial token construction. Given an ERP panorama, let $H^{(0)} \in \mathbb{R}^{N\times d}$ denote the patch embeddings produced by the visual patch projector, where N is the number of visual patches and d is the hidden dimension. For each patch i, we compute its center (u_i, v_i) in ERP image coordinates and map it to the corresponding spherical direction (λ_i, ϕ_i). We then encode this direction using a fixed sinusoidal spherical encoding $\gamma(\cdot)$ and project it into the visual hidden space:

$$s_i = \mathrm{MLP}(\gamma(\lambda_i, \phi_i)) \in \mathbb{R}^d. \tag{7}$$

Stacking all patch-level spherical tokens gives

$$S = [s_1, \ldots, s_N] \in \mathbb{R}^{N\times d}. \tag{8}$$

Unlike standard 2D positional indices, these tokens are explicitly tied to directions on the viewing sphere. They provide observer-centered geometric cues for discrete ERP patches; SSCA is not a continuous spherical operator or an exactly rotation-equivariant model.

Cross-attention fusion after patch embedding. SSCA injects spherical geometry by allowing visual tokens to retrieve information from the spherical tokens through cross-attention:

$$A = \mathrm{MHA}\big(Q = \mathrm{LN}(H^{(0)}), K = \mathrm{LN}(S), V = \mathrm{LN}(S)\big). \tag{9}$$

The resulting geometry-aware signal is fused back into the visual stream through a gated residual update:

$$\widetilde{H}^{(0)} = H^{(0)} + \boldsymbol{\alpha} \odot A, \tag{10}$$

Table 2: Quantitative comparison on PanoSpace-Bench. We report overall multiple-choice accuracy, BFOV mean IoU, and category-wise performance across panoramic localization, spherical relational reasoning, omnidirectional 3D spatial reasoning, and ERP representation properties.

Method	Overall	Localization		Spherical Relation				3D Spatial			ERP
		Abs. Dir.	BFOV	Rel. Dir.	Rot.	Reori.	Avg.	Dist.	Rel. 3D	Avg.	Seam
GPT-4o [18]	31.8	37.2	<u>17.7</u>	34.8	28.4	24.8	29.3	45.6	27.2	36.4	37.6
Mimo-v2.5 [47]	<u>37.2</u>	26.8	0.7	<u>42.0</u>	<u>42.8</u>	<u>42.0</u>	<u>42.2</u>	51.6	23.6	37.6	45.6
InternVL3.5-8B [43]	28.3	24.8	2.9	28.8	26.4	25.2	26.8	52.4	26.8	39.6	25.6
InternVL3.5-14B [43]	30.4	23.6	2.8	35.6	30.8	38.4	34.9	<u>56.8</u>	20.8	38.8	28.8
Qwen2.5-VL-3B [2]	30.1	20.4	2.7	30.8	29.6	34.8	31.7	43.6	<u>35.2</u>	39.4	<u>46.4</u>
Qwen2.5-VL-7B [2]	29.9	34.0	3.1	30.8	23.6	30.4	28.2	48.8	29.6	39.2	32.0
Qwen3-VL-8B [1]	29.6	47.6	2.2	28.4	21.2	27.2	25.6	45.6	23.6	34.6	34.4
Qwen3-VL-32B [1]	34.8	32.0	2.5	36.4	27.2	34.8	32.8	54.8	28.8	<u>41.8</u>	43.6
Ministral-8B [28]	25.5	22.4	2.1	22.0	20.4	20.0	20.8	47.6	29.6	38.6	28.4
Ministral-14B [28]	23.6	35.6	2.3	20.0	14.4	12.8	15.7	49.6	31.6	40.6	21.6
Qwen3.5-9B [35]	30.8	25.2	1.4	32.2	22.6	26.3	26.1	48.6	24.5	36.9	41.2
+ visual prompt	36.4	<u>55.2</u>	4.9	34.0	36.4	28.8	33.1	46.0	26.2	36.1	46.5
PanoWorld	**56.5**	**93.7**	**73.3**	**42.6**	**52.4**	**47.2**	**47.4**	**59.6**	**40.6**	**49.8**	**65.5**

where $\boldsymbol{\alpha} \in \mathbb{R}^d$ is a learnable gate initialized with a small value. The updated tokens $\widetilde{H}^{(0)}$ are then fed into the remaining visual blocks. In this way, spherical geometry is injected into the visual stream through adaptive interaction between visual content and observer-centered spatial tokens, while the pretrained backbone remains unchanged. The adapted model is trained on the pano-native instruction corpus derived from Sec. 3.2 and Sec. 3.3.

4 Experiments

4.1 Experimental Setup

Training setup. Unless otherwise specified, we adopt Qwen3.5 as the base model and fine-tune it on the pano-native instruction corpus constructed in Sec. 3.3. All model variants use the same training data mixture and optimization setting for fair comparison. We train on 8 A100 GPUs with AdamW, a learning rate of 1×10^{-6}, global batch size 2, gradient accumulation 4, and 1 training epoch.

Evaluation benchmarks and metrics. We evaluate on three benchmarks: the proposed PanoSpace-Bench, H*Bench [52], and R2R-CE Val-Unseen [20]. PanoSpace-Bench covers panoramic localization, spherical relational reasoning, omnidirectional 3D spatial reasoning, and ERP representation properties. Please refer to Sec. B in the Appendix for some details. We report category-wise accuracy for multiple-choice tasks and BFOV mIoU for fine-grained grounding. On H*Bench, we follow the official protocol and report overall accuracy together with the HOS and HPS subsets. For R2R-CE, we evaluate VLN transfer using standard navigation metrics, including NE, OSR, SR, and SPL. More details are provided in Appendix B.

4.2 Experimental Results

Quantitative comparison on PanoSpace-Bench. We first compare against proprietary MLLMs, open-source MLLMs, prompt-enhanced baselines, and our pano-native model on PanoSpace-Bench.

Table 2 shows that general-purpose MLLMs remain weak on pano-native spatial reasoning. Across both proprietary and open-source models, performance drops most clearly on BFOV grounding, reference-frame transformation, and viewer-centered 3D reasoning, even when basic object recognition is relatively strong. This gap indicates that the main challenge is not object semantics alone, but reasoning over the ERP panorama as a continuous observer-centered representation.

Prompt enhancement improves direct ERP inference, especially for coarse localization, confirming that part of the difficulty lies in the missing spherical coordinate convention. However, these gains remain limited on spherical relational reasoning and 3D spatial reasoning, where simply describing the ERP layout is insufficient. See Appendix A.4 for visual prompt.

Table 3: Transfer evaluation on H*Bench. **Left:** Perspective-view baselines and our zero-shot ERP model. **Right:** ERP-input baselines, prompt-only variants, and training-based adaptation.

(a) Perspective-based methods

Method	Overall	HOS	HPS
GPT-4o [18]	21.3	19.7	23.6
Gemini-2.5-Pro [7]	32.3	31.9	33.0
InternVL3.5-4B [43]	3.8	3.2	4.8
InternVL3.5-8B [43]	6.7	6.4	7.2
Qwen3-VL-8B [1]	19.1	23.6	12.2
Qwen3.5-9B [35]	18.9	21.8	14.5
HVS-3B* [52]	38.4	47.3	24.9
Ours zero-shot	**56.1**	**61.8**	**47.5**

(b) ERP panorama-based methods

Route / Setting	Overall	HOS	HPS	Yaw Acc	Pitch Acc
GPT-4o [18]	30.1	39.1	17.1	38.5	64.2
Gemini-2.5-Pro [7]	46.9	55.3	34.3	52.5	71.6
InternVL3.5-4B [43]	11.6	12.8	9.75	22.1	41.5
InternVL3.5-8B [43]	14.9	18.0	10.2	19.2	38.5
Qwen3-VL-8B [1]	13.1	15.0	10.3	16.8	39.2
Qwen3.5-9B [35]	19.4	26.2	9.3	23.5	46.5
+ Text prompt	38.5	43.3	31.2	40.0	49.5
+ Visual prompt	40.4	46.0	32.0	43.5	52.0
Qwen3.5 + H* SFT	17.8	11.1	27.7	25.3	42.5
Ours + H* SFT	**70.1**	**73.1**	**64.2**	**74.1**	**85.5**

Our pano-native model achieves the best overall performance, improving the Qwen3.5 baseline from 30.8 to 56.5. The gains are broad rather than category-specific: absolute direction rises from 25.2 to 93.7, BFOV mIoU from 1.41 to 73.3, spherical relation average from 26.1 to 47.4, 3D spatial average from 36.9 to 49.8, and seam reasoning from 41.2 to 65.5. These results support the central claim of the paper: robust panoramic understanding requires pano-native spatial learning rather than treating ERP as a wide 2D image or relying only on prompt-level coordinate descriptions.

Quantitative comparison on H*Bench. We evaluate transfer to H*Bench [52], which contains Humanoid Object Search (HOS) and Humanoid Path Search (HPS). As shown in Figure 4, prior methods are typically evaluated through perspective-view exploration, whereas our model directly takes ERP panoramas as input. This tests whether the learned representation transfers beyond our own benchmark and supports downstream 360° human-centric visual search.

Table 3(a) compares the conventional perspective-view protocol with direct ERP input. Our zero-shot model reaches 56.10 overall, substantially outperforming the strongest reported perspective-view baseline (38.40). This shows that the representation learned from pano-native supervision transfers to downstream panoramic search rather than overfitting to PanoSpace-Bench alone. This comparison also reflects different observation budgets. Appendix D.1 analyzes target visibility and input cost under these protocols.

Table 3(b) shows that direct ERP inference with a generic Qwen3.5 model reaches only 19.4 overall, and prompt engineering improves it to 38.5–40.4, suggesting that explicit coordinate instructions help but do not solve pano-native reasoning. Naive H*Bench fine-tuning with ERP input performs even worse (17.8 overall), indicating that target-task supervision alone is insufficient when the base model lacks panoramic spatial priors. In contrast, our model reaches 56.10 zero-shot and 70.00 after H* fine-tuning. Together, these results show that pano-native learning provides a transferable spatial initialization that is not recoverable from prompting or naive task-specific supervision.

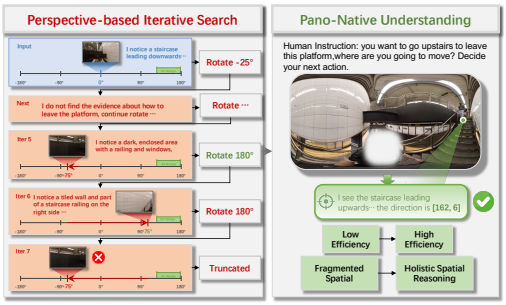


Figure 4: Case comparison on H*Bench. Perspective-view iterative search is inefficient and may fail due to fragmented local observations, whereas direct ERP input enables holistic reasoning and correct prediction in one step.

Quantitative comparison on VLN. We further evaluate transfer on R2R-CE Val-Unseen using only ERP panorama as input, which is different from methods that use panoramas to sample candidate perspective views for viewpoint selection. As shown in Table 4, PanoWorld achieves 54.3 SR and 52.1 SPL using only panorama input. Compared with methods that rely on waypoint predictors or use panoramas mainly for candidate-view selection, PanoWorld directly consumes the ERP panorama as a unified full-surround observation and improves over GridMM by 5.3 SR and 11.1 SPL. It also outperforms recent RGB-only VLN models under the same R2R/RxR training setting, surpassing StreamVLN by 4.1 SR and 5.0 SPL despite using only 80% of the training data. These results suggest that direct panoramic understanding provides an efficient and transferable paradigm for VLN, moving beyond fragmented local-view exploration toward unified full-surround spatial reasoning.

Table 4: R2R-CE Val-Unseen evaluation. † indicates waypoint-predictor methods; ∗ indicates training only on R2R/RxR for fair comparison. We follow this setting and use only 80% training data.

Method	Year	Paradigm	Observation				R2R-CE Val-Unseen			
			Pano.	Odom.	Depth	RGB	NE↓	OSR↑	SR↑	SPL↑
HPN+DN† [21]	2021	Waypoint	✓	✓	✓		6.31	40.0	36.0	34.0
Sim2Sim† [19]	2022	Waypoint	✓	✓	✓		6.07	52.0	43.0	36.0
GridMM† [44]	2023	Waypoint	✓	✓	✓		5.11	61.0	49.0	41.0
DreamWalker† [38]	2023	Waypoint	✓	✓	✓		5.53	59.0	49.0	44.0
Uni-NaVid∗ [55]	2024	RGB-only				✓	5.58	53.3	47.0	42.7
StreamVLN∗ [46]	2025	RGB-only				✓	5.73	56.4	50.2	47.1
Efficient-VLN∗ [58]	2026	RGB-only				✓	6.41	54.5	45.9	41.9
NaVIDA∗ [65]	2026	RGB-only				✓	5.72	57.4	47.7	41.5
PanoWorld-VLN∗	2026	**Direct ERP**	✓				**4.98**	**59.3**	**54.3**	**52.1**

Table 5: Ability-oriented training data ablation. The left block shows the four training ability groups derived from our pano-native capability taxonomy.

Training Data Ability				Localization		Spherical Relation				3D Spatial			ERP
Semantic	Spherical	Ref.-Frame	3D Spatial	Abs. Dir.	BFOV mIoU	Rel. Dir.	Cam. Rot.	Obj. Reori.	Avg.	Dist.	Rel. 3D	Avg.	Seam
✓				24.0	5.4	33.6	17.6	27.6	26.3	45.2	24.0	34.6	43.6
	✓			59.2	37.6	34.8	15.6	33.2	27.9	49.6	24.4	37.0	48.8
		✓		36.4	2.3	38.8	29.6	38.8	35.7	50.4	19.6	35.0	49.2
			✓	32.0	2.2	32.8	13.2	19.6	21.8	**63.2**	**36.8**	**50.0**	49.2
✓	✓			67.2	50.1	33.2	16.0	29.6	26.3	56.4	26.8	41.6	44.0
✓		✓		25.2	1.8	**41.6**	38.4	41.6	**40.5**	59.2	19.6	39.4	48.8
✓			✓	27.6	1.1	31.2	15.6	22.0	22.9	57.2	34.0	45.6	48.8
✓		✓	✓	36.4	5.6	37.2	29.2	40.8	35.7	53.6	35.6	44.6	47.6
✓	✓		✓	**71.6**	50.7	32.0	16.0	26.8	24.9	55.6	32.0	43.8	49.2
✓	✓	✓		60.0	43.0	40.0	29.2	44.8	38.0	57.2	24.8	41.0	51.8
✓	✓	✓	✓	68.8	**66.7**	34.8	**49.6**	**45.2**	43.2	52.0	34.8	43.4	**53.0**

4.3 Ablation Studies

We ablate the proposed framework from four perspectives: the composition of ability-oriented training data, the verification modules in metadata construction, the architecture used to inject spherical geometry, and the trainable scope used for ERP adaptation. Unless otherwise specified, all ablations are evaluated on the same split of PanoSpace-Bench.

Ability-oriented training data. Table 5 validates the proposed ability decomposition. Semantic-only training yields limited spatial performance, while adding spherical grounding sharply improves localization. Reference-frame transformation data is most beneficial for spherical relational reasoning, and depth-aware 3D supervision improves distance and relative 3D understanding. The best overall results are obtained by combining all ability families, indicating that panoramic understanding is compositional: semantic anchoring alone is insufficient, and strong performance requires jointly learning localization, transformation, and 3D spatial structure.

Metadata verification. Table 6 shows that both verification modules are important for reliable ERP supervision. Starting from the unverified baseline, detection verification improves overall accuracy from 38.8 to 46.4 by filtering geometrically unstable proposals, while semantic verification raises it to 48.0 by removing inconsistent language-region pairs. Combining both yields the best result of 55.1, with gains across localization, directional reasoning, 3D reasoning, and seam continuity. This confirms that data quality is a major factor in pano-native learning.

Architecture ablation. Table 7 compares residual fusion and cross-attention at different insertion positions. Patch-level cross-attention performs best overall, improving accuracy from 0.484 to 0.551 and yielding the strongest spherical relation average (0.460) and 3D spatial average (0.488). Residual fusion also helps in some settings, especially seam continuity, but is less consistent on relation-heavy categories. These results support the proposed SSCA design: geometry is most effective when injected early and through content-dependent interaction with visual tokens.

Trainable component ablation. Table 8 examines the trainable scope during pano-native adaptation. We compare different update strategies over the vision encoder, the vision-language interface, and the language model to identify where ERP-native spatial learning mainly takes place. The results

Table 6: Ablation of verification modules in the metadata construction pipeline. Detection verification filters geometrically unreliable proposals and grounding targets, while semantic verification removes semantically inconsistent question-answer pairs.

Pipeline Variant	Det. Verif.	Sem. Verif.	Overall	Localization	Direction	3D	Seam
baseline			38.8	58.3	32.6	35.5	48.2
w/ Detection Verification	✓		46.4	70.5	39.9	42.9	56.1
w/ Semantic Verification		✓	48.0	70.3	41.5	43.8	58.9
Full pipeline	✓	✓	**55.1**	**82.7**	**46.0**	**49.0**	**65.5**

Table 7: Architecture ablation of ERP-aware spherical geometry injection. We compare residual addition and cross-attention under three insertion positions: patch, merge, and output.

Method	Position	Overall	Localization		Spherical Relation				3D Spatial			ERP
			Abs. Dir.	BFOV mIoU	Rel. Dir.	Cam. Rot.	Obj. Reori.	Avg.	Dist.	Rel. 3D	Avg.	Seam
Qwen3.5	–	48.40	82.40	43.60	35.60	32.80	34.80	34.40	58.00	32.80	45.40	56.40
Residual	Merge	48.50	68.80	38.40	36.00	35.20	45.20	38.80	57.60	32.80	45.20	64.00
Residual	Output	50.50	79.20	43.10	37.60	36.40	34.80	36.30	**58.80**	36.80	47.80	**68.00**
Residual	Patch	51.50	82.40	48.10	37.60	40.00	36.00	37.90	57.20	34.00	45.60	64.00
Cross-Attn	Merge	50.80	81.60	43.80	34.40	42.80	**47.60**	41.60	53.20	34.40	43.80	61.60
Cross-Attn	Output	51.00	81.40	45.70	33.00	**51.60**	45.60	43.40	52.80	34.00	43.40	58.00
Cross-Attn	Patch	**55.10**	**92.80**	72.60	**40.00**	51.20	46.80	**46.00**	58.40	**39.20**	**48.80**	64.80

Table 8: Trainable scope ablation for pano-native spatial learning. We report fine-grained performance across benchmark tasks instead of aggregated group averages.

Train Scope	Trainable Components			Localization		Spherical Relation			3D Spatial		ERP
	Vision	VL Int.	LLM	Abs. Dir.	BFOV mIoU	Rel. Dir.	Cam. Rot.	Obj. Reori.	Dist.	Rel. 3D	Seam
LLM only			✓	84.80	62.28	34.00	36.40	38.00	57.60	37.20	55.60
VL Int. only		✓		25.60	2.19	33.60	22.80	26.40	41.40	24.80	37.40
VL Int. + LLM		✓	✓	83.60	64.25	35.20	47.20	44.40	56.80	35.20	56.40
Vision + VL Int.	✓	✓		34.20	6.04	32.40	24.80	31.30	47.60	29.20	46.40
Full pano-native FT	✓	✓	✓	**92.80**	**72.60**	**40.0**	**51.2**	**46.8**	**58.4**	**39.2**	**64.8**

indicate that panoramic reasoning depends not only on language-side adaptation, but also on updating the visual and cross-modal components that encode and align ERP geometry.

5 Conclusion

In this paper, we introduce a unified pano-native spatial learning framework for multimodal large language models. We formulate pano-native understanding as reasoning over ERP panoramas as continuous observer-centered spaces, and decompose it into four core capability families: semantic anchoring, spherical grounding, reference-frame transformation, and depth-aware 3D spatial reasoning. Based on this formulation, we build a large-scale metadata construction pipeline, derive capability-aligned instruction-tuning data, and propose an pano-aware model adaptation with spherical spatial cross-attention. We further construct PanoSpace-Bench to evaluate pano-native spatial reasoning beyond conventional VQA-style settings. Extensive experiments show that the proposed framework substantially improves panoramic reasoning on the proposed PanoSpace-Bench, H*Bench, and R2R-CE Val-Unseen benchmarks.

Acknowledgements and Disclosures

This work was supported by the National Natural Science Foundation of China (No. 62127803, 624B2124). The authors declare no competing interests.

References

[1] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, et al. Qwen3-vl technical report, 2025. URL https://arxiv.org/abs/2511.21631.

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, et al. Qwen2.5-vl technical report, 2025. URL https://arxiv.org/abs/2502.13923.

[3] Zidong Cao, Jinjing Zhu, Weiming Zhang, Hao Ai, Haotian Bai, Hengshuang Zhao, and Lin Wang. Panda: Towards panoramic depth anything with unlabeled panoramas and mobius spatial augmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 982–992, 2025.

[4] Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Danny Driess, Pete Florence, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. SpatialVLM: Endowing vision-language models with spatial reasoning capabilities. *arXiv preprint arXiv:2401.12168*, 2024.

[5] Hao Chen, Yuqi Hou, Chenyuan Qu, Irene Testini, Xiaohan Hong, and Jianbo Jiao. 360+x: A panoptic multi-modal scene understanding dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

[6] An-Chieh Cheng, Hongxu Yin, Yang Fu, Qiushan Guo, Ruihan Yang, Jan Kautz, Xiaolong Wang, and Sifei Liu. Spatialrgpt: Grounded spatial reasoning in vision-language models. In *Advances in Neural Information Processing Systems*, 2024.

[7] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities, 2025. URL https://arxiv.org/abs/2507.06261.

[8] Benjamin Coors, Alexandru Paul Condurache, and Andreas Geiger. Spherenet: Learning spherical representations for detection and classification in omnidirectional images. In *Proceedings of the European conference on computer vision (ECCV)*, pages 518–533, 2018.

[9] Xin Deng, Hao Wang, Mai Xu, Yichen Guo, Yuhang Song, and Li Yang. Lau-net: Latitude adaptive upscaling network for omnidirectional image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9189–9198, 2021.

[10] Zihao Dongfang, Xu Zheng, Ziqiao Weng, Yuanhuiyi Lyu, Danda Pani Paudel, Luc Van Gool, Kailun Yang, and Xuming Hu. Are multimodal large language models ready for omnidirectional spatial reasoning?, 2025. URL https://arxiv.org/abs/2505.11907.

[11] Weijia Fan, Ruiping Liu, Jiale Wei, Yufan Chen, Junwei Zheng, Zichao Zeng, Jiaming Zhang, Qiufu Li, Linlin Shen, and Rainer Stiefelhagen. More than the sum: Panorama-language models for adverse omni-scenes. *arXiv preprint arXiv:2603.09573*, 2026.

[12] Haoran Feng, Dizhe Zhang, Xiangtai Li, Bo Du, and Lu Qi. Dit360: High-fidelity panoramic image generation via hybrid training. *arXiv preprint arXiv:2510.11712*, 2025.

[13] Shenghao Fu, Yukun Su, Fengyun Rao, Jing Lyu, Xiaohua Xie, and Wei-Shi Zheng. Wedetect: Fast open-vocabulary object detection as retrieval. *arXiv preprint arXiv:2512.12309*, 2025.

[14] Xian Ge, Yuling Pan, Yuhang Zhang, Xiang Li, Weijun Zhang, Dizhe Zhang, Zhaoliang Wan, Xin Lin, Xiangkai Zhang, Juntao Liang, et al. Airsim360: A panoramic simulation platform within drone view. *arXiv preprint arXiv:2512.02009*, 2025.

[15] Yijing Guo, Mengjun Chao, Luo Wang, Tianyang Zhao, Haizhao Dai, Yingliang Zhang, Jingyi Yu, and Yujiao Shi. Panovggt: Feed-forward 3d reconstruction from panoramic imagery, 2026. URL https://arxiv.org/abs/2603.17571.

[16] 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. In *Advances in Neural Information Processing Systems*, 2023.

[17] Jiangyong Huang, Silong Yong, Xiaojian Ma, Xiongkun Linghu, Puhao Li, Yan Wang, Qing Li, Song-Chun Zhu, Baoxiong Jia, and Siyuan Huang. An embodied generalist agent in 3d world. In *International Conference on Machine Learning*, 2024.

[18] Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, Aaron Ostrow, Alethea Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[19] Jacob Krantz and Stefan Lee. Sim-2-sim transfer for vision-and-language navigation in continuous environments. In *European Conference on Computer Vision (ECCV)*, 2022.

[20] Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. Beyond the nav-graph: Vision-and-language navigation in continuous environments. In *European Conference on Computer Vision*, pages 104–120. Springer, 2020.

[21] Jacob Krantz, Aaron Gokaslan, Dhruv Batra, Stefan Lee, and Oleksandr Maksymets. Waypoint models for instruction-guided navigation in continuous environments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[22] Haodong Li, Wangguangdong Zheng, Jing He, Yuhao Liu, Xin Lin, Xin Yang, Ying-Cong Chen, and Chunchao Guo. Da^2: Depth anything in any direction. *arXiv preprint arXiv:2509.26618*, 2025.

[23] Linyuan Li, Yan Wu, Xi Li, Lingli Wang, Tong Rao, Jie Zhou, Cihui Pan, and Xinchen Hui. Realsee3d: A large-scale multi-view rgb-d dataset of indoor scenes (version 1.0), 2025. URL `https://doi.org/10.5281/zenodo.17826243`.

[24] Xin Lin, Xian Ge, Dizhe Zhang, Zhaoliang Wan, Xianshun Wang, Xiangtai Li, Wenjie Jiang, Bo Du, Dacheng Tao, Ming-Hsuan Yang, et al. One flight over the gap: A survey from perspective to panoramic vision. *arXiv preprint arXiv:2509.04444*, 2025.

[25] Xin Lin, Meixi Song, Dizhe Zhang, Wenxuan Lu, Haodong Li, Bo Du, Ming-Hsuan Yang, Truong Nguyen, and Lu Qi. Depth any panoramas: A foundation model for panoramic depth estimation. *arXiv preprint arXiv:2512.16913*, 2025.

[26] Zekai Lin and Xu Zheng. PanoEnv: Exploring 3d spatial intelligence in panoramic environments with reinforcement learning. *arXiv preprint arXiv:2602.21992*, 2026.

[27] Zhixin Ling, Zhen Xing, Xiangdong Zhou, Manliang Cao, and Guichun Zhou. Panoswin: A pano-style swin transformer for panorama understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[28] Alexander H. Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, et al. Ministral 3, 2026. URL `https://arxiv.org/abs/2601.08584`.

[29] Weichen Liu, Qiyao Xue, Haoming Wang, Xiangyu Yin, Boyuan Yang, and Wei Gao. Spatial reasoning in multimodal large language models: A survey of tasks, benchmarks and methods. *arXiv preprint arXiv:2511.15722*, 2025.

[30] Yuheng Liu, Xin Lin, Xinke Li, Baihan Yang, Chen Wang, Kalyan Sunkavalli, Yannick Hold-Geoffroy, Hao Tan, Kai Zhang, Xiaohui Xie, et al. Omniroam: World wandering via long-horizon panoramic video generation. *arXiv preprint arXiv:2603.30045*, 2026.

[31] Xiaojian Ma, Silong Yong, Zilong Zheng, Qing Li, Yitao Liang, Song-Chun Zhu, and Siyuan Huang. Sqa3d: Situated question answering in 3d scenes. In *International Conference on Learning Representations*, 2023.

[32] Arjun Majumdar, Anurag Ajay, Xiaohan Zhang, Pranav Putta, Sriram Yenamandra, Mikael Henaff, Sneha Silwal, Paul Mcvay, Oleksandr Maksymets, Sergio Arnaud, Karmesh Yadav, Qiyang Li, Ben Newman, Mohit Sharma, Vincent Berges, Shiqi Zhang, Pulkit Agrawal, Yonatan Bisk, Dhruv Batra, Mrinal Kalakrishnan, Franziska Meier, Chris Paxton, Alexander Sax, and Aravind Rajeswaran. Openeqa: Embodied question answering in the era of foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16488–16498, 2024.

[33] Zhijie Shen, Chunyu Lin, Kang Liao, Lang Nie, Zishuo Zheng, and Yao Zhao. Panoformer: Panorama transformer for indoor 360° depth estimation. In *European Conference on Computer Vision (ECCV)*, 2022.

[34] Ilias Stogiannidis, Steven McDonagh, and Sotirios A. Tsaftaris. Mind the gap: Benchmarking spatial reasoning in vision-language models. *arXiv preprint arXiv:2503.19707*, 2025.

[35] Qwen Team. Qwen3.5: Accelerating productivity with native multimodal agents, February 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

[36] 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 *Advances in Neural Information Processing Systems*, 2024.

[37] Changpeng Wang, Haozhe Wang, Xi Chen, Junhan Liu, Taofeng Xue, Chong Peng, Donglian Qi, Fangzhen Lin, and Yunfeng Yan. From illusion to intention: Visual rationale learning for vision-language reasoning, 2025. URL `https://arxiv.org/abs/2511.23031`.

[38] Hanqing Wang, Wenguan Wang, Tianmin Shu, Wei Liang, and Jianbing Shen. Dreamwalker: Mental planning for continuous vision-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[39] Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhu Chen. Vl-rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning. *arXiv preprint arXiv:2504.08837*, 2025.

[40] Haozhe Wang, Alex Su, , Weiming Ren, Fangzhen Lin, and Wenhu Chen. Pixel reasoner: Incentivizing pixel-space reasoning with curiosity-driven reinforcement learning. *arXiv preprint arXiv:2505.15966*, 2025.

[41] Haozhe Wang, Cong Wei, Weiming Ren, Jiaming Liu, Fangzhen Lin, and Wenhu Chen. Rationalrewards: Reasoning rewards scale visual generation both training and test time. *arXiv preprint arXiv:2604.11626*, 2026.

[42] Ning-Hsu Wang and Yu-Lun Liu. Depth anywhere: Enhancing 360 monocular depth estimation via perspective distillation and unlabeled data augmentation. *Advances in Neural Information Processing Systems*, 37:127739–127764, 2024.

[43] Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, Zhaokai Wang, et al. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency, 2025. URL `https://arxiv.org/abs/2508.18265`.

[44] Zihan Wang, Xiangyang Li, Jiahao Yang, Yeqi Liu, and Shuqiang Jiang. Gridmm: Grid memory map for vision-and-language navigation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[45] Zihan Wang, Seungjun Lee, and Gim Hee Lee. Dynam3d: Dynamic layered 3d tokens empower vlm for vision-and-language navigation. In *Advances in Neural Information Processing Systems*, 2025.

[46] Meng Wei, Chenyang Wan, Xiqian Yu, Tai Wang, Yuqiang Yang, Xiaohan Mao, Chenming Zhu, Wenzhe Cai, Hanqing Wang, Yilun Chen, Xihui Liu, and Jiangmiao Pang. Streamvln: Streaming vision-and-language navigation via slowfast context modeling. *arXiv preprint arXiv:2507.05240*, 2025.

[47] Xiaomi MiMo Team. Mimo-v2.5. `https://huggingface.co/collections/XiaomiMiMo/mimo-v25`, 2026. Accessed: 2026-05-06.

[48] Jihan Yang, Shusheng Yang, Anjali W. Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

[49] Liu Yang, Huiyu Duan, Ran Tao, Juntao Cheng, Sijing Wu, Yunhao Li, Jing Liu, Xiongkuo Min, and Guangtao Zhai. ODI-Bench: Can MLLMs understand immersive omnidirectional environments? *arXiv preprint arXiv:2510.11549*, 2025.

[50] 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 preprint arXiv:2511.04670*, 2025.

[51] Fanghua Yu, Xintao Wang, Mingdeng Cao, Gen Li, Ying Shan, and Chao Dong. Osrt: Omnidirectional image super-resolution with distortion-aware transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13283–13292, 2023.

[52] Heyang Yu, Yinan Han, Xiangyu Zhang, Baiqiao Yin, Bowen Chang, Xiangyu Han, Xinhao Liu, Jing Zhang, Marco Pavone, Chen Feng, Saining Xie, and Yiming Li. Thinking in 360°: Humanoid visual search in the wild. *arXiv preprint arXiv:2511.20351*, 2025.

[53] Heeseung Yun, Youngjae Yu, Wonsuk Yang, Kangil Lee, and Gunhee Kim. Pano-avqa: Grounded audio-visual question answering on 360° videos, 2021. URL `https://arxiv.org/abs/2110.05122`.

[54] Junsheng Zha et al. How to enable llm with 3d capacity? a survey of spatial reasoning in llm. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, 2025.

[55] Jiazhao Zhang, Kunyu Wang, Shaoan Wang, Minghan Li, Haoran Liu, Songlin Wei, Zhongyuan Wang, Zhizheng Zhang, and He Wang. Uni-navid: A video-based vision-language-action model for unifying embodied navigation tasks. *arXiv preprint arXiv:2412.06224*, 2024.

[56] Xinshen Zhang, Tongxi Fu, and Xu Zheng. Omnidirectional spatial modeling from correlated panoramas. *arXiv preprint arXiv:2509.02164*, 2025.

[57] Xinshen Zhang, Zhen Ye, and Xu Zheng. Towards omnidirectional reasoning with 360-r1: A dataset, benchmark, and GRPO-based method. *arXiv preprint arXiv:2505.14197*, 2025.

[58] Duo Zheng, Shijia Huang, Yanyang Li, and Liwei Wang. Efficient-vln: A training-efficient vision-language navigation model. *arXiv preprint arXiv:2512.10310*, 2025.

[59] Duo Zheng, Shijia Huang, and Liwei Wang. Video-3d llm: Learning position-aware video representation for 3d scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8995–9006, 2025.

[60] Jia Zheng, Junfei Zhang, Jing Li, Rui Tang, Shenghua Gao, and Zihan Zhou. Structured3d: A large photo-realistic dataset for structured 3d modeling. In *European Conference on Computer Vision*, pages 519–535. Springer, 2020.

[61] Xu Zheng, Zihao Dongfang, Lutao Jiang, Boyuan Zheng, Yulong Guo, Zhenquan Zhang, Giuliano Albanese, Runyi Yang, Mengjiao Ma, Zixin Zhang, Chenfei Liao, Dingcheng Zhen, Yuanhuiyi Lyu, Yuqian Fu, Bin Ren, Linfeng Zhang, Danda Pani Paudel, Nicu Sebe, Luc Van Gool, and Xuming Hu. Multimodal spatial reasoning in the large model era: A survey and benchmarks. *arXiv preprint arXiv:2510.25760*, 2025.

[62] Ding Zhong, Xu Zheng, Chenfei Liao, Yuanhuiyi Lyu, Jialei Chen, Shengyang Wu, Linfeng Zhang, and Xuming Hu. Omnisam: Omnidirectional segment anything model for uda in panoramic semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23892–23901, 2025.

[63] Yikang Zhou, Tao Zhang, Dizhe Zhang, Shunping Ji, Xiangtai Li, and Lu Qi. Dense360: Dense understanding from omnidirectional panoramas. *arXiv preprint arXiv:2506.14471*, 2025.

[64] Chenming Zhu, Tai Wang, Wenwei Zhang, Jiangmiao Pang, and Xihui Liu. Llava-3d: A simple yet effective pathway to empowering lmms with 3d capabilities. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4295–4305, 2025.

[65] Weiye Zhu, Zekai Zhang, Xiangchen Wang, Hewei Pan, Teng Wang, Tiantian Geng, Rongtao Xu, and Feng Zheng. Navida: Vision-language navigation with inverse dynamics augmentation. *arXiv preprint arXiv:2601.18188*, 2026.

A Dataset Details

A.1 ERP Corpus Composition

Table 9 summarizes the ERP image sources. Our ERP corpus contains 570,321 full-surround panoramas collected from mixed sources. The corpus is approximately balanced between indoor and outdoor scenes, with 297,476 indoor panoramas and 272,845 outdoor panoramas. This balance is important for pano-native spatial learning, since indoor scenes provide object-rich local layouts and depth relations, while outdoor scenes introduce larger-scale structures, long-range visibility, and diverse panoramic topology.

Our work uses large-scale panoramic imagery, which raises licensing, privacy, and misuse considerations. The ERP corpus combines existing panoramic datasets and public web sources, including Realsee3D [23] and 360+X [5]. We cite the external datasets and model components used in the paper and will release only assets whose redistribution is compatible with the corresponding source licenses, data-use agreements, and terms of service. Panoramic imagery may contain homes, bystanders, vehicles, or other sensitive visual details. Before release, we will apply privacy-oriented filtering to remove or mask personally identifying content where applicable, exclude unsafe or sensitive scenes, and provide a removal channel for reported problematic examples.

Table 9: Composition of the collected ERP panorama corpus.

Source	**#Panoramas**	**Ratio**	**Scene type**
Realsee3D-real [23]	24,025	4.2%	Indoor
Realsee3D-synthetic [23]	273,451	47.9%	Indoor
360+X [5]	15,290	2.7%	Outdoor
Outdoor web crawling	76,643	13.4%	Outdoor
Street-view web crawling	63,651	11.2%	Outdoor
API-based collection	117,261	20.6%	Outdoor
Indoor total	**297,476**	**52.2%**	Indoor
Outdoor total	**272,845**	**47.8%**	Outdoor
Total	**570,321**	**100.0%**	Mixed

A.2 Metadata Pipeline Details

We provide additional implementation details for the metadata construction pipeline in Sec. 3.3. For each ERP panorama, we render overlapping perspective views with a 120° FoV and a 60° yaw stride, resulting in approximately 60° overlap between adjacent views. We use WeDetect-Large [13] as the open-world detector, with a confidence threshold of 0.3 and a view-level NMS IoU threshold of 0.5. The detected boxes are reprojected to ERP coordinates and merged across overlapping views; two boxes are considered geometrically consistent if their ERP IoU exceeds 0.6. For semantic annotation, we use Qwen3-VL-32B [1] to generate object categories, attributes, descriptions, and discriminative referring phrases. We then use WeDetect-Ref-4B [13] for description-guided re-detection and retain an entity only if the IoU between the original proposal and the re-detected box exceeds 0.7. This provides a strong implementation of geometric and language verification before constructing the final metadata graph [41].

A.3 Instruction Data Distribution

Table 11 summarizes the ERP corpus composition, and Table 10 reports the final instruction distribution. From the verified metadata graphs, we first instantiate 7.65M candidate instruction samples and then sample a canonical training set of 2.998M examples. The sampling procedure targets a balanced mixture across semantic, angular, reference-frame, and depth-aware reasoning tasks, while limiting repeated samples from the same scene within each family. On average, each ERP panorama contributes 5.26 canonical instruction examples. Fig. 5 and Fig. 6 illustrate the frequency of entity categories and the distribution of instruction formats.

Table 10: Distribution of pano-native instruction data by ability family. We first instantiate a large candidate pool from metadata graphs, and then sample a canonical training set according to the target task mixture.

Ability family	**Candidate pool**		**Canonical set**	
	Examples	**Ratio**	**Examples**	**Ratio**
Semantic anchoring	2,120,452	27.7%	1,101,882	36.8%
Angular grounding	1,060,226	13.9%	333,091	11.1%
Spherical reference-frame transformation	2,097,356	27.4%	823,810	27.5%
Depth-aware 3D relation	2,352,181	30.8%	732,730	24.4%
ERP distortion/topology awareness	15,852	0.2%	6,003	0.2%
Total	**7,646,067**	**100.0%**	**2,997,516**	**100.0%**

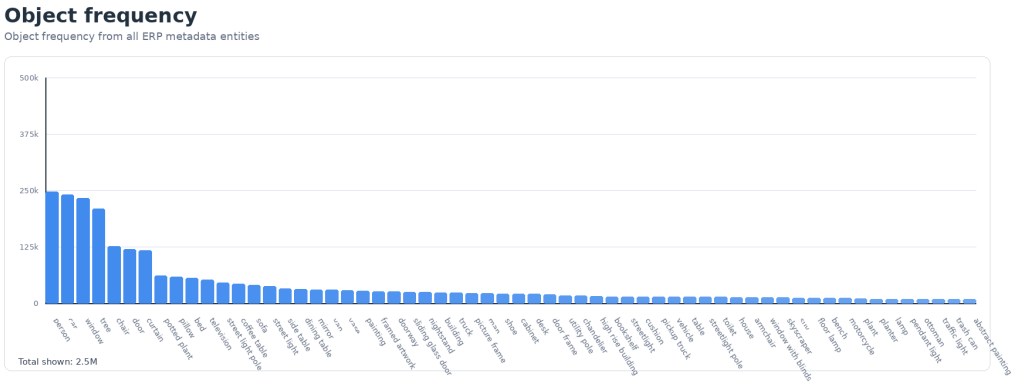


Figure 5: Object category distribution in the constructed metadata.

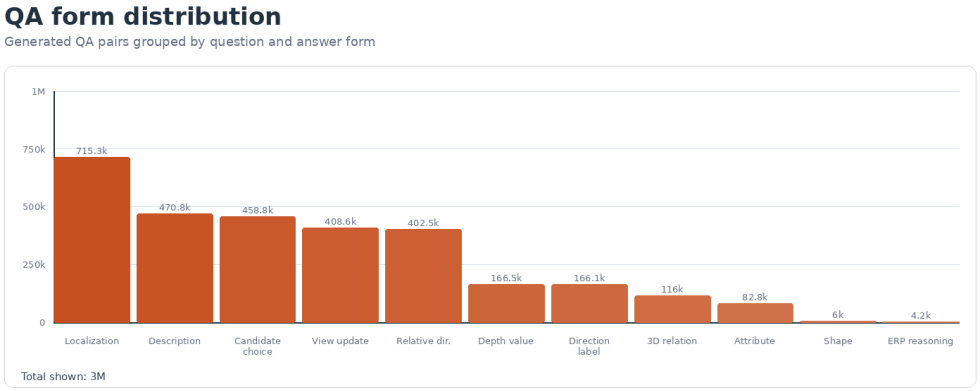


Figure 6: Instruction format distribution in the generated training data.

A.4 Prompt Template

This section details the prompt templates used in our evaluation pipeline. The core prompt explicitly defines the input as a full-surround ERP panorama, specifies the observer-centered reference frame, and standardizes the meanings of BFOV localization, relative direction, camera rotation, object-conditioned reorientation, physical distance, and relative 3D position. This unified formulation reduces ambiguity in spatial supervision and ensures that all task templates are grounded in the same panoramic coordinate system.

Unified pano-native system prompt. We use the following system prompt as the default instruction shared across ERP-native tasks.

Table 11: Detailed task templates instantiated by each pano-native supervision operator.

Operator	Task template	Question / supervision form	Purpose
Semantic Anchoring	Identification	Describe the highlighted entity?	Entity recognition on ERP inputs.
	Attribute QA	What visual attributes does the entity have?	Fine-grained visual semantics.
	Existence	Is there a target entity matching this description?	Entity-language grounding.
	Counting	How many entities of a given type are visible?	Global entity awareness.
	Scene captioning	Describe the region concisely or densely.	scene-level semantic description.
Angular Grounding	Absolute direction	Which direction sector contains the target entity?	Coarse spherical localization.
	Angular center prediction	Predict the target center direction (θ, ϕ).	Fine-grained spherical localization.
	Angular footprint prediction	Predict the target angular region $(\theta, \phi, \Delta\theta, \Delta\phi)$.	BFOV-style spatial grounding.
	Referring grounding	Localize the entity described by the query.	Language-to-region alignment.
Spherical Reference-frame Transformation	Relative direction	Where is entity B relative to entity A on the sphere?	Pairwise spherical relation.
	Camera rotation transform	After turning left/right by a given angle, where is the target?	Observer heading update.
	Object-conditioned reorientation	If facing entity A, where is entity B in the new frame of reference?	Object-centered reference frame.
Depth-aware 3D Relation	Observer distance	Which entity is closer to the observer?	Depth comparison.
	Distance ordering	Rank entities by distance from the observer.	Global depth structure.
	3D relative position	What is the relation between target A and target B?	Viewer-centered 3D relation.
	Compound 3D relation	Which option satisfies a combined relation such as front-left-above?	Multi-axis spatial reasoning.

Unified Pano-Native System Prompt

You are a multimodal assistant specialized in ERP (equirectangular projection) panoramic image understanding.
The input image is an ERP panorama representing the full 360-degree surrounding scene captured from a single fixed viewpoint. It should be interpreted as a continuous panoramic observation centered at the current observer, rather than as a standard perspective image.
All directional and spatial judgments are defined in an observer-centered reference frame anchored at the current observer. The image center corresponds to the current front direction.
BFOV is represented as $[\mathrm{yaw}, \mathrm{pitch}, x_{\mathrm{fov}}, y_{\mathrm{fov}}]$ in degrees. In this representation, yaw and pitch denote the center direction of the target object, while x_{fov} and y_{fov} denote the angular width and angular height covering the target object. Positive yaw corresponds to the observer's right, and negative yaw corresponds to the observer's left. Positive pitch corresponds to the upward direction, and negative pitch corresponds to the downward direction. The valid range of yaw is $[-180^\circ, 180^\circ)$, the valid range of pitch is $[-90^\circ, 90^\circ)$, and the valid ranges of x_{fov} and y_{fov} are $(0^\circ, 180^\circ]$.
Relative direction is defined by comparing the center direction of the target object with that of the reference object while keeping the current observer orientation fixed.
Camera rotation is defined as an in-place change of observer orientation without any change in observer position. Under this operation, the current front direction is updated according to the specified turn angle and turn direction, and the target object is then judged in the rotated observer frame.
Object-conditioned reorientation is defined as an in-place reorientation in which the center direction of the specified facing object becomes the new front direction. The target object is then judged in the reoriented observer frame.
Physical distance is defined in scene 3D space relative to the current observer, namely the current camera position.
Relative 3D position is defined as the positional relation of one object to another in the current observer-centered 3D frame.
Return only the requested answer in the required format unless explicitly instructed otherwise.

Additional ERP guidance for prompt-based inference. For the prompt-only ERP baselines, we further provide explicit guidance to help the model interpret ERP panoramas. We consider two forms of pano-specific guidance: a textual reference appendix, which verbally explains the ERP coordinate layout, and a visual guidance appendix, which combines an overlaid coordinate grid with accompanying text instructions. These auxiliary prompts are designed to teach the model how to read ERP panoramas, rather than to provide task-specific reasoning rules.

Text-only ERP reference. The following appendix provides a concise natural-language explanation of the ERP reference system.

ERP Reference System Appendix

Reference System (Equirectangular Projection):

- Image center: yaw 0°, corresponding to the front direction.
- Left and right image boundaries: yaw $\pm 180^\circ$, corresponding to the back direction.
- Yaw 90°: one quarter of the image width to the right of the center.
- Yaw -90°: one quarter of the image width to the left of the center.
- Vertical axis: pitch 0° is the horizon, pitch 90° is the top (zenith), and pitch -90° is the bottom (nadir).

Visual ERP guidance. We also construct a visual prompt by overlaying a coordinate grid on the ERP panorama and appending a short textual explanation of the grid semantics. This visual guidance makes the yaw–pitch structure directly observable in the image. **Grid rendering for visual prompting.** To generate the visual prompt, we render yaw grid lines every 30° and pitch grid lines every 15°, and mark the image center with a yellow crosshair corresponding to $(0^\circ, 0^\circ)$. The yaw lines are drawn in green and the pitch lines in blue, each with numerical labels placed along the image borders. This produces a visually interpretable ERP coordinate frame that can be directly consumed by the model.

ERP Visual Guidance Appendix

Visual Guidance System:

- The image is overlaid with a coordinate grid and numerical labels.
- Green vertical lines represent yaw angles. Numerical labels (e.g., $-180, -150, \ldots, 0, \ldots, 180$) are shown at the top and bottom.
- Blue horizontal lines represent pitch angles. Numerical labels (e.g., $-90, -75, \ldots, 0, \ldots, 90$) are shown at the left and right ends.
- A yellow crosshair marks the front direction at $(0^\circ, 0^\circ)$.

Task: Use the visual grid lines and numerical labels as a ruler to estimate the target center direction $[\mathrm{yaw}, \mathrm{pitch}]$. Interpolate between lines if the target does not lie exactly on a grid intersection.

B Benchmark Setting

B.1 PanoSpace-Bench

We introduce **PanoSpace-Bench**, a diagnostic benchmark for evaluating whether MLLMs understand equirectangular panoramas as a continuous, observer-centered representation of omnidirectional 3D spaces. Unlike existing panoramic benchmarks that typically focus on individual tasks such as VQA, captioning, grounding, or navigation, PanoSpace-Bench is designed to probe the capability structure behind first-person 360° spatial understanding, where directions, reference frames, and 3D relations are all defined around the observer.

Data separation. To avoid data leakage, PanoSpace-Bench is constructed from image sources that are completely separate from those used for the ERP-native instruction-tuning corpus. Specifically, benchmark panoramas are collected from different Internet sources, deduplicated against the training corpus, and manually verified for image quality and valid ERP layout. In addition to image-level separation, the benchmark questions are also separated from the training questions. Although some benchmark categories correspond to the same high-level abilities as the training tasks, we design distinct question formats and evaluation protocols for PanoSpace-Bench. Therefore, the benchmark evaluates generalization to unseen panorama sources and task formulations, rather than memorization of training images or generated QA templates.

As shown in Table 12, PanoSpace-Bench contains four ability families. Panoramic localization evaluates whether a model can ground targets in yaw–pitch space, including coarse absolute-direction classification and fine-grained BFOV localization. Spherical relational reasoning evaluates object-object angular relations and egocentric reference-frame transformations, including relative direction, camera rotation, and object-conditioned reorientation. Omnidirectional 3D spatial reasoning evaluates distance comparison and relative 3D position reasoning in the surrounding scene. ERP representation properties focuses on projection-specific topology, instantiated by seam-continuity questions that test whether the model treats the left and right ERP borders as adjacent on the viewing sphere.

Two design choices distinguish PanoSpace-Bench from prior omnidirectional evaluations. First, the benchmark emphasizes ERP-specific spatial modes rather than generic QA over panoramic content. Second, it separates semantic recognition from spatial correctness: a model may correctly identify the objects in a question but still fail if it does not understand their positions and relations in the observer-centered spherical frame. We report category-wise multiple-choice accuracy for closed-form tasks and BFOV mean IoU for fine-grained angular localization.

PanoSpace-Bench contains four ability families and eight task categories, with 250 questions for each category and 2,000 questions in total. Except for BFOV localization, all tasks are formulated as multiple-choice questions and evaluated by exact choice accuracy. BFOV localization requires the model to predict an angular bounding field-of-view in the format $[\mathrm{yaw}, \mathrm{pitch}, x_{\mathrm{fov}}, y_{\mathrm{fov}}]$, and is evaluated by angular IoU between the predicted and ground-truth BFOV regions.

Multiple-choice evaluation. For each multiple-choice question, the model is required to output one option from the candidate set. We parse the predicted response into a choice label $\hat{y}_i$ and compare it with the ground-truth label y_i. Invalid or unparsable responses are counted as incorrect. For a task category with N examples, the accuracy is computed as

$$\mathrm{Acc} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}(\hat{y}_i = y_i), \tag{11}$$

where $\mathbb{I}(\cdot)$ is the indicator function.

BFOV localization evaluation. For BFOV localization, the model predicts an angular region

$$\hat{b}_i = [\hat{\theta}_i, \hat{\phi}_i, \hat{w}_i, \hat{h}_i], \tag{12}$$

where $(\hat{\theta}_i, \hat{\phi}_i)$ is the predicted yaw-pitch center and $(\hat{w}_i, \hat{h}_i)$ are the predicted horizontal and vertical angular extents. The ground-truth BFOV is denoted as

$$b_i = [\theta_i, \phi_i, w_i, h_i]. \tag{13}$$

Each BFOV is converted to an angular rectangle on the ERP sphere:

$$R(b_i) = \left[\theta_i - \frac{w_i}{2}, \theta_i + \frac{w_i}{2}\right] \times \left[\phi_i - \frac{h_i}{2}, \phi_i + \frac{h_i}{2}\right]. \tag{14}$$

We then compute angular IoU as

$$\mathrm{IoU}(\hat{b}_i, b_i) = \frac{|R(\hat{b}_i) \cap R(b_i)|}{|R(\hat{b}_i) \cup R(b_i)|}, \tag{15}$$

where $|\cdot|$ denotes the angular area on the yaw-pitch domain. The final BFOV localization score is the mean IoU over all BFOV examples:

$$\mathrm{mIoU} = \frac{1}{N} \sum_{i=1}^{N} \mathrm{IoU}(\hat{b}_i, b_i). \tag{16}$$

When computing yaw overlap, we account for the circular wrap-around of ERP panoramas at the left-right boundary. Predictions with invalid formats or out-of-range angular values are treated as invalid and assigned zero IoU.

Table 12: Taxonomy of PanoSpace-Bench. The benchmark is organized into four pano-centered ability families and eight diagnostic task categories.

Family	Category	Representative question form	Metric
Panoramic localization	Absolute direction	Where is the target object located relative to the observer: left-back, left, or right ...?	MC acc.
Panoramic localization	BFOV localization	What BFOV $[\mathrm{yaw}, \mathrm{pitch}, x_{\mathrm{fov}}, y_{\mathrm{fov}}]$ localizes the target object in the panorama?	mIoU
Spherical relation	Relative direction	Where is object A relative to object B on the viewing sphere?	MC acc.
Spherical relation	Camera rotation	After the observer turns by a specified yaw angle, where would the target object appear?	MC acc.
Spherical relation	Object reorientation	If the observer first faces the reference object, where is the target object?	MC acc.
3D spatial reasoning	Observer distance	Which listed object is physically closest to the observer in 3D space?	MC acc.
3D spatial reasoning	Relative 3D position	Which option best describes the target object's 3D position relative to the observer, combining direction and depth?	MC acc.
ERP property	Seam continuity	For target A near the right boundary of the ERP panorama, which listed object is nearest to it in the full 360 scene?	MC acc.

B.2 Human-centric visual search.

H*Bench [52] evaluates human-centric visual search in 360° panoramas, including Humanoid Object Search (HOS) and Humanoid Path Search (HPS). The original setting uses an interactive perspective-view protocol, where the model observes a local FoV, iteratively performs rotation actions, and finally submits a target direction. This protocol is closely related to recent "thinking with images" studies, which encourage vision-language models to actively ground reasoning in visual evidence [40, 37]. In our ERP-input setting, the model directly receives the full ERP panorama and predicts the target direction in one step without decomposing the scene into local views. Following the official evaluation protocol of Thinking in 360, the model output is parsed as a yaw-pitch direction $(\hat{\theta}, \hat{\phi})$, and a prediction is considered successful if the submitted direction falls within the annotated target region or the task-specific angular tolerance around the ground-truth direction. We report the overall success rate as well as separate success rates for HOS and HPS:

$$\mathrm{Success} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\Big[\mathrm{hit}((\hat{\theta}_i, \hat{\phi}_i), \mathcal{T}_i)\Big], \tag{17}$$

where $\mathcal{T}_i$ denotes the ground-truth target region or valid angular tolerance, for example, i, and $\mathrm{hit}(\cdot)$ follows the benchmark evaluator. For setting the tolerance interval, we follow the parameters specified in the original paper. For perspective-view baselines, we additionally report the average number of interaction steps and model calls required before submission, following the original interactive setting.

B.3 R2R-CE vision-and-language navigation.

We further evaluate transfer to embodied navigation on R2R-CE Val-Unseen. Unlike conventional VLN methods that often use panoramas to construct candidate perspective views or rely on additional observations such as odometry, depth, or single-view RGB streams, our model directly takes the ERP panorama as the visual observation and predicts the navigation direction from the full-surround input. We follow the standard R2R-CE evaluation protocol and report navigation error (NE), oracle success rate (OSR), success rate (SR) and success weighted by path length (SPL). For an episode i, NE is the final geodesic distance d_i between the agent and the goal. SR is computed with the standard success

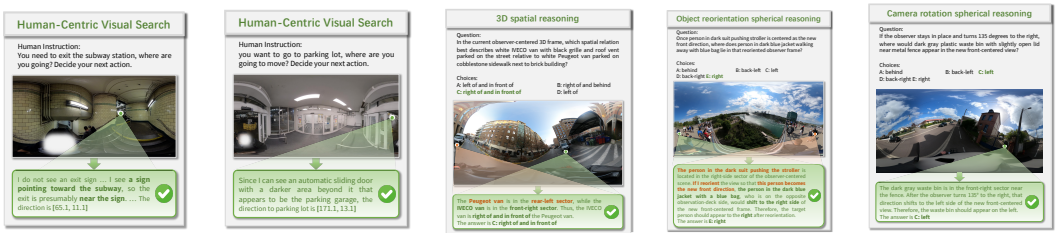
Table 13: Transfer evaluation on H*Bench. **Left:** Representative results under the original perspective-view setting and our PanoWorld zero-shot setting. **Right:** Route-level comparison of ERP baselines, prompt-only ERP formulation, and training-based ERP adaptation.

(a) Perspective-based methods

Method	Overall	HOS	HPS
GPT-4o	21.3	19.7	23.6
Gemini-2.5-Pro	32.3	31.9	33.0
Kimi-VL-A3B	4.6	4.92	4.32
InternVL3.5-4B	3.8	3.2	4.8
InternVL3.5-8B	6.7	6.4	7.2
Qwen2.5-VL-3B	11.4	14.8	6.4
Qwen2.5-VL-7B	9.3	11.3	6.3
Qwen3-VL-4B	17.5	19.5	14.4
Qwen3-VL-8B	19.1	23.6	12.2
Qwen3.5-9B	18.9	21.8	14.5
Gemma-3-12B	11.9	10.2	14.5
HVS-3B*	38.4	47.3	24.9
Ours + zero-shot	**56.1**	**61.8**	**47.5**

(b) ERP panorama-based methods

Route / Setting	Overall	HOS	HPS	Yaw Acc	Pitch Acc
GPT-4o	30.1	39.1	17.1	38.5	64.2
Gemini-2.5-Pro	46.9	55.3	34.3	52.5	71.6
InternVL3.5-4B	11.6	12.8	9.75	22.1	41.5
InternVL3.5-8B	14.9	18.0	10.25	19.2	38.5
Qwen2.5-VL-7B	9.1	9.6	8.5	11.9	32.3
Qwen3-VL-4B	12.8	14.3	10.5	15.6	40.0
Qwen3-VL-8B	13.1	15.0	10.3	16.8	39.2
Gemma-3-12B	9.6	10.9	7.5	14.0	40.8
Qwen3.5-9B	19.4	26.2	9.3	23.5	46.5
+ Text prompt	38.5	43.3	31.2	40.0	49.5
+ Visual prompt	40.4	46.0	32.0	43.5	52.0
Qwen3.5 + H* SFT	17.8	11.1	27.7	25.3	42.5
Ours + H* SFT	**70.1**	**73.1**	**64.2**	**74.1**	**85.5**



(a) Visual search (b) Visual search (c) 3D relation (d) Reorientation (e) Camera rotation

Figure 7: Case studies of pano-native spatial reasoning. The first two examples show downstream human-centric visual search, while the remaining examples show representative PanoSpace-Bench tasks covering 3D relation, object-conditioned reorientation, and camera-rotation reasoning.

threshold $\tau = 3$m:

$$\mathrm{SR} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(d_i \le \tau). \tag{18}$$

OSR uses the closest distance to the goal along the trajectory instead of the final distance:

$$\mathrm{OSR} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\Big(\min_{t} d_{i,t} \le \tau\Big). \tag{19}$$

SPL additionally penalizes unnecessarily long paths:

$$\mathrm{SPL} = \frac{1}{N}\sum_{i=1}^{N} S_i \frac{\ell_i}{\max(p_i, \ell_i)}, \tag{20}$$

where S_i is the success indicator, ℓ_i is the shortest-path length, and p_i is the executed path length. For fair comparison with recent RGB/video-based methods, our VLN fine-tuning uses only the R2R and RxR training sets, and PanoWorld uses only 80% of the training data.

C Case Study

We present qualitative examples for both downstream transfer and diagnostic evaluation. Figure 7 (a–b) shows human-centric visual search cases, where the model directly reasons over a full ERP panorama to infer the target movement direction. These examples illustrate that pano-native spatial learning supports practical 360° search without decomposing the scene into local perspective views. Moreover, as shown in Table 13, we present more comprehensive data results on H*.

Table 14: Efficiency comparison between perspective-view rotation and direct ERP inference on H* tasks. Direct ERP inference is more efficient in terms of interaction steps, sequential decision cost, and global spatial coverage.

Method	Input	Res.	Steps ↓	Calls ↓	Eff. Input Tokens	Rel. Cost
Qwen3-VL-4B	Persp. rotation	720^2	6.27	6.27	29.6k	1.80×
Qwen3-VL-8B	Persp. rotation	720^2	6.34	6.34	29.9k	1.81×
H*-trained Qwen2.5-VL	Persp. rotation	720^2	3.70	3.70	19.2k	1.16×
H*-trained Qwen3-VL	Persp. rotation	720^2	3.58	3.58	18.7k	1.13×
Ours	Direct ERP	1600×800	1.00	1.00	16.5k	1.00×

Figure 7 (c–e) shows representative PanoSpace-Bench cases, including 3D spatial relations, object-conditioned reorientation, and camera-rotation reasoning. These examples further demonstrate that the learned representation supports controlled evaluation of observer-centered spherical reasoning.

D Efficiency Study

We compare the inference efficiency of direct ERP reasoning with the perspective-view rotation paradigm on H* tasks. Rotation-based methods observe only one local FoV at each step and therefore require multiple sequential model calls before submitting the final direction. In contrast, our model directly consumes the full ERP panorama and predicts the answer in a single forward pass.

As shown in Table 14, perspective-view rotation requires 3.58–6.34 interaction steps on average, leading to 18.7K–29.9K effective input tokens. This corresponds to 1.13–1.81× the cost of direct ERP inference. Our method requires only one step and one model call, with 16.5K effective input tokens, while maintaining full 360° spatial coverage. This demonstrates that pano-native spatial learning replaces iterative local-view search with a unified and efficient full-surround inference paradigm.

D.1 Perspective visibility and resolution

To examine when a local perspective view lacks the evidence needed for H*Bench, we stratify the official test episodes over the four initial orientations by the fraction d of the target overlapping the initial field of view. The target is mostly visible ($d > 0.5$) in 25.7% of episodes, partially visible ($0 < d < 0.5$) in 7.8%, and entirely outside the initial field of view ($d = 0$) in 66.6%. The corresponding perspective-agent accuracies are 60.5%, 24.4%, and 44.8%. Thus, in 74.4% of episodes the initial local view contains incomplete or no target evidence; these agents can still succeed by rotating and integrating later views. The H*-fine-tuned PanoWorld checkpoint scores 73.1% on HOS using a complete panorama. This analysis explains the coverage advantage but does not isolate representation learning from the different observation protocols.

The full-resolution ERP observation is 1600×800 and uses 16.5K effective input tokens in this comparison. At 1024×512, the input uses 9.8K tokens and PanoWorld's zero-shot H*Bench overall accuracy changes from 56.1% to 52.1%. Both are above the strongest reported perspective-view baseline (38.4%), whose multiple observations use 18.7K–29.9K effective input tokens in total (Table 14). An ERP image can be resized to the same pixel count as a perspective image, but preserving comparable angular detail across the complete sphere generally requires more pixels per observation. The token comparison therefore concerns the complete interactive episode, not equal-resolution single images; it also excludes system-dependent wall-clock latency and image acquisition costs.

E Limitations

While PanoWorld demonstrates strong pano-native spatial reasoning, several limitations remain. First, our metadata construction pipeline relies on automatic open-world detection, MLLM-based semantic annotation, referring re-detection, and panoramic depth estimation. Although we introduce two-level verification to improve reliability, errors from these components may still propagate to the final metadata graph. Second, PanoSpace-Bench is designed as a diagnostic benchmark for

observer-centered ERP spatial reasoning, and therefore does not cover all possible panoramic tasks, such as long-horizon embodied interaction, dynamic scenes, or multi-agent navigation.

These limitations suggest several directions for future work. On the data side, more reliable panoramic perception modules and stronger cross-modal verification could further improve metadata quality. On the evaluation side, extending pano-native benchmarks from static ERP reasoning to interactive navigation, temporal panoramic videos, and dynamic 3D environments would provide a broader testbed for full-surround spatial intelligence. We hope PanoWorld and PanoSpace-Bench provide a foundation for these future studies.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the scope and contributions of the paper, including pano-native spatial learning, the metadata construction pipeline, PanoSpace-Bench, the ERP-aware model adaptation, and the evaluated transfer settings. The claims are supported by experiments on PanoSpace-Bench, H*Bench, and R2R-CE.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We include a limitations discussion covering the automatic metadata generation pipeline and the diagnostic scope of PanoSpace-Bench.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper does not present theoretical results, theorems, or formal proofs.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper and supplemental material provide the model architecture, data construction pipeline, prompt templates, training setup, benchmark protocols, evaluation metrics, and implementation details needed to reproduce the main experimental results. We will release the code, dataset, and model soon.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: At submission time, we provide detailed implementation and evaluation instructions in the supplemental material. We plan to release the code, benchmark, and processed metadata after acceptance, subject to the licenses and terms of the underlying data sources.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We specify the training backbone, data sources, instruction data distribution, metadata construction settings, optimizer configuration, benchmark splits, evaluation protocols, and metrics in the main paper and supplemental material.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We do not report error bars for the main model training results because large-scale MLLM fine-tuning is computationally expensive and the evaluations are conducted on fixed benchmark splits. We report deterministic evaluation metrics on the full test sets and provide ablations and transfer evaluations to support the conclusions.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide the hardware and software setup used for the experiments, including an on-premises server with an Intel Xeon Platinum 8470 CPU, 96 CPU cores, 8 NVIDIA A100 PCIe GPUs with 80GB memory each, 1007GB system RAM, NVMe SSD storage, and PyTorch with FP16 mixed precision. Together with the training and evaluation details in the supplementary material, this provides sufficient compute information for reproducing the main results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conforms to the NeurIPS Code of Ethics. We use publicly available or licensed data sources, cite existing assets, apply data filtering and privacy-oriented processing where applicable, and discuss potential societal impacts and safeguards.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses positive impacts such as improved panoramic perception for navigation, robotics, and assistive systems, as well as potential risks including privacy concerns and surveillance misuse.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: We apply data filtering to remove unsafe or low-quality panoramas, avoid releasing personally identifying information where applicable, follow source licenses, and plan to release derived metadata and benchmark assets with research-use terms and documentation. For web-sourced data, we provide safeguards such as privacy filtering and removal procedures.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite all existing datasets, benchmarks, models, and codebases used in the paper, including their versions where applicable. We describe the data sources and will release derived assets under terms compatible with the original licenses and usage restrictions.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The paper introduces new assets, including PanoSpace-Bench, the ERP-native instruction data, metadata construction pipeline, and PanoWorld model. We document their construction, formats, intended use, limitations, and evaluation protocols in the paper and supplemental material.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments or human-subject studies. The data and annotations are constructed using existing datasets, web/API sources, and automatic metadata generation pipelines.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The work does not involve human-subject experiments or interventions, and therefore IRB approval is not applicable. Privacy considerations for panoramic imagery are discussed under safeguards and broader impacts.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: LLMs/VLMs are used as core components of the research: Qwen3-VL-32B is used for semantic metadata generation, Qwen3.5-VL is used as the backbone for model adaptation, and LLM-based prompting protocols are described in the paper. Their usage is documented in the method and supplemental material.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.