From Indoor to Open World: Revealing the Spatial Reasoning Gap in MLLMs

Supplementary Material

Appendix Outline

In the supplementary material, we provide:
- Visual illustration of linguistic priors;
- Technical details of data collection and benchmark construction;
- Error analysis and pipeline validation;
- Evaluation setups and experiment details;
- More evaluation results.
- Privacy statement.

A. Illustration of Linguistic Priors

To clearly demonstrate that models rely on **linguistic priors** when answering spatial questions, and that this reliance can lead to wrong answers, we use a scene from a miniature room to ask models about size questions, as illustrated in Fig. 6.

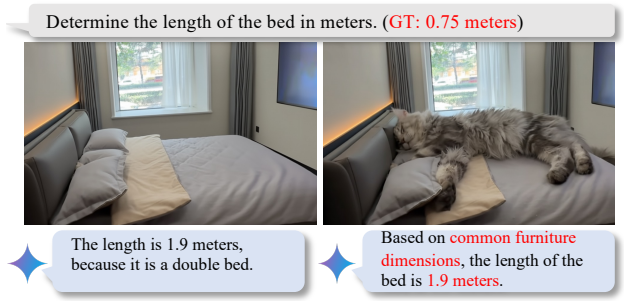


Figure 6. **Illustration of a Prior-Driven Reasoning Failure.** The scene is a 1:3 scale miniature bed. **(Left)** Gemini-2.5-Pro defaults to its internal knowledge, identifying the object as a "double bed" and outputting a prior-based estimate of 1.9m. **(Right)**Even when a strong, contradictory visual clue (a real cat) is introduced, the model fails to ground its reasoning in the visual evidence, and still defaults to the same "common furniture dimension" prior and outputs 1.9m.

B. Details of Data Collection

B.1. Hardware Specifications

Our custom-built data collection platform, illustrated in the main paper, is equipped with a multi-sensor suite designed for high-fidelity, pedestrian-centric data capture. The core components include:
- A synchronized stereo RGB camera system (rolling shutter, 1080p, 15 FPS).
- A 32-beam omnidirectional LiDAR (10 FPS).
- A high-frequency Inertial Measurement Unit (IMU) operating at 100 Hz.
- A GPS unit operating at 1 Hz.

All sensors are connected to an onboard Intel NUC mini PC running Ubuntu, which is orchestrated by the widely-adopted ROS2 framework. This system provides unified, single-command control over the entire sensor suite and logs all multimodal data streams as timestamp-synchronized ROS2 bag files, facilitating robust subsequent extraction and processing. The entire platform is powered by an onboard battery, which provides approximately one hour of continuous operation per charge.

To ensure smooth motion capture, the platform was mounted on a manual cart. The camera system was positioned at approximately 1.4 meters from the ground. This height was deliberately chosen as a trade-off to balance the need for a pedestrian-like perspective with the critical requirement of maintaining motion stability during collection.

B.2. Calibration and Rectification

A rigorous multi-sensor calibration pipeline was executed prior to all data processing to ensure the metric accuracy and spatio-temporal alignment of our dataset. This process was divided into three main components:

Stereo Camera Calibration and Rectification. We first calibrated the stereo camera system. This process involved using a standard chessboard pattern with the OpenCV library [8] to precisely determine the intrinsic parameters of each camera and the extrinsic transformation between the left and right camera units. Following calibration, a stereo rectification algorithm, also from OpenCV, was applied to all image pairs. This step is critical as it warps the two images such that their epipolar lines become collinear and horizontal, which allows for subsequent stereo matching and SLAM tasks. All visual data used in the downstream modules of our pipeline consists of these rectified stereo images.

LiDAR-to-Camera Calibration. To fuse visual and depth information, we calibrated the extrinsic parameters (the 6-DoF transformation) between the LiDAR and the left camera of the stereo system. This calibration was performed using the OpenCalib ToolBox [65], which provides robust automatic calibration. The resulting transformation matrix is essential for our pipeline, as it allows us to accurately project the 3D LiDAR point clouds onto the 2D image plane of the camera to generate the metric-scale sparse depth maps.

IMU Calibration. Finally, the Inertial Measurement Unit (IMU) was calibrated in two stages. We first determined its intrinsic parameters using the Kalibr toolbox [48]. Subsequently, the extrinsic transformation between the IMU and

the stereo camera system was computed using the OpenCalib ToolBox [65]. This precise IMU-camera extrinsic calibration is a prerequisite for ensuring the accuracy and robustness of the stereo-inertial SLAM module used for camera pose estimation.

B.3. Operators and Scene Selection

Operators were instructed to navigate the cart through the selected scenes at a typical walking speed. Turns were intentionally included in the routes to better represent natural pedestrian movement and to generate diverse trajectories for the final evaluation tasks. The entire data collection process totaled over 100 person-hours.

As referenced in the main paper, our scene selection strategy focused on maximizing diversity while capturing typical environments a pedestrian encounters. Furthermore, other than outdoor scenes, we include large-scale shopping malls to incorporate complex indoor scenarios. Unlike the confined residential or office environments common in existing benchmarks, large malls feature open layouts and a significantly larger range of scales, presenting open-world spatial challenges. Representative samples of our chosen scenes are shown in Fig. 7. We recognize that open-world environments can be more spacious and semantically sparse compared to object-dense indoor scenes. Therefore, our collection protocol intentionally prioritized pedestrian-centric areas known for a high density of potential query targets (e.g., street furniture, signage, complex storefronts, and other pedestrians) to ensure the final benchmark is both diverse and challenging.

Following a manual curation process, where we discarded sequences with poor visual quality (e.g., blur, low light) or those lacking distinct queryable objects, we obtained a 20-hour high-quality multimodal dataset. While approximately 6 hours of this data were used for the benchmark construction detailed in this paper, the remaining dataset, including all raw sensor data, will be made publicly available to the research community to foster further development in spatial intelligence.

C. Details for OSI-Bench construction

Video Segmentation and Synchronization. For processing efficiency, we segment the long-form raw recordings **logically** rather than physically. Instead of duplicating or moving source data, each clip is defined by a `JSON` file that bundles all necessary, timestamp-synchronized metadata, including paths to the stereo image frames, LiDAR PCD files, and IMU/GPS data. These logical clips are defined with randomized durations between 15 to 30 seconds. This duration, shorter than those commonly used in indoor datasets [17], is a deliberate design choice for two reasons: (1) the high semantic density of open-world scenes ensures sufficient complexity, comparable to much longer indoor recordings; and (2) it aligns with the sparse-frame sampling approach inherent to video loading for most MLLMs, ensuring effective capture of dynamic events.

Camera Pose Estimation. We employ ORB-SLAM3 [11] in its stereo-inertial mode, which utilizes our rectified 15 FPS stereo image sequences and IMU data. We found that processing the original full-resolution (1080p) frames was unreliable for this algorithm. Therefore, all input images were first downsampled to 960x540, and their corresponding camera intrinsic parameters were rescaled accordingly. This process yields a precise, metric-scale pose in a world coordinate system for every frame. This method was selected after an empirical evaluation on our data, where it demonstrated superior robustness and accuracy compared to the classical SfM pipeline COLMAP [50, 51].

Densified Depth Map Generation. To address the inherent sparsity of single-frame LiDAR scans, we leverage the estimated camera poses to perform multi-frame point cloud fusion. For each frame, point clouds from temporally adjacent frames are transformed and aggregated according to their relative poses, yielding a denser fused point cloud. This fused representation is then projected onto the image plane—following the projection protocol described earlier—to produce a densified depth map. These enhanced depth maps are essential for accurate downstream 3D information extraction. In our final implementation, we set the temporal fusion window to 3, meaning the point cloud for each frame is aggregated with those of its immediately preceding and succeeding frames.

Keyframe Selection. We extract keyframes from each video clip at a fixed 30-frame interval. This sampling rate (equivalent to 2 seconds at 15 FPS) was chosen to strike a balance: it is frequent enough to capture the vast majority of objects appearing in the pedestrian-speed video, yet sparse enough to avoid redundant annotation efforts on highly similar, consecutive frames.

Captioning with MLLMs. For captioning objects in the keyframes, our initial approach adopted the pipeline from GroundedSAM [49], using an open-vocabulary tagging model [76] to generate candidate class names. However, we observed that for our pedestrian-centric, open-world scenes, these models exhibit a strong bias towards labeling large, semantically dominant regions. This resulted in a high density of unsuitable, non-object labels such as 'road', 'sky', or 'buildings', which are ill-suited as query targets for a spatial benchmark. Thus, we choose to instruct a locally-run MLLM, Qwen-2.5-VL-8B-Instruct [6] to identify multiple semantically clear, physically distinct objects in each keyframe. For each object, the model generates a detailed caption describing only its intrinsic properties (*e.g.*, 'a red fire hydrant') and classifies it as either static or dynamic. This yields high-quality, relevant textual descriptions for subsequent grounding. See Fig. 8 for the prompt used in

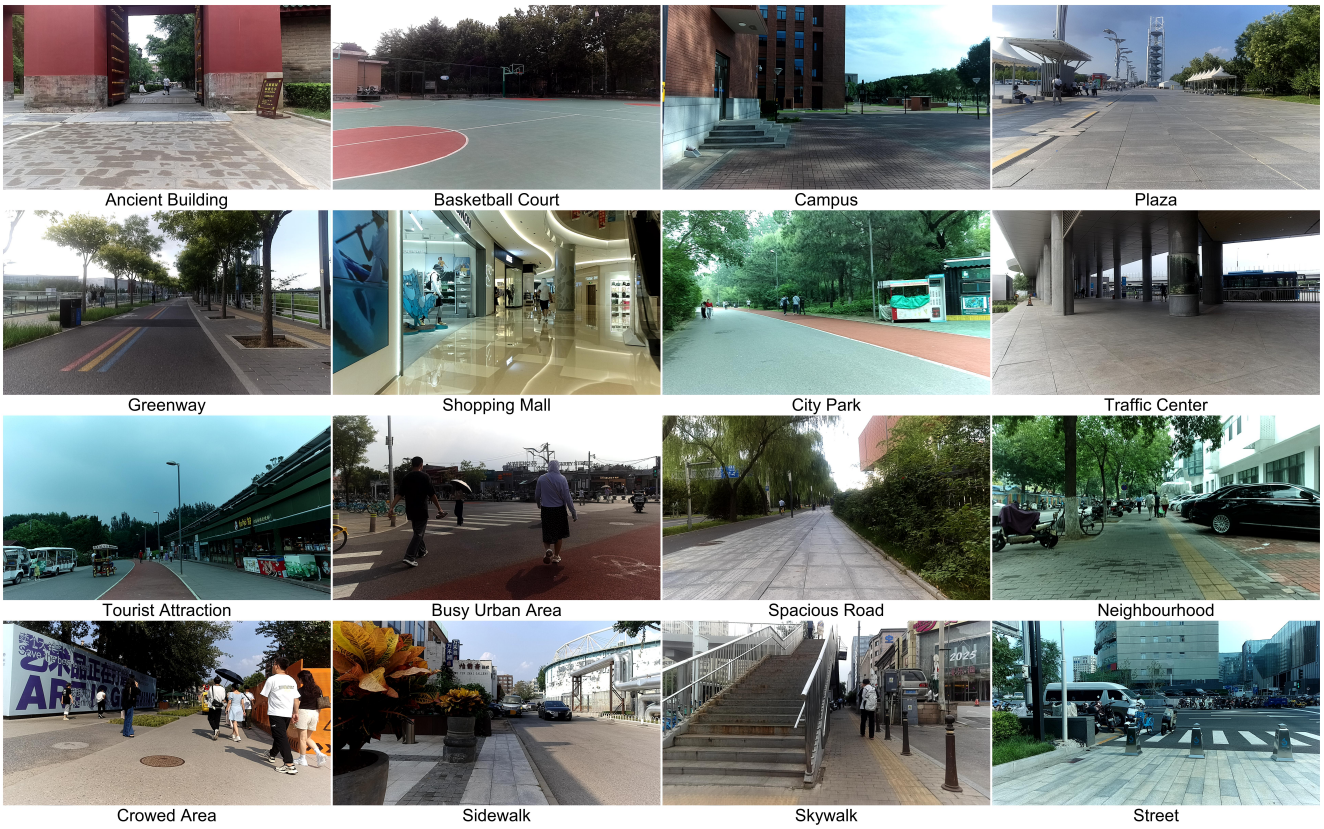


Figure 7. **Representative Samples of Scenes in OSI-Bench.**

this phase.

Object Detection and Segmentation. The object captions generated in the previous step serve as text prompts for GroundingDINO [42] to produce 2D bounding boxes. These boxes are then refined into pixel-level segmentation masks by the SAM model [32]. As each caption corresponds to a single object instance, we select only the highest-confidence detection per caption, followed by Non-Maximum Suppression (NMS) to resolve potential part-whole ambiguities.

Temporal Object Tracking. To ensure robust temporal consistency, we employ a point-tracking paradigm over a mask-propagation approach. While recent segmentation models like SAM2 [47] offer tracking, their propagation-based method can lead to drift and errors in long-term tracking. We use a point tracking model (CoTracker3 [31]) to establish motion correspondence. We sample points within an object's keyframe mask, track them bidirectionally, and then use these tracked points on non-key frames with SAM to generate a final, temporally coherent mask sequence.

3D Spatial Registration. To obtain each object's 3D representation, we use the per-frame mask to extract the corresponding depth values from the depth map. These values are then used to deproject the masked pixels into a 3D point cloud, whose centroid serves as the object's estimated 3D position relative to the camera. By combining this relative position with the frame's global camera pose, we register all objects into a unified world coordinate system. We note that as our LiDAR data primarily captures the front-facing surfaces of objects, we do not provide estimations of their full 3D dimensions (width, height, depth) in the current version of the benchmark. A final de-duplication step is performed in this world space to merge instances of the same static object detected in different keyframes.

See Fig. 16 for detailed samples illustrating the Joint-Annotation Module's workflow.

Template-based Generation. We use a template-based approach to generate all question-answer pairs in order to separate and test the models' spatial reasoning capabilities, minimizing the influence of complex language understanding or multi-step logical reasoning as confounding variables. The full templates for 9 tasks are shown in Tab. 5.

MLLM-assisted Curation. First, a powerful closed-source MLLM (e.g., Gemini-2.5-Pro [58]) performs an initial pass to correct inaccurate captions and filter out questions related to objects that are poorly visible due to occlusion or detection errors. Second, the MLLM assigns a confidence score to the remaining QA pairs, flagging those with potential am-

Captioner Prompt

[Task]
Your task is to analyze an image and output a JSON object containing lists of captions forstatic and dynamic objects.
[Rules]
1. Content Principle: Identify distinct physical objects. Each string in the lists must describe a **single entity**. Avoid plural or group descriptions (e.g., instead of “cars”, list “blue sedan”, “white SUV”).
2. Ignore: Backgrounds (walls, roads), 2D elements (text, signs), minor parts of larger objects, natural or amorphous categories like “tree”, “bush”, “flowers”, “grass”, or “cloud”.
3. Uniqueness: If an object has already been listed, do not list it again. Once no new unique objects can be found, STOP immediately and close the JSON list with a bracket.
4. Classification: The identified objects should first be classified into static or dynamic.
static objects: Stationary items. (e.g., a parked car, a bench, a trash bin).
dynamic objects: Items visibly in motion. (e.g., a person walking, a car driving, a bird flying).
5. Captions: All captions must be purely visual descriptions (e.g., “red sports car”, “person in yellow jacket”). Do not describe motion (e.g., avoid “person walking”).
6. Empty Lists: If no objects of a certain type are found, use an empty list “[]”.

[Example]

```
{
  "static_objects": [
    "red sports car",
    "black street lamp",
    "green park bench"
  ],
  "dynamic_objects": [
    "man in blue jacket"
  ]
}
```

Your entire response must be a single, valid JSON object, strictly following the format shown in the ‘[example]‘. Do not add any other text.

Figure 8. **Prompt for the MLLM captioner.**

biguities or unstable visual tags. Finally, human annotators conduct a final review of all low-confidence samples, either correcting or discarding them.

See Fig. 13, Fig. 14 and Fig. 15 for more examples in OSI-Bench.

D. Error Analysis of Benchmark Construction

Unlike benchmarks built upon existing, manually annotated 3D datasets [24, 37, 66], OSI-Bench employs a highly automated pipeline to extract spatial information and generate QA pairs. While this automation enables scalability, it is crucial to analyze the potential sources of error. In this section, we analyze these errors and demonstrate that they are minimal, and that the quality of OSI-Bench is ensured through our rigorous calibration, validation, and curation processes.

Three primary aspects could introduce errors into the final ground truth answers: (i) the extrinsic and intrinsic calibration of the sensors, (ii) the pose estimation from the SLAM algorithm, and (iii) the final 3D registration of objects in the world coordinate.

D.1. Errors in Calibrations

To ensure data fidelity, we conducted several independent calibration processes for all sensors and selected the optimal results. The quality of the stereo calibration can be measured by its *reprojection error*, which quantifies the distance between a detected pattern keypoint and its corresponding point projected from the other camera view. Using a standard chessboard pattern, our final selected calibration (from 25 image pairs) achieved a mean reprojection error of **0.32 pixels**.

Task	Question Template
Relative Distance	Measuring from the closest point of each object, which of the following is closest to the {Q_class}(id:{Q_id}): {A_class}(id:{A_id}), {B_class}(id:{B_id}), {C_class}(id:{C_id}), or {D_class}(id:{D_id})?
Relative Direction	If I am standing by the {C_class}(id:{C_id}) and facing the {A_class}(id:{A_id}), is the {B_class}(id:{B_id}) to my front-left, front-right, back-left, or back-right? The directions refer to the quadrants of a Cartesian plane.
Qualitative Ego-Motion	Assuming the video is recorded from a first-person perspective, which of the provided options best describes the person's overall movement throughout the entire duration of the video? Choose from straight, left turn, right turn or U turn.
Object 3D Localization	At approximately {T} seconds into the video, what is the Euclidean distance of the {A_class}(id:{A_id}) from the camera in meters?
Absolute Distance	What's the distance between the center of the {A_class}(id:{A_id}) and the {B_class}(id:{B_id}) in meters?
Depth-aware Counting	At approximately {time_sec}s, how many {class_name}s are visible within {distance_threshold} meters from the camera?
Absolute Displacement	What is the displacement distance of the {A_class}(id: {A_id}) between {T1}s and {T2}s in meters?
Absolute Speed	What is the average speed of the {A_class}(id: {A_id}) between {T1}s and {T2}s in m/s?
Quantitative Ego-Motion	How long has the camera travelled throughout the entire duration of the video in meters?

Table 5. **Templates used for question-answer pairs generation.**

Similarly, the LiDAR-to-camera calibration quality was measured by the reprojection error between 3D LiDAR points and their corresponding 2D image keypoints. Using 25 pairs, the mean reprojection error was **0.51 pixels**. We further validated this by measuring the planarity error between the checkerboard plane in the LiDAR point cloud and in the camera view. The mean translation error was **0.002 meters** and the mean rotation error was **0.978 degrees**, indicating a highly accurate spatial alignment between the sensors.

D.2. Errors in SLAM Pose Estimation

A quantitative evaluation of the final pose accuracy from ORB-SLAM3 [11] on our dataset is not feasible due to the lack of ground-truth trajectories in our collection. Instead, we rely on the extensive public validation of the algorithm itself. The original ORB-SLAM3 paper, for example, reports an Absolute Trajectory Error (ATE) of **0.035 meters** on comparable stereo-inertial datasets, demonstrating its high metric precision.

D.3. Errors in World 3D Registration

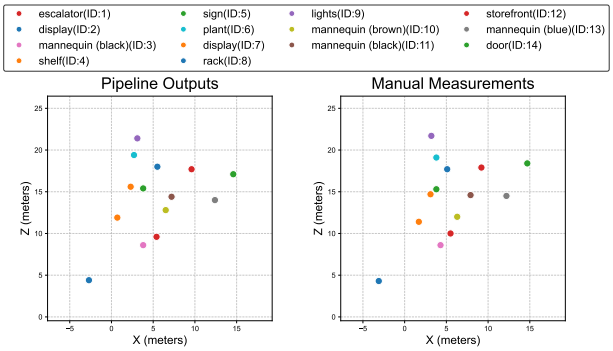
Several potential error sources could introduce error to the final 3D registration of semantic objects. These include visual occlusions, imperfect segmentation masks, minor temporal misalignments between the keyframe and the fused depth map, and the approximation of an object's center using its visible point cloud centroid.

To provide a qualitative validation of the pipeline's end-to-end accuracy, we conducted a real-world verification. We conducted this verification in two representative scenes: an indoor mall and an outdoor campus. For each scene, we first generated a 2D map of all static objects using our full pipeline. Subsequently, we returned to the physical locations to create a corresponding reference map by manually measuring the same objects' relative positions. As visualized in Fig. 9, the comparison reveals a low mean positional error between the pipeline output and the manual ground truth: 0.68 meters for the indoor scene and 0.79 meters for the outdoor scene, respectively. This result confirms the high metric fidelity of our automated generation pipeline.

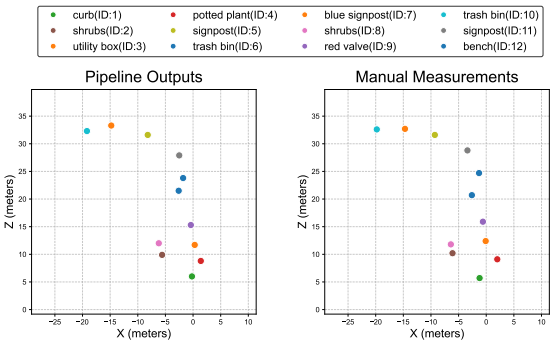
E. Evaluation Details

E.1. General Evaluation Setup

All evaluations are conducted using the VLMEvalKit framework [21] to ensure a standardized protocol, running on NVIDIA RTX 4090 GPUs. Unless specified otherwise, we employ a greedy decoding strategy (i.e., temperature=0, top-k=1, top-p=1) for all models to ensure reproducibility.



(a) Validation of our pipeline in an indoor mall scene.



(b) Validation of our pipeline in an outdoor campus scene.

Figure 9. **Qualitative Validation of the Pipeline.** We compare the map of static objects generated by our automated pipeline (left) against a ground-truth map of the same scene, which was measured manually on-site (right). The two maps are aligned for comparison.

For closed-source models, we evaluate Gemini-2.5-Pro, Gemini-2.5-Flash [58], GPT-5, GPT-4o [28], Claude-3.7-Sonnet, Claude-4-Sonnet [3], Doubao-Seed-Vision [26]. For open-source models, we evaluate Qwen2.5-VL [6], Qwen3-VL [6], InternVL2 [14], InternVL3.5 [14], LLaVA-OneVision [33], LLaVA-Video [77], and Ovis2 [43], covering their variants with difference scales.

The handling of the video frames modality varies by model. For all open-source models and the GPT series, we uniformly sample 32 frames per video as input. The Gemini-2.5-Pro model is the primary exception, as it supports native video ingestion, allowing us to provide the full MP4 file directly. See Sec. F.2 for more detailed discussion for this. Additionally, for models accessed via the OpenAI API (*e.g.*GPT series), we enable the low-quality image mode, which forces each frame to be processed at a fixed cost of 85 tokens.

The input for all models follows a standard structure: "[Pre-prompt][Question][Post-prompt][Video Frames]". The specific prompt templates for NA(Numerical Answers) and MCA(Multiple Choice Answers) are detailed in Fig. 10.

Prompts

Pre-prompt:
"These are frames of a video. In the video, objects are identified by numeric tags shown nearby. With that in mind, please answer the following question based on the video."
Question:
NA: question_text.
MCA: question_text + options.
Post-prompt:
NA: "Your answer must be only the final numeric value, without units or any other text."
MCA: "Your answer must be only the single letter (e.g., A, B, C, or D) of the correct option."

Figure 10. **Prompts employed when constructing inputs for evaluations.**

E.2. Metrics and Baselines

Although the prompt explicitly instructs the models to output only the final answer, we implement a regex-based fallback mechanism to handle non-compliant outputs. This parser extracts the last occurring numerical value or a valid multiple-choice option from the full generated text.

We employ two distinct metrics based on the answer format: standard *Accuracy* ($\mathcal{ACC}$) for MCA questions, and *Mean Relative Accuracy* ($\mathcal{MRA}$) for NA questions. $\mathcal{MRA}$ measures the proportion of predefined error thresholds that the Mean Relative Error (MRE) can pass:

$$\mathcal{MRA} = \frac{1}{10} \sum_{e \in \mathcal{C}} \left(\frac{|\hat{y} - y|}{y} < e \right), \tag{2}$$

where $\hat{y}$ and y denote the prediction and ground truth, respectively, and $\mathcal{C} = \{0.05, 0.10, \ldots, 0.50\}$ is the set of 10 error thresholds. This metric results in a step-wise score.

A special consideration is required for dynamic tasks where a near-zero ground truth makes the MRE denominator numerically unstable. We address this by defining a small threshold, $\mathcal{ST}$. For any ground truth $y < \mathcal{ST}$, a prediction $\hat{y} < \mathcal{ST}$ is awarded a full score of 1.0. If the prediction is incorrect ($\hat{y} \geq \mathcal{ST}$), the MRE is calculated with the denominator floored to $\mathcal{ST}$ to avoid division by zero. This threshold is set to a small value in practice, ensuring it only affects genuinely stationary or near-stationary cases.

For the four-choice MCA tasks, the chance-level baseline is 0.25. In contrast, we define the baseline for all NA tasks as *zero*. This is because, for an uninformed

guesser agnostic to the data distribution, the answer space is effectively unbounded—potentially ranging from centimeters to hundreds of meters—making any specific numerical guess fundamentally arbitrary. We argue that alternative baselines, such as the "Frequency Chance Level" proposed by [66], are invalid as they represent a data leak from the ground truth distribution.

E.3. Human Evaluation Setup

The human evaluation subset is a balanced collection of 270 questions, created by uniformly sampling 30 questions from each of our nine tasks. To ensure a consistent and high-quality evaluation, human annotators were given the instructions in Fig. 11.

Our human evaluation protocol follows the methodology of [66], allowing annotators unrestricted control over video playback (e.g., play, pause, re-watch) to gather comprehensive information. Additionally, we employ a calibration phase (or "warm-up" phase) specifically for open-world scenes, as we recognize the difficulty for human annotators to estimate real-world metric values based solely on visual inputs without training. Five human evaluators complete the entire subset independently and their scores are averaged to get the final human performance.

E.4. Details of Synthetic Scenes.

We generated our synthetic indoor data using the Blender Engine. First, we manually modeled 20 distinct indoor scenes spanning common layouts like washrooms, bedrooms, and living rooms. These scenes featured objects with conventional, real-world scales. We then leveraged the engine's internal metadata (e.g., object dimensions and locations) to automatically generate 120 template-based questions for our **Normal Set**, covering object size and distance tasks. Next, we created a parallel **Abnormal Set** to serve as our testbed. For each of the 20 scenes, the object scales were deliberately manipulated to be counter-intuitive (e.g., a tiny bathtub, an oversized plant). Crucially, the overall scene layout and camera positions were kept identical to the Normal Set, isolating the variable of object scale. The same automated QA generation pipeline was then applied to this altered metadata to produce a corresponding set of 120 questions.

In the generated video, the camera is positioned at the room's center and performs two full 360-degree pans to ensure complete object coverage: the first pan is executed with a downward tilt, and the second with an upward tilt. Human evaluators for this test followed the same instructions as in the main evaluation.

See Fig. 12 for a sample of our synthetic test set and Tab. 6 for detailed results.

Model	Task	Normal	Abnormal	Drop (Δ)
Gemini-2.5-pro	Distance	37.3	33.2	4.1
	Size	54.7	29.7	25.0
	Overall	**46.0**	**31.4**	**14.6**
Qwen2.5-VL-32B-Instruct	Distance	43.0	33.0	10.0
	Size	54.5	28.3	26.2
	Overall	**48.8**	**30.7**	**18.1**
Human Performance	Distance	51.7	51.4	0.3
	Size	62.5	60.5	2.0
	Overall	**57.1**	**56.0**	**1.1**

Table 6. Performance Degradation from Normal to Abnormal Conditions.

E.5. Details of Geometric Information Test

To design this experiment, we first sampled *absolute distance* questions from OSI-Bench, filtering for cases where the two queried objects are not co-visible in any single frame. This selection criterion necessitates that the model reason using camera ego-motion, rather than simply calculating the distance between two objects in a static image. For this specific subset of questions, we then extracted the raw metadata $(p_1, p_2, t_1, t_2, R, T)$ from our pipeline's intermediate outputs and formatted the tasks as shown in Tab. 7.

F. More Results

F.1. Full Evaluation Results

In addition to the key results shown in the main paper, we evaluated other models such as the full InternVL2 series [14] and Qwen2.5VL series [6]. We provide the comprehensive results for all generalist models tested in our study in Tab. 8.

During this extensive evaluation, we found that two models, Claude-4-Sonnet and Grok5 (with 8-frame input), exhibited collapsed performance. This was due to a consistent failure to adhere to the task's prompt instructions, rather than a specific failure in spatial reasoning.

F.2. Effects of Input Frames

For all evaluated models, with the exception of the Gemini family, we uniformly sample 32 frames per video as image inputs. These frames are presented chronologically, and the prompt is augmented with temporal information (e.g., video duration and number of sampled frames) to enable time-related reasoning. The Gemini family is the primary exception, as it supports native video ingestion; we provide the full MP4 file directly, as we consider this an integral part of its capability. A second exception is Grok-5, which was limited to 8 frames due to the constraints of the API we used.

We then conducted an ablation study to quantify the impact of frame count on performance, using Qwen3VL-32B (the top-performing open-source model) as our testbed. The

Human Annotator Instructions

- Annotators are permitted unrestricted control over video playback. This includes the ability to play, pause, scrub the progress bar, and re-watch the video multiple times for each question to ensure their answer is as accurate as possible.
- The provided object captions (class names) may occasionally be imprecise. In cases of a conflict or discrepancy between the textual caption and the numerical ID tag shown in the video, the numerical ID should be considered the definitive ground truth. The object designated by the visual ID tag is the correct target for the question.
- For Numerical Answer (NA) tasks, provide the numerical value only, without any units (e.g., 12). For Multiple-Choice Answer (MCA) tasks, provide only the corresponding capital letter of the correct option (e.g., A).
- Before beginning the formal evaluation, annotators are provided with a calibration set. This set consists of 2 sample videos, their 20 corresponding question-answer pairs, and the associated ground truth (GT) answers. Annotators are instructed to review this material to familiarize themselves with the camera properties, the various question types, and to gain a reasonable sense of the metric scales used in the benchmark. This step ensures all annotators are properly calibrated before proceeding to the main evaluation tasks.

Figure 11. **Human annotator instructions for evaluation.**

Component	**Content**
Question	What's the distance between the center of the {A_class}(id: {A_id}) and the {B_class}(id: {B_id}) in meters?
Formula post-prompt	In camera coordinates, x points right, y points down, and z points forward. To solve this, apply the following formula: $Distance = \|(R \cdot p_2 + T) - p_1\|$. In this formula, p_1 is the 3D position in the camera coordinate of first queried object observed at the earlier time t_1, and p_2 is the 3D position of second queried object observed in the camera coordinate at the later time t_2. The matrix R and vector T represent the rotation and translation the camera pose has changed at time t_2 relative to time t_1. If any piece of information required to use the formula is not present in the text, you must infer it from the video and then use it in the formula.
Obj1 Info	At {t1}s(t_1) of the video, {B_class}(id: {B_id}) is located at p_1 = [{p1_x},{p1_y},{p1_z}] meters relative to the camera.
Obj2 Info	At {t2}s(t_2) of the video, {A_class}(id: {A_id}) is located at p_2 = [{p2_x},{p2_y},{p2_z}] meters relative to the camera.
Ego-motion Info	Between {t1}s(t_1) and {t2}s(t_2), the camera's relative translation is T = [{T_x}, {T_y}, {T_z}] and the rotation matrix is R = [[{R11}, {R12}, {R13}], [{R21}, {R22}, {R23}], [{R31}, {R32}, {R33}]].
Setting	**Question Template**
Vanilla	{Question} {Formula post-prompt}
+ One Localization(p_1)	{Obj1 Info} {Question} {Formula post-prompt}
+ Both Localization(p_1,p_2)	{Obj1 Info} {Obj2 Info} {Question} {Formula post-prompt}
+ Ego-Motion(R, T)	{Ego-motion Info} {Question} {Formula post-prompt}
+ All(p_1, p_2, R, T)	{Obj1 Info} {Obj2 Info} {Ego-motion Info} {Question} {Formula post-prompt}
+ All(w/o Formula)	{Obj1 Info} {Obj2 Info} {Ego-motion Info} {Question}

Table 7. **Templates for the Geometric Information Test.**

(a) Normal scene.

(b) Abnormal scene. From the same perspective.

Figure 12. Samples from our synthetic test set.

results, shown in Tab. 9, indicate that while increasing the number of sampled frames yields a slight performance benefit, the overall gain is marginal.

F.3. Results of Spatial Models

As a comparison, we also evaluated three specialized spatial models: SpatialRGPT-VILA1.5-8B [15], Space-Thinker-Qwen, and SpaceOm [13] (the latter two finetuned on Qwen2.5-VL-3B). The results are presented in Tab. 10. We warn, however, that these scores are not a direct, like-for-like comparison due to fundamental misalignments in task design. SpatialRGPT is a region-prompted model, and to adapt it to our whole-video tasks, we provided a full-image mask as input to give access for the whole depth map to the model. Similarly, SpaceThinker and SpaceOm were not specifically trained for video-based reasoning, which may explain their performance relative to their base models. Therefore, these results are reported primarily as an initial reference.

F.4. Detailed Results of the Comparison for Model Generations

See Tab. 11 and Tab. 12 for the detailed results comparing the QwenVL and InternVL families across different sizes and generations on both OSI-Bench and VSI-Bench. While a direct, like-for-like comparison at each model size is constrained by model availability, two primary findings emerge from this data: (i) on OSI-Bench, the relationship between performance and model size is non-monotonic, often saturating or degrading; and (ii) the performance gain from newer generations on OSI-Bench is marginal, which stands in stark contrast to the significant and stable gains reported on VSI-Bench (*e.g.*>+23.0 for both families and all sizes).

Our main paper's analysis focuses on the comparison between OSI-Bench and VSI-Bench [66], as their shared video modality and tasks allow for a direct comparison. To validate that our findings regarding illusory progress on indoor benchmarks are not an artifact of VSI-Bench alone, we conducted an additional test on the multi-view indoor spatial benchmark, All-Angle-Bench [70]. The results in Tab. 13 confirm that the trend of performance gains from larger model scales and newer model generations persists on this benchmark as well. However, the magnitude of the gain from newer model generations is less significant than that observed on VSI-Bench. We hypothesize that this reduced gain may be due to the models' lack of specific training for the multi-view data format required by All-Angle-Bench.

	InternVL2	InternVL3.5	QwenVL2.5	QwenVL3
2B	21.7	21.7	—	17.1
3B/4B	21.7	23.7	24.2	21.1
7B/8B	24.5	28.5	27.1	31.2
14B/26B	26.0	28.5	—	—
32B[1]	22.9	26.9	30.0	32.2
72B/76B	25.5	—	26.5	—

[1] This line also includes 38B/40B for InternVL series.

Table 11. **Comparisons of models on OSI-Bench across size and generations.**

	InternVL2	InternVL3.5	QwenVL2.5	QwenVL3
2B	26.7	52.8	—	53.9
3B/4B	33.5	56.6	27.9	58.4
7B/8B	37.6	56.1	36.8	59.4
32B[1]	37.3	61.4	37.7	61.5

[1] This line also includes 38B/40B for InternVL series.

Table 12. **Comparisons of models on VSI-Bench [66] across size and generations.**

	InternVL2	InternVL3.5	QwenVL2.5	QwenVL3
2B	40.9	45.9	—	44.6
3B/4B	43.2	49.0	42.7	49.2
7B/8B	47.7	52.1	48.5	50.9
14B/26B	50.3	51.0	—	—
32B[1]	50.8	53.7	54.6	55.7
72B/76B	50.9	—	55.1	—

[1] This line also includes 38B/40B for InternVL series.

Table 13. **Comparisons of models on ALL-Angles-Bench [70] across size and generations.**

G. Privacy

To address privacy concerns, our data acquisition was conducted exclusively in publicly accessible locations. Subsequently, all collected data underwent a rigorous human curation process to identify and remove any potentially sensitive or private information.

Methods	Rank	Avg.	Rel. Dis.	Rel. Dir.	Qual. S-Motion	Obj. Loc.	Abs. Dis.	Depth Count	Abs. Displ.	Abs. Speed	Quan. S-Motion
			Relational($\mathcal{MCA}$)			Static Metric($\mathcal{NA}$)			Dynamic Metric($\mathcal{NA}$)		
Against Human on tiny											
Human-level	-	60.3	85.7	83.3	73.7	43.9	39.2	67.5	42.9	65.8	66.8
Gemini-2.5-Pro	-	36.8	53.1	23.1	46.7	39.7	33.8	40.3	22.2	27.8	40.0
GPT-5	-	27.9	37.5	30.8	40.0	35.3	25.3	12.8	9.2	31.4	33.0
Qwen2.5VL-32B-Instruct	-	32.1	68.8	23.1	33.3	14.4	29.7	31.3	17.5	32.8	35.7
Closed-source Models											
Gemini-2.5-Pro	-	37.2	50.0	28.1	52.5	37.4	28.1	37.9	26.8	31.1	40.8
Gemini-2.5-Flash	-	19.5	17.9	2.8	50.6	22.7	16.1	26.8	8.1	6.8	20.0
GPT-5	-	29.7	34.4	33.1	49.5	32.5	23.7	20.9	10.5	33.8	30.6
GPT-4o	-	25.9	30.8	29.1	42.2	22.9	27.0	21.6	17.5	15.5	28.8
Claude-3.7-Sonnet	-	26.5	38.9	32.8	47.6	31.3	22.4	31.5	5.2	30.1	5.0
Doubao-Seed-1.6V	-	27.3	35.9	24.1	44.0	16.6	18.9	38.7	25.7	31.8	9.2
Claude-4-Sonnet	-	13.2	7.2	0.7	2.8	20.0	21.6	26.4	9.0	17.9	9.0
Grok5(8f)	-	13.6	23.5	19.6	2.1	10.1	10.5	24.2	12.8	13.9	7.9
Open-source Models											
InternVL2-1B	-	19.6	33.2	32.1	40.8	7.6	15.4	21.8	7.1	14.3	8.8
InternVL2-2B	-	17.1	30.4	18.6	40.6	7.0	7.8	32.9	4.7	13.2	0.1
InternVL2-4B	-	21.7	27.2	28.1	40.6	14.6	21.0	36.9	12.2	17.2	0.0
InternVL2-8B	-	24.5	35.1	31.7	40.8	21.8	17.8	39.7	15.0	17.8	3.8
InternVL2-26B	-	26.0	34.5	34.0	40.7	28.9	20.6	38.3	12.3	23.6	4.6
InternVL2-40B	-	22.9	36.7	21.0	41.9	21.1	19.2	33.0	10.0	22.0	1.7
InternVL2-76B	-	25.5	32.3	30.6	41.3	15.8	13.5	39.9	15.3	25.0	18.2
InternVL3.5-1B	-	19.0	33.4	33.0	36.5	2.4	1.0	28.1	7.0	14.8	20.8
InternVL3.5-2B	-	21.7	34.8	32.1	40.1	3.8	2.7	40.8	11.5	16.6	17.0
InternVL3.5-4B	-	23.7	37.6	32.8	44.8	4.6	6.7	40.7	15.6	23.4	10.7
InternVL3.5-8B	-	28.5	37.6	33.6	47.3	12.2	13.2	42.3	20.3	30.2	21.5
InternVL3.5-14B	-	28.5	40.3	33.9	47.1	15.5	15.6	42.8	21.8	32.1	9.0
InternVL3.5-38B	-	26.9	40.2	34.0	45.3	11.6	7.7	42.7	20.3	31.4	11.1
Qwen2.5VL-3B-Instruct	-	24.2	30.3	32.7	43.0	17.2	26.1	18.4	13.3	19.1	21.1
Qwen2.5VL-7B-Instruct	-	27.1	33.1	17.3	41.5	22.0	25.1	27.6	18.3	26.5	30.0
Qwen2.5VL-32B-Instruct	-	30.0	41.7	32.7	44.4	23.9	24.0	27.7	16.7	32.8	27.3
Qwen2.5VL-72B-Instruct	-	26.5	38.5	20.1	46.5	27.7	26.0	29.4	9.4	29.7	9.8
Qwen3VL-2B-Instruct	-	18.4	33.7	32.4	37.5	2.2	4.2	22.4	6.6	19.0	12.5
Qwen3VL-4B-Instruct	-	21.1	34.8	23.7	46.4	3.9	6.9	24.1	13.6	20.4	17.5
Qwen3VL-8B-Instruct	-	31.2	38.3	31.2	49.3	21.0	15.1	33.3	21.3	34.3	37.8
Qwen3VL-32B-Instruct	-	32.2	41.9	28.8	47.1	25.3	11.5	30.2	18.6	36.8	49.2
Qwen3VL-30B-A3B-Instruct	-	29.3	37.5	29.9	44.0	12.4	8.7	33.1	20.4	31.7	46.9
LLaVA-OneVision-0.5B	-	19.6	27.9	23.8	40.8	13.5	13.1	19.9	10.3	13.6	15.7
LLaVA-OneVision-7B	-	25.7	35.1	32.7	40.8	16.1	25.5	25.6	16.8	28.3	13.4
LLaVA-OneVision-72B	-	26.9	38.6	32.2	41.7	19.6	18.3	35.3	19.2	23.0	16.6
LLaVA-Video-Qwen2-7B	-	22.9	37.1	31.2	40.9	17.6	22.1	18.2	17.5	19.0	5.7
LLaVA-Video-Qwen2-72B	-	28.3	39.8	31.1	42.2	23.5	18.0	34.2	20.7	29.9	17.0
Ovis2-4B	-	25.9	34.6	30.3	40.1	16.1	18.6	36.8	17.7	22.6	18.7
Ovis2-16B	-	28.1	37.0	35.8	42.0	20.0	6.2	41.7	21.7	23.3	28.2
Ovis2-34B	-	26.8	37.3	35.5	40.8	19.1	13.1	37.5	18.7	27.4	15.5

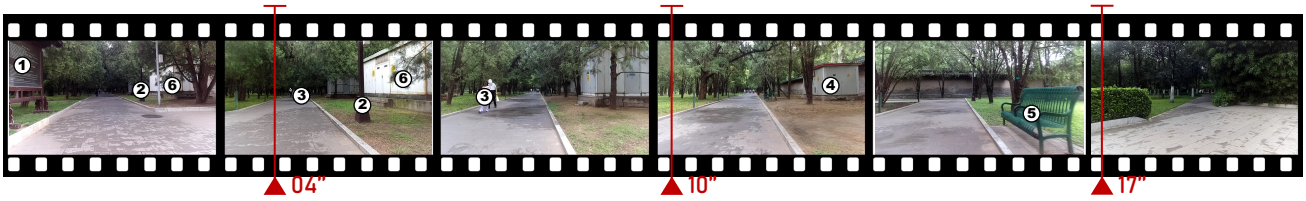
Table 8. **Full Evaluation results for all MLLMs we tested.**

Num. of Frames	Avg.	Rel. Dis.	Rel. Dir.	Qual. S-Motion	Obj. Loc.	Abs. Dis.	Depth Count	Abs. Displ.	Abs. Speed	Quan. S-Motion
		Relational			Static Metric			Dynamic Metric		
8	30.2	40.6	29.7	47.4	21.9	12.6	29.2	18.5	32.0	40.5
16	31.8	42.2	29.7	47.3	23.3	11.5	31.0	20.5	35.6	45.1
32	32.2	41.9	28.8	47.1	25.3	11.5	30.2	18.6	36.8	49.2
64	32.5	42.8	30.2	47.0	26.6	10.4	31.6	17.5	36.5	50.2

Table 9. **Ablation study on the number of sampled input frames for Qwen3-VL-32B.**

Model	Avg.	Rel. Dis.	Rel. Dir.	Qual. S-Motion	Obj. Loc.	Abs. Dis.	Depth Count	Abs. Displ.	Abs. Speed	Quan. S-Motion
		Relational			Static Metric			Dynamic Metric		
VILA-1.5-8B	20.4	30.5	30.0	40.8	17.7	25.2	17.1	15.9	8.8	1.3
SpatialRGPT-VILA-1.5-8B	24.0(+3.6)	31.9	33.0	40.8	23.1	16.1	26.0	19.5	13.9	15.5
Qwen2.5VL-3B-Instruct	24.2	30.3	32.7	43.0	17.2	26.1	18.4	13.3	19.1	21.1
SpaceThinker-Qwen-3B	21.7(-2.5)	29.3	31.9	40.8	0.2	0.3	38.4	6.2	28.0	23.6
SpaceOm-3B	23.2(-1.0)	29.8	26.3	44.0	13.0	23.4	26.5	16.1	22.6	8.5

Table 10. **Results for Specialized Spatial Models and Their Corresponding Base Model.**



Relative Distance:
Choose from sign board(id:1), electrical housing(id:4),green bench(id:5) and electrical housing(id:6): which one is the closest to street lamp(id:2)?
Answer: electrical housing(id:6)

Relative Direction:
If I am standing by electrical housing(id:6) and facing electrical housing(id:4) is sign board(id:1) to my front-left, front-right, back-left, or back-right?
Answer: back-left

Qualitative Ego-Motion:
Which one best describes the camera's overall movement throughout the entire video? Choose from 'straight', 'left turn', 'right turn' or 'U-turn'.
Answer: left turn

Object Localization:
At 10 seconds into the video, what is the Euclidean distance of the electrical housing(id:4) from the camera in meters?
Answer: 22

Absolute Distance:
What's the distance between street lamp(id:2) and electrical housing(id:6) in meters?
Answer: 4

Depth-Aware-Counting:
At 17 seconds into the video, how many trees are visible within 20 meters from the camera?
Answer: 2

Absolute Displacement:
What is the displacement distance of girl in pink (id:3) between 3s and 7s of the video in meters?
Answer: 8.9

Absolute Speed:
What is the average speed of girl in pink (id:3) between 4s and 6s of the video in m/s?
Answer: 2.1

Quantitative Ego-Motion:
How long has the camera travelled throughout the entire duration of the video in meters?
Answer: 36

Figure 13. **OSI-Bench examples.**(Part 1)

Relative Distance:
Choose from vacuum (id:1), fire extinguisher cabinet(id:3),square pillar(id:4) and fire extinguisher cabinet(id:6): which one is the closest to fridge(id:2)?
Answer: fire extinguisher cabinet(id:3)

Relative Direction:
If I am standing by square pillar(id:4) and facing fire extinguisher cabinet(id:6), is vacuum (id:1) to my front-left, front-right, back-left, or back-right?
Answer: back-right

Qualitative Ego-Motion:
Which one best describes the camera's overall movement throughout the entire video? Choose from 'straight', 'left turn', 'right turn' or 'U-turn'.
Answer: U-turn

Object Localization:
At 5 seconds into the video, what is the Euclidean distance of the fridge(id:2) from the camera in meters?
Answer: 4

Absolute Distance:
What's the distance between sign board(id:7) and vacuum(id:1) in meters?
Answer: 38

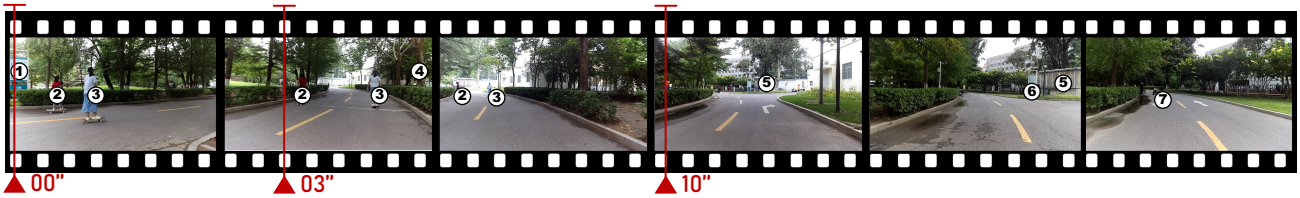
Depth-Aware-Counting:
At 22 seconds into the video, how many tables are visible within 15 meters from the camera?
Answer: 1

Absolute Displacement:
What is the displacement distance of person in black(id:5) between 7s and 12s of the video in meters?
Answer: 9.8

Absolute Speed:
What is the average speed of person in black(id:5) between 7s and 9s of the video in m/s?
Answer: 1.9

Quantitative Ego-Motion:
How long has the camera travelled throughout the entire duration of the video in meters?
Answer: 47

Figure 14. **OSI-Bench examples.**(Part 2)



Relative Distance:
Choose from tree(id:4), sign board(id:1), electrical housing(id:5) and vehicle(id:7): which one is the closest to sign board(id:6)?
Answer: electrical housing(id:5)

Relative Direction:
If I am standing by sign board(id:1) and facing tree(id:4), is electrical housing(id:5) to my front-left, front-right, back-left, or back-right?
Answer: front-right

Qualitative Ego-Motion:
Which one best describes the camera's overall movement throughout the entire video? Choose from 'straight', 'left turn', 'right turn' or 'U-turn'.
Answer: left turn

Object Localization:
At 0 seconds into the video, what is the Euclidean distance of the fridge(id:2) from the camera in meters?
Answer: 7

Absolute Distance:
What's the distance between sign board(id:6) and electrical housing(id:5) in meters?
Answer: 4

Depth-Aware-Counting:
At 10 seconds into the video, how many trees are visible within 10 meters from the camera?
Answer: 2

Absolute Displacement:
What is the displacement distance of girl in red(id:5) between 0s and 5s of the video in meters?
Answer: 16

Absolute Speed:
What is the average speed of girl in white(id:3) between 3s and 5s of the video in m/s?
Answer: 3.5

Quantitative Ego-Motion:
How long has the camera travelled throughout the entire duration of the video in meters?
Answer: 31

Figure 15. **OSI-Bench examples.**(Part 3)

Figure 16. **Samples showing the workflow of the Joint-Annotation Module.** In the actual implementation, multiple keyframes are processed for each scene.