Structured 3D Scene Representations Improve VLM Spatial Reasoning

Simona Kocour[1,2,3*] Yutong Chen[3] Torsten Sattler[1] Sergey Prokudin[3] Siyu Tang[3]

[1] Czech Institute of Informatics, Robotics and Cybernetics, Czech Technical University in Prague

[2] Faculty of Electrical Engineering, Czech Technical University in Prague [3] ETH Zurich

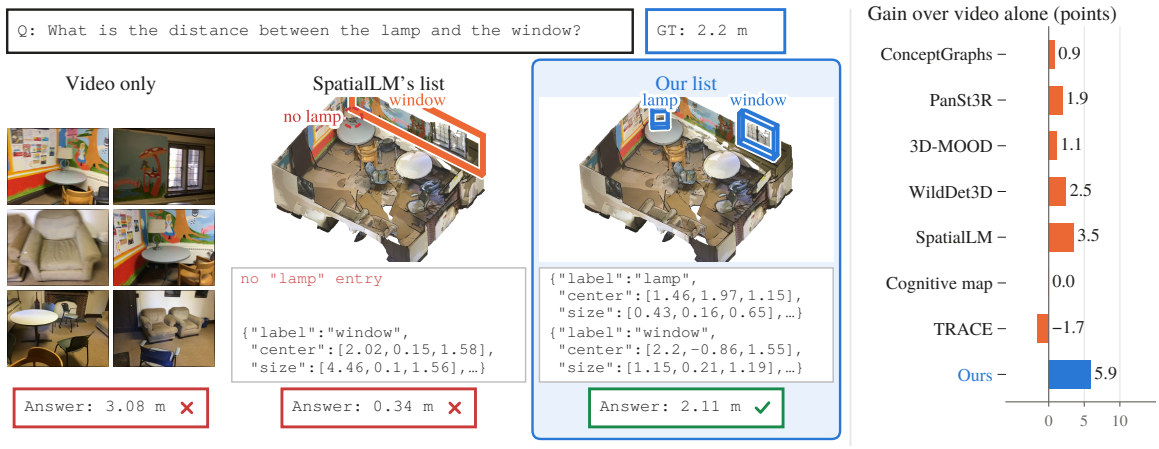

Figure 1. **The list must contain the objects a question asks about.** Left, a VSI-Bench question [69] answered by Gemini 3.8 Flash. From the video alone, it misjudges the distance. With the list of SpatialLM [48], built from the same video, it answers 0.34 m, because the list has no lamp and an enlarged window. With our list, which contains both objects with metric 3D boxes, it answers 2.11 m, close to the true 2.2 m. Right, the gain over the video alone on 800 VSI-Bench questions with the same VLM and frames (Tab. 4). Our list adds the most, while the lists of ConceptGraphs [22], PanSt3R [79], 3D-MOOD [73] and WildDet3D [30] and the prompting methods cognitive map [69] and TRACE [28] add little or nothing.

Abstract

Vision-language models (VLMs) still struggle to reason about space from video, misjudging distances, sizes and directions. Yet simply telling them where objects are can make a large difference. Given the video together with a text list of the objects in the scene and their ground-truth metric 3D boxes, Gemini 3.8 Flash answers correctly 94.8% of a sample of questions from the spatial benchmark VSI-Bench, compared with 68.6% from the video alone. This 26.2-point gain shows that the VLM can use 3D geometry to answer spatial questions much more accurately than it can infer that geometry from video. The challenge is to recover such a list from images. We therefore build an object-centric 3D representation with off-the-shelf reconstruction and segmentation models, once per scene and before any question is known. Given to the VLM alongside the original frames, it significantly improves accuracy on four spatial benchmarks, without fine-tuning any model. It also improves all five frontier VLMs we tested and provides larger gains than other 3D perception methods with the same VLM. In a controlled study, we introduce one type of error at a time into ground-truth lists to see which errors matter most to the VLM. Missing an object needed to answer a question costs the VLM two thirds of what the whole list adds. In contrast, adding unrelated objects barely affects its answers, although detection metrics count each of them as an error. Our results show that conventional detection metrics do not fully reflect what matters for VLM spatial reasoning. Covering every object a question may ask about matters more than avoiding unrelated detections.

1. Introduction

Suppose a vision-language model (VLM) watches a video of a room and is asked how far the lamp is from the window (Fig. 1). To answer, it must find both objects and work out

*Work done during a research visit at ETH Zurich.

where they are in 3D. Current VLMs often struggle with such questions [69]. One approach is to fine-tune them on spatial question-answering data. Some of the strongest models on VSI-Bench are trained on questions generated from the benchmark's source datasets [29, 70, 72]. Some questions can be answered without seeing the video. When those questions are removed in VSI-Bench-Debiased [4], fine-tuned models lose about a tenth of their accuracy [6, 33, 72]. This suggests that they may learn shortcuts from the questions instead of relying fully on the scene.

We study a different approach. We use the VLM off the shelf, and tell it where the objects are, without fine-tuning. Gemini 3.8 Flash achieves 68.6% average accuracy across eight spatial tasks using video alone (on a sample of 800 VSI-Bench questions [69]). When given the ground-truth objects and their metric 3D boxes as a text list alongside the video, its accuracy rises to 94.8%, a gain of 26.2 points (Tab. 3). This is an oracle experiment, so this is an upper bound, but the result shows that VLMs can use 3D geometry to answer spatial questions much more accurately than they can infer it from video alone.

The challenge is to recover such a list from the video. Existing 3D perception systems reconstruct scenes and detect objects, and their outputs can be converted into the same format. Yet these representations provide much smaller gains in VLM accuracy than the ground-truth annotations (Fig. 1, right and Tab. 4). Figure 1 shows one example: SpatialLM [48] reconstructs the room but enlarges the window, misses the lamp, and the VLM answers the distance question incorrectly. These systems are generally evaluated by how accurately they reconstruct, detect or segment objects, rather than by how well their outputs support spatial reasoning. A list for a VLM has a different requirement. Since the questions are not known when it is built, it needs broad object coverage and sufficiently accurate geometry to answer questions about any part of the scene.

We build such a list from images alone, using off-the-shelf models. A feed-forward reconstruction model reconstructs metric 3D geometry and camera poses [64], while an open-vocabulary segmenter finds objects across frames [8]. We lift the masks into 3D, merge observations of the same object across views, and fit a 3D box to each object. The list also includes the reconstructed floor area. It is built once per scene, before any question is known, and given to the VLM as text alongside the original frames.

With the same VLM and frames, our list improves accuracy by 5 to 8 percentage points on four spatial benchmarks, including two that were not used during pipeline development (Tab. 1), and exceeds the best published results on four of the five benchmarks. It improves all five frontier VLMs we tested (Tab. 3). The list representation complements the video. Under the same VLM, our list provides larger accuracy gains than the evaluated 3D perception methods [22, 30, 48, 73, 79], prompting the VLM to map [69] or describe the scene [28], or having the VLM generate its own list [52] from the same frames.

To understand what such a list must get right, we conduct a controlled study (Sec. 5). Starting from VSI-Bench's ground-truth lists, we introduce one type of error at a time, such as removing, duplicating, resizing or moving an object needed to answer a question, and measure the effect on VLM accuracy. Removing that object costs two thirds of what the whole list adds. Adding objects of unrelated categories, by contrast, barely affects the VLM, although detection metrics penalize them heavily. The VLM also often follows incorrect information in the list despite having the video. Oracle fixes of our own lists point the same way. Of five corrections, only restoring the missed objects is needed for a significant gain. Since the questions are unknown when the list is built, broad object coverage matters more than avoiding unrelated detections. Conventional detection metrics do not fully capture these priorities.

We make the following contributions:

- We show that frontier VLMs can reason accurately about 3D space when given explicit scene geometry. On a sample of VSI-Bench, five VLMs reach 85.5–95.5% accuracy with ground-truth object lists, without fine-tuning.
- We build such lists from images alone, using off-the-shelf models. Our list significantly improves spatial reasoning on four benchmarks and provides larger gains than the evaluated 3D perception and prompting methods under the same VLM.
- We study which errors in a 3D representation affect spatial reasoning. Our controlled experiments show that missing objects needed to answer a question is particularly costly, while conventional detection metrics often penalize errors that have little effect on the VLM.
- We will release the source code, the lists of all evaluated scenes, the prompts and VLM answers, together with the error-injection code and the protocol fixed before the controlled study

2. Related Work

Spatial reasoning in VLMs. VSI-Bench [69] evaluates spatial reasoning from indoor video walkthroughs, while subsequent benchmarks cover single and multiple images [34, 71, 77], mental models from sparse views [63], reasoning from another person's viewpoint [32], and situated questions in 3D scenes [45]. Some benchmarks contain questions that can be answered without visual evidence [46], a limitation examined by VSI-Bench-Debiased [4]. Most approaches improve spatial reasoning by fine-tuning VLMs on spatial question-answering data [6, 9, 12, 29, 34, 72], sometimes with 3D features [18, 33, 66], camera information [70], spatial memory [38, 42], or reinforcement learning [10, 36]. We study a complementary setting: the VLM is off-the-shelf and

receives, together with the frames, an explicit 3D representation reconstructed from them. This needs no fine-tuning, and applies to any VLM.

Structured scene representations for VLMs. Scene graphs provide a compact description of objects and their relationships and have been used for robotic planning, navigation, and embodied question answering [22, 47, 56, 65]. Explicit geometry has also been used to improve spatial reasoning without fine-tuning: SandboxVLM [40] reasons over bouding boxes, SpatialPIN [44] uses 3D priors inferred from a single image, and Struct2D [78] supplies bird's-eye-view renderings with object annotations and metadata. Ego3D-VLM [20] builds a geometry-based cognitive map from egocentric views. Closer to our setting, GR3D [75] reconstructs object geometry from images and links textual 3D attributes to objects marked in the images. GraFT [16] operates on a 3D scene graph through geometric tools, task-specific bird's-eye views, and selected frames, evaluating both ground-truth and predicted scene graphs. We instead provide the VLM with the video frames and a wide list of objects detected in the scene, with their 3D metric bouding boxes, constructed without knowing the questions. Our experiments examine not only whether this representation improves reasoning, but also how errors in the representation affect the answers.

Prompting and tool-based reasoning. Other training-free approaches help VLMs reason about space by marking objects across frames [37], adding reconstructed views [55], visualizing camera motion [35], or prompting the model to construct a cognitive map or describe the camera trajectory [28, 69]. Visual programs [24, 43, 60] and agents [11, 23, 49] instead invoke perception or geometric tools while answering individual questions. Our pipeline performs perception once per scene, outside the VLM. The same representation is then available for every question, without question-dependent perception, rendering, or tool calls. We compare against cognitive-map prompting [69], TRACE [28], and GCA [11] under the same VLM.

3D perception and downstream evaluation. Object-centric 3D representations can be recovered through open-vocabulary instance segmentation [51, 61, 68], panoptic reconstruction [21, 79], open-set 3D detection [7, 27, 73, 76], and scene-layout prediction [48]. These systems are generally evaluated using detection or segmentation metrics, which measure object coverage, false positives, and geometric agreement with ground truth. Prior work has analyzed detection errors separately [3, 25], while planner-centric metrics measure their effect on downstream decisions [53]. Language models are also known to follow incorrect textual context, including when it contradicts visual evidence [15, 58]. We study how missing, duplicated and geometrically inaccurate objects in a structured 3D input affect spatial answers. By measuring these effects in a controlled setting and testing them on predicted representations, we examine whether conventional detection metrics reflect the errors that matter for VLM reasoning.

3. A Structured 3D Representation

We represent a scene as a *text list* of objects, each described by a label and a metric, gravity-aligned 3D bounding box (Fig. 2). The list is constructed once per scene, before the questions are known. Our aim is therefore to recover every object that might be asked about, associate its observations across views without merging distinct objects, and preserve the geometry needed for spatial questions.

The pipeline takes only RGB frames, without camera intrinsics, poses or depth. For a video, we select the two sharpest frames per second, at most 106. For an image set, we use all images. Every learned component is an off-the-shelf model with released weights, and none is fine-tuned.

3.1. Metric geometry from images

Pi3X, the metric-scale version of π^3 [64], jointly estimates metric 3D points, confidence values and camera poses from the input frames. We discard pixels with a confidence below 0.6 and pixels on depth edges, where the depth varies by more than 3% within the pixel's 3×3 neighborhood, the rule of π^3's released code. When an absolute confidence threshold would remove most points, we additionally keep those above the scene's median confidence, following the relative filtering rule of VGGT [62]. We estimate a gravity-aligned coordinate frame from the predicted camera orientations and fitted planes, align its horizontal axes with the dominant walls, and place the floor at $z = 0$.

3.2. Finding and grouping objects

Because the questions are unknown, the object names can come neither from the question nor from a benchmark's labels. We use two sources of names. The first is a fixed vocabulary of 157 everyday indoor categories, which we wrote without consulting any benchmark's labels. The second comes from the scene itself. Florence-2 [67] writes a caption for each frame, and the 60 object names that occur most often in these captions are added to the vocabulary. SAM 3 [8] segments every name in every frame. Non-maximum suppression across names removes overlapping masks of the same object and keeps their names as alternative labels.

We lift each mask to the reconstructed 3D points of its pixels. A point counts as a depth outlier, and is removed, if its depth differs from the median depth of the mask by more than three median absolute deviations. Observations of the same object must then be associated across views, while distinct objects of the same category must remain separate.

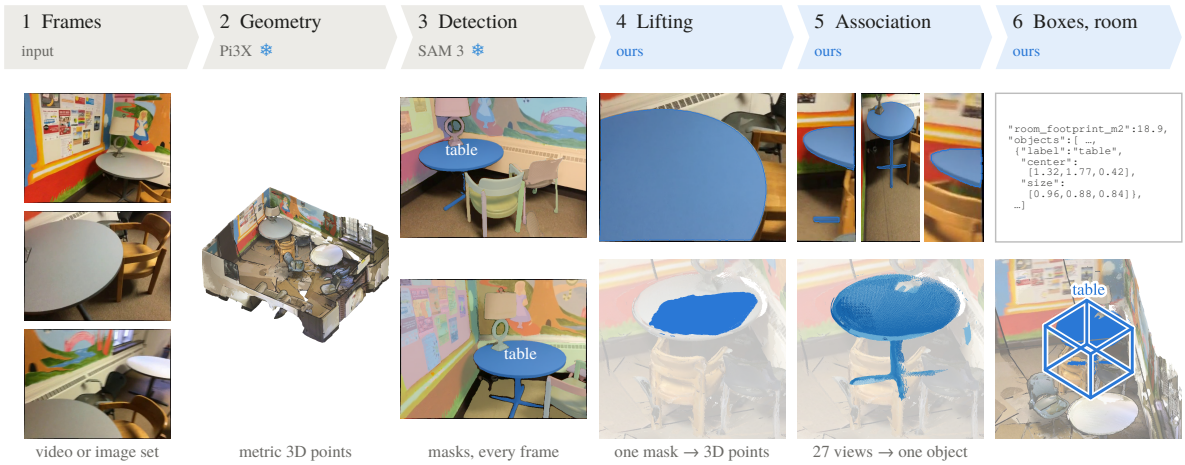

Figure 2. **Building a scene-level 3D representation from images.** Given video frames or an image set (1), Pi3X [64] reconstructs metric 3D points and camera poses (2). SAM 3 [8] segments the frames, prompted with a fixed vocabulary and with object names from captions of the frames (3). We lift the masks to 3D (4), associate observations across views into objects (5), and fit metric boxes (6). The list also gives the floor area. The VLM receives the list as text, together with the frames. Snowflakes mark off-the-shelf models that we do not fine-tune.

We link two observations from different frames if their 5 cm voxel sets overlap by at least a quarter of the smaller set and their centres are within 1.5 m. Links are merged greedily, starting with the strongest overlap, unless a merge would join incompatible observations, for example two disjoint masks with the same name in one frame (two chairs side by side), a mask inside a differently named mask of the same frame (a pillow on a sofa), or two objects seen in several frames under names that disagree. We keep an object only if it has at least 100 points seen in two or more frames, or at least 300 points, and remove floor, wall and ceiling. Each object receives the point-weighted majority label of its masks.

3.3. Boxes, room and interface

For each object, we fit a 3D bounding box. For a video, we add the room's floor area, measured on the reconstruction.

The list is written as JSON text, it is a structured 3D representation. Each object has a label, a centre and three side lengths in metres. The prompt explains the coordinate frame and warns that the list may omit, duplicate or split objects. When answering a question about a video, the VLM receives 32 evenly spaced frames. For an image set, it receives all images. The same list serves every question about the scene. The VLM answers each question in a single call, from the frames and the list together, without calling perception tools or changing the list. The prompt and the pipeline settings are the same on every benchmark.

4. Experiments

4.1. Setup

We evaluate on VSI-Bench [69], 5130 questions about 288 videos of indoor rooms from ScanNet [14], ScanNet++ [74] and ARKitScenes [2], and on VSI-Bench-Debiased [4], a subset of 2362 of its questions. We used VSI-Bench while developing the pipeline. Three further benchmarks were not used during development, MMSI-Bench [71] (1000 questions about sets of images), SPBench-MV [34] (319) and MindCube-Tiny [63] (1050).

Accuracy is the percentage of correct answers. For numerical answers such as distances, VSI-Bench and SPBench-MV give partial credit by how close the answer is, averaged over several tolerances (mean relative accuracy [69]), and both benchmarks average the accuracy over their tasks. A gain is a difference in accuracy, in percentage points. It is significant if its 95% confidence interval excludes zero, where the interval comes from a paired bootstrap that resamples scenes, or image sets for the image benchmarks [17]. Tables mark gains that are not significant with °.

All results use Gemini 3.8 Flash [54] unless another VLM is named. Table 3 also tests Gemini 3.5 Flash-Lite and 3.1 Pro [13], GPT-5 [59], GPT-6 Astra [52], and the open-weight Qwen3-VL-8B and 32B [1] (settings in the supplement). In every comparison, the VLM answers the same question with the same visual input, 32 frames of a video or all images of an image set, once without and once with a list, and only the list differs. The *ground-truth list* is built in our format from the benchmark's own 3D annotation, from which its answers were generated, and serves as an upper bound. Tables 1 and 2 use the full test sets. All other VSI-Bench results use the same 800 questions, 100 per task, drawn at random before the experiments (Tab. 3). The hypotheses and statistical tests for the controlled error study (Sec. 5) were specified before the experiments.

Table 1. **Results on five benchmarks.** Accuracy (%) on the full test sets. The right columns give Gemini 3.8 Flash with the same frames, without and with our list. For context, the left columns give the best results published at top-tier venues, with the VLM of each method without spatial training in parentheses. They use other VLMs and serve as context only. Section 4.3 gives controlled comparisons, and Sec. 4.4 evaluates the list without the frames. †Trained on data generated from the benchmark's source datasets or with its question generator. °Not statistically significant.

	Best published result		Gemini 3.8 Flash, same frames		
Benchmark	Fine-tuned for spatial reasoning	Without spatial training	Video only	+ our list	Gain
VSI-Bench [69]	76.6† ConsiSpace [29]	60.2 TRACE [28] (Gemini-3-Pro)	67.5	**73.6**	6.0
VSI-Bench-Debiased [4]	66.3 GeoThinker [33]	–	65.3	**71.6**	6.3
MMSI-Bench [71]	43.3 SenseNova-SI [6]	55.0 GCA [11] (Gemini-2.5-Pro)	64.1	**72.2**	8.1
SPBench-MV [34]	70.9† SpatialLadder [34]	61.9 GCA [11] (Qwen3-VL)	72.8	**78.0**	5.2
MindCube-Tiny [63]	85.7† SenseNova-SI [6]	64.2 GCA [11] (Qwen3-VL)	92.4	**94.3**	1.9°

Table 2. **Accuracy by task on VSI-Bench.** All 5,130 questions, Gemini 3.8 Flash with the same frames, without and with our list. Tasks that geometry decides gain most. Object size loses because boxes can cover only part of a large object. °Not statistically significant.

Task	Video	+ list	Gain
Room size	54.2	75.2	21.0
Abs. distance	52.2	69.7	17.5
Route plan	79.9	91.8	11.9
Rel. direction	76.4	84.4	8.0
Appear. order	73.0	74.3	1.3°
Counting	56.9	56.0	−0.8°
Rel. distance	71.3	68.2	−3.1°
Object size	76.5	69.1	−7.4
All (task average)	67.5	73.6	6.0

4.2. Results across benchmarks and VLMs

Table 1 shows the results. With Gemini 3.8 Flash and the same frames, our list raises accuracy on VSI-Bench from 67.5% to 73.6%. The gain of 6.3 points on VSI-Bench-Debiased shows that it does not come from questions that can be answered without looking, whereas fine-tuned models lose 6 to 8 points there [6, 33, 72]. On the image benchmarks that we did not use during development, the unchanged pipeline adds 8.1 points on MMSI-Bench and 5.2 on SPBench-MV. On MindCube-Tiny, the images alone already give 92.4%, which leaves little room, and the gain of 1.9 points is not significant. Figure 3 shows examples. With our list, Gemini 3.8 Flash is more accurate than every published result in Tab. 1 but one, with or without spatial training. The exception is ConsiSpace [29] on VSI-Bench, which is fine-tuned on questions generated in the style of VSI-Bench from the same source datasets. These results use other VLMs, so Sec. 4.3 compares representations with the same VLM, and our list is again the most accurate.

The gains follow what the list states explicitly (Tab. 2). Room size gains most, because our list gives the floor area of the room, which our pipeline measures. Absolute distance, route planning and relative direction gain because the list places all objects in one metric frame, so the VLM can compute distances and directions instead of estimating them across views. Appearance order, counting and relative distance do not change significantly. The list does not record when an object is seen, and its potential missed, duplicated or partial boxes can mislead counts and comparisons. Object size gets worse because the box of a large object can cover only the part that was reconstructed and segmented, and the VLM reports the size of the box instead of correcting it from the video, as it often follows the list over the frames (Sec. 5).

The list does not replace the video. Given only the list, without any frame, Gemini 3.8 Flash is about 6 points less accurate on the 800 questions than from the video alone, and 11.8 points less accurate than with both (Tab. 5). The list alone is more accurate than the video alone on tasks that geometry decides, such as absolute distance (60.1% *vs*. 48.8%) and relative direction (83.0% *vs*. 72.0%). It fails on appearance order (39.0% *vs*. 76.0%), because the list does not record when an object is seen, and on counting (32.6% *vs*. 62.3%), where the frames correct objects that the list misses or duplicates. In all other results, the VLM therefore receives the list together with the frames.

Table 3 repeats the comparison with other VLMs. On the 800 questions, our list raises the accuracy of all three Gemini models [13] significantly, by 5.9 to 10.2 points. our list adds about 10 points to GPT-5 [59] and GPT-6 Astra [52]. With the ground-truth list, the Gemini and GPT models reach 85.5% to 95.5%, so better 3D perception would pay off further. The open-weight Qwen3-VL models [1] do not gain from our list. Both gain significantly from the ground-truth list, so they can use the format, but they are not robust to the errors in our list. The list also makes answers cheaper. With it, Gemini 3.8 Flash uses less than half as many thinking tokens per question, so an answer costs about 40% less although the prompt is longer (see supplement).

4.3. Comparison with the same VLM

We compare our list with the representations of five published 3D perception systems, converted into our format and gravity-aligned frame, so that the same VLM answers the

Table 3. **Our list helps every frontier VLM.** Accuracy (%) with the same 32 frames for every VLM. The 800 VSI-Bench questions of Tab. 4, 100 per task. Each gain is over the video alone, computed before rounding. The ground-truth (GT) list is an upper bound. ‡Open-weight models. °Not statistically significant.

VLM	Video	+ our list	Gain	+ GT list	Gain
800 questions, 100 per task					
Gemini 3.5 Flash-Lite	53.6	61.5	7.9	85.5	31.9
Gemini 3.1 Pro	59.4	69.6	10.2	91.4	32.0
Gemini 3.8 Flash	68.6	74.5	5.9	94.8	26.2
GPT-5	51.2	61.4	10.1	87.9	36.7
GPT-6 Astra	65.0	74.7	9.7	95.5	30.4
Qwen3-VL-8B-Thinking‡	51.4	51.9	0.5°	68.2	16.8
Qwen3-VL-32B-Instruct‡	63.2	61.8	−1.4°	70.7	7.5

Table 4. **Comparison with other methods.** Gemini 3.8 Flash with the same frames on 800 VSI-Bench questions, 100 per task. Accuracy (%) and gain over the video alone. The row *ConceptGraphs, our input* applies its association step to our reconstruction and masks. °Not statistically significant.

Method	Accuracy	Gain
Video only	68.6	–
Cognitive map [69]	68.6	0.0°
TRACE [28]	66.9	−1.7°
ConceptGraphs [22]	69.4	0.9°
PanSt3R [79]	70.5	1.9°
3D-MOOD [73]	69.7	1.1°
3D-MOOD, fused across frames	68.4	−0.1°
WildDet3D [30]	71.0	2.5°
SpatialLM [48]	72.1	3.5
ConceptGraphs, our input	73.6	5.0
Our list	**74.5**	**5.9**
Ground-truth list	94.8	26.2

same 800 questions with the same frames (Tab. 4). They are the robot scene graph ConceptGraphs [22], the panoptic reconstruction PanSt3R [79], the single-image open-set 3D detectors 3D-MOOD [73] and WildDet3D [30], whose per-frame boxes we collect into one list, and the layout model SpatialLM [48], which we run on our reconstruction. For 3D-MOOD, we also fuse the boxes of one object across frames. PanSt3R, 3D-MOOD and, on ARKitScenes, ConceptGraphs receive the datasets' ground-truth camera intrinsics, which our pipeline does not use. Where a system gives no list for a scene, the VLM answers from the video alone, and the result is the same on the questions each system covers (see supplement). ConceptGraphs was run for an earlier version of this work, on our reconstruction and masks of that time, and adds only 0.9 points over the video alone. Applied to our final reconstruction and masks, its association step is about as accurate as our full pipeline (73.6% *vs*. 74.5%), so the gain comes mainly from our reconstruction and masks.

The lists of the published systems add at most 3.5 points over the video alone, against 5.9 for ours, and only SpatialLM's gain is significant. Two properties of the lists explain the difference. The first is metric geometry. On absolute distance, our list adds 28.9 points and the other lists 2.8 to 9.8. The second is whether the list contains the objects a question asks about, since a missing object costs the VLM two thirds of what the whole list adds (Sec. 5).

Asking the VLM to recover the layout itself does not help. Drawing a cognitive map first [69] changes accuracy by 0.0 points, and TRACE [28], which has the VLM describe the scene and its camera path, by −1.7, neither significantly. Nor does a list that the VLM writes itself help. Asked to write a list in our format from the same up to 106 frames, once per scene and without any question, GPT-6 Astra [52] lists about 120 objects per scene. It then answers no better than from the video alone and 10.5 points worse than with our list. All three methods ask the VLM to infer the layout from the frames, so they inherit the errors of its perception, whereas our list adds geometry measured by a reconstruction model. The tool-using agent GCA [11] calls reconstruction, detection and code tools for every question. With Gemini 3.8 Flash, and GroundingDINO [39] in place of its own detector, which supports only two specific VLMs, it reaches 62.8% on MMSI-Bench, below the images alone (64.1%) and our list (72.2%).

Table 5. **Ablations on VSI-Bench.** Change in accuracy, in percentage points, when one part of our list is changed, with Gemini 3.8 Flash and the same frames, on the 800 questions of Tabs. 3 and 4. With the full list, accuracy is 74.5%. °Not statistically significant.

Change to our list	Accuracy change
Names only, without positions or sizes	−8.6
Only names that map to VSI-Bench categories	−2.6
Half of the objects removed at random	−3.8
Fixed vocabulary only, no caption names	−0.5°
Stricter detection threshold	−0.5°
Only objects seen in at least three frames	−0.2°
List only, without the frames (text only)	−11.8
No list (video alone)	−5.9

4.4. Ablations

Table 5 removes one part of the list at a time. Names without positions and sizes do not help. They are 8.6 points less accurate than the full list and even below the video alone, so the gain comes from the geometry. Coverage matters as well. Removing half of the objects at random costs 3.8 points. Keeping only the objects whose names map to one of VSI-Bench's categories costs 2.6 points, although the questions ask only about these categories. Our names are open-vocabulary, so an asked-about object can carry a name that is similar to its category but not identical, and the mapping then drops it.

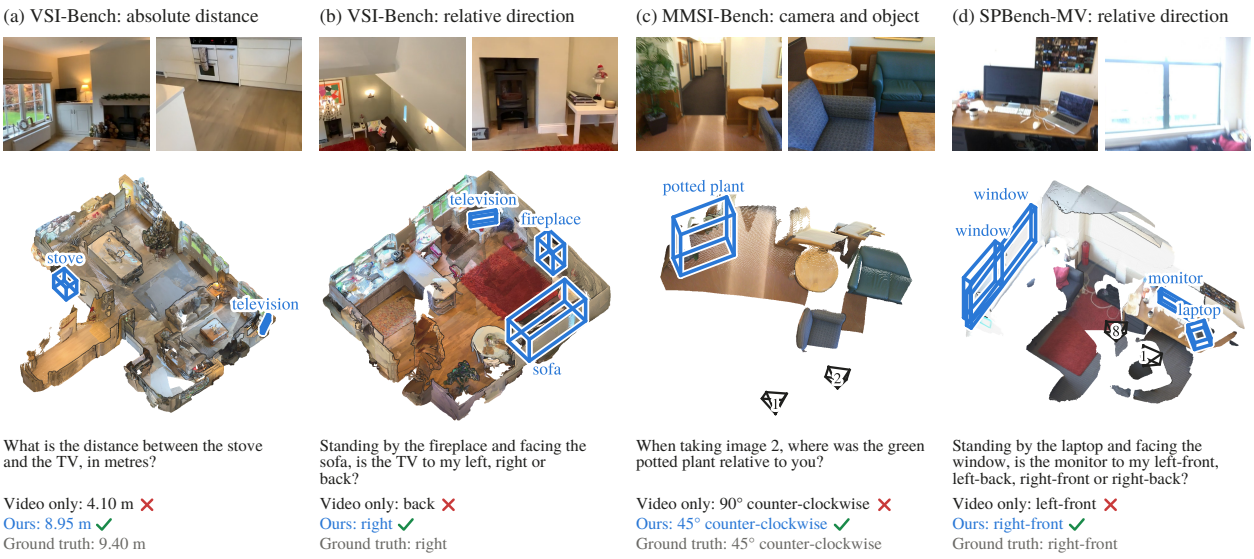

Figure 3. **Examples across benchmarks** (Gemini 3.8 Flash). Each panel shows two input images and our 3D representation visualized on the reconstructed scene. Objects referred to by the question are highlighted in blue; camera poses are also shown for image sets. In these selected examples, the VLM answers incorrectly from the images alone and correctly with our representation. Questions are shortened for display.

Variants that make the list more precise, a stricter detection threshold or only objects seen in at least three frames, change nothing measurable. This agrees with Sec. 5, where extra unrelated objects barely affect the VLM. Our object names come from two sources, a fixed vocabulary and the captions of the frames (Sec. 3). Without the caption names, accuracy on VSI-Bench stays the same, because almost every object category of VSI-Bench is already in the fixed vocabulary. The captions add names for objects outside this vocabulary, which other scenes may contain. The last two rows of Tab. 5, the list without the frames and the frames without the list, are discussed in Sec. 4.2.

The list therefore needs metric geometry and the objects that questions ask about, while a more precise list with fewer objects does not help.

5. What Does a VLM Need from a 3D Representation?

Our representation improves spatial reasoning more than the evaluated 3D perception systems, although all are lists of objects with 3D boxes in the same format. We investigate which errors in these lists affect the VLM's answers. We first introduce controlled errors into ground-truth representations, then examine whether the findings hold for predicted lists.

5.1. Controlled errors in ground-truth lists

We start from VSI-Bench's ground-truth lists, built from the annotation from which its answers were generated. They give each object's name, the position of its centre and its size in metres, and the room's floor area. We change one thing at a time, following error analyses of object detectors [3, 5, 25]. The object a question asks about is removed, duplicated, doubled in size, or shifted by 0.5 m or 1 m, or objects of categories that no question asks about are added to the scene. The supplement defines each change.

Table 6. **What one error costs the VLM and the detection metrics.** Each row changes VSI-Bench's ground-truth list in one way, on the 592 questions that ask about specific objects (572 for unrelated objects). The VLM columns give the accuracy Gemini 3.8 Flash loses against the intact list, over all six tasks, the three numerical tasks (counting, absolute distance, object size) and the three multiple-choice tasks (relative distance, relative direction, route planning). *All* is the mean of *Num.* and *Choice*. *Match* is the share of changed boxes that still count as correct detections at 3D IoU 0.25. The last two columns give the points lost by F1 at 3D IoU 0.25 and by the open detection score [73], both computed over all objects of a scene. °Not statistically significant.

	Lost by the VLM			Match	Score lost	
Change to the GT list	All	Num.	Choice	@.25	F1@.25	ODS
The object the question asks about						
Missing	17.7	29.3	6.2	0	9.1	5.0
Duplicated	6.3	11.6	1.1°	100	6.0	3.0
Size ×2	23.9	45.7	2.1°	0	13.5	2.0
Shifted 1 m	14.5	18.6	10.4	4	12.6	6.2
Shifted 0.5 m	7.6	11.9	3.3	16	9.8	4.0
The rest of the scene						
Unrelated objects added	1.7	1.5	1.9°	–	20.2	10.1
No list	26.2	36.4	15.9	–	100.0	100.0

Gemini 3.8 Flash answers every question with the same 32 frames and the same prompt, which warns that the list may contain errors. We leave out two of VSI-Bench's eight tasks. Room size asks about no object, and appearance order is answered from the video, because the list does not record when an object is seen. This leaves 592 of the 800 questions. For each change, we report how much accuracy the VLM loses compared with the intact ground-truth list.

We reuse the changed lists of a larger study, reported in full in the supplement, answering 38,582 queries with more kinds of errors. Before that study began, we wrote down nine hypotheses, such as that a missing object costs more than a duplicated one, together with how each would be tested.

5.2. Which errors matter?

Missing the object a question asks about is among the costliest errors (Tab. 6). It costs 17.7 points, two thirds of what the whole list adds (26.2), because a question concerns one or two specific objects and without them the list cannot answer it. Adding objects of unrelated categories costs only 1.7 points. The loss from a missing object is largest on numerical questions (29.3 points), where the VLM takes the count, distance or size from the list, and smaller on multiple-choice questions (6.2), where it can fall back on the video.

Geometric errors matter mainly for numerical answers. A box doubled in size costs 45.7 points on numerical questions and nothing significant on multiple-choice ones. A shift by 0.5 m or 1 m costs 7.6 or 14.5 points over the six tasks, more on numerical than on multiple-choice questions.

The VLM often follows a wrong list even though it sees the video. For errors whose effect on the answer has a known direction, such as a doubled box on a size question or a duplicated object on a counting question, it moves its answer in that direction 66.4% of the time. Its answers from the video alone move that way only 21.5% of the time under the same rule. This resembles how VLMs follow text that contradicts the image [15].

Detection metrics weigh these errors differently. We compute the two box metrics used to evaluate SpatialLM [48] and 3D-MOOD [73]. The first is F1 at 3D IoU 0.25, where a box counts as a correct detection if it has the right name and its intersection over union with a ground-truth box is at least 0.25. The second is the open detection score (ODS) [73], which also penalizes errors in centre, size and orientation. We use their F1-based form, because the lists carry no confidence scores. Unlike the VLM's answers, both metrics average over all objects of a scene. Unrelated extra objects cost F1 20.2 points and the VLM only 1.7. A 1 m shift costs F1 more than a missing object (12.6 *vs*. 9.1 points), whereas the VLM loses more from the missing object (17.7 *vs*. 14.5). A doubled box, the costliest error for the VLM, lowers ODS by only 2.0 points.

The same mismatch appears in real lists. ConceptGraphs' list has about the same F1@0.25 as ours (0.33 *vs*. 0.32) and a higher precision (0.50 *vs*. 0.27, see supplement). Yet it contains the asked-about objects far less often, and it adds less to the VLM's accuracy (Tab. 4). Oracle fixes of an earlier version of our lists point the same way. We correct five kinds of errors with ground truth and leave out one correction at a time. Only leaving out the restored missing objects costs significant accuracy (see supplement). Detection scores alone therefore do not tell whether a list helps a VLM, because what matters depends on which objects and measurements a question needs (Tab. 6).

6. Discussion

Because the question is unknown when a 3D representation for a VLM is built, it should first of all contain every object that may be asked about, while extra objects of unrelated categories cost little. Precision still matters for the asked-about objects, since a duplicate misleads counting and a wrong box misleads numerical answers. The ground-truth lists show how much better perception could still help. For most questions, our list still misses at least one of the objects the question asks about.

Our study has limitations. The list describes a static scene, so on MMSI-Bench questions [71] about camera motion, accuracy drops from 83.8% to 71.6% with it. Boxes of large objects can be incomplete, which lowers accuracy on object size. The open-weight VLMs we tested do not gain from our list, although they gain from the ground-truth list. These smaller models may need fine-tuning to cope with the errors in the list, which we leave to future work.

7. Conclusion

VLMs answer spatial questions far more accurately when they are told where the objects are. We build such a list of objects and their metric 3D boxes from images alone, with off-the-shelf models, once per scene and before any question is known. Given to the VLM together with the frames, it improves accuracy on four spatial benchmarks and for all five frontier VLMs we tested, without fine-tuning, and it helps more than the outputs of other 3D perception systems. Our controlled study shows what such a list must get right. A missing asked-about object costs two thirds of what the whole list adds, whereas unrelated extra objects barely matter, although detection metrics count them as errors. Because VLMs also follow lists that are wrong, 3D perception for VLMs should be judged by the questions it lets them answer, and not by detection metrics alone.

Acknowledgments

This work was supported by the Czech Science Foundation (GACR) EXPRO (grant no. 23-07973X), the Ministry of Education, Youth and Sports of the Czech Republic

through the e-INFRA CZ (ID:90254), the Grant Agency of the Czech Technical University in Prague, grant No. SGS26/098/OHK3/1T/13. Yutong Chen is funded by a Google PhD Fellowship. The author(s) used generative AI to assist with editing the manuscript, cleaning up figures and coding. All was reviewed by the authors, who take full responsibility for its content.

References

[1] Shuai Bai et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025. 4, 5, 15

[2] Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer, Brandon Joffe, Daniel Kurz, Arik Schwartz, and Elad Shulman. ARKitScenes: A diverse real-world dataset for 3D indoor scene understanding using mobile RGB-D data. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2021. 4, 13, 14

[3] Daniel Bolya, Sean Foley, James Hays, and Judy Hoffman. TIDE: A general toolbox for identifying object detection errors. In *Computer Vision – ECCV 2020*, pages 558–573. Springer, 2020. 3, 7

[4] Ellis Brown, Jihan Yang, Shusheng Yang, Rob Fergus, and Saining Xie. Benchmark designers should "train on the test set" to expose exploitable non-visual shortcuts. In *Conference on Language Modeling (COLM)*, 2026. 2, 4, 5, 16

[5] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11621–11631, 2020. 7

[6] Zhongang Cai, Ruisi Wang, Chenyang Gu, Fanyi Pu, Junxiang Xu, Yubo Wang, Wanqi Yin, Zhitao Yang, Chen Wei, Tongxi Zhou, Qingping Sun, Hui En Pang, Jiaqi Li, Oscar Qian, Zhiqian Lin, Xuanke Shi, Kewang Deng, Xiaoyang Han, Zukai Chen, Xiangyu Fan, Hanming Deng, Lewei Lu, Liang Pan, Bo Li, Ziwei Liu, Quan Wang, Dahua Lin, and Lei Yang. Scaling spatial intelligence with multimodal foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7879–7890, 2026. 2, 5

[7] Yang Cao, Feize Wu, Dave Zhenyu Chen, Yingji Zhong, Lanqing Hong, and Dan Xu. VGGT-Det: Mining VGGT internal priors for sensor-geometry-free multi-view indoor 3D object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4708–4717, 2026. 3

[8] Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris Coll-Vinent, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. SAM 3: Segment anything with concepts. In *International Conference on Learning Representations (ICLR)*, 2026. 2, 3, 4, 14

[9] Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. SpatialVLM: Endowing vision-language models with spatial reasoning capabilities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14455–14465, 2024. 2

[10] Siyi Chen, Mikaela Angelina Uy, Chan Hee Song, Faisal Ladhak, Adithyavairavan Murali, Qing Qu, Stan Birchfield, Valts Blukis, and Jonathan Tremblay. SpaceTools: Tool-augmented spatial reasoning via double interactive RL. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 37109–37120, 2026. 2

[11] Zeren Chen, Xiaoya Lu, Zhijie Zheng, Pengrui Li, Lehan He, Yijin Zhou, Jing Shao, Bohan Zhuang, and Lu Sheng. Geometrically-constrained agent for spatial reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 38689–38699, 2026. 3, 5, 6, 16

[12] 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 (NeurIPS)*, 2024. 2

[13] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025. 4, 5, 15

[14] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 4, 14

[15] Ailin Deng, Tri Cao, Zhirui Chen, and Bryan Hooi. Words or vision: Do vision-language models have blind faith in text? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3867–3876, 2025. 3, 8

[16] Junqing Du, Fernando Ropero, Erkin Turkoz, Yanfeng Zhang, and Lu Liu. GraFT: A training-free framework for spatial reasoning in multimodal large language models via 3D scene graphs. *arXiv preprint arXiv:2609.03892*, 2026. 3

[17] Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, 1993. 4, 13

[18] Zhiwen Fan, Jian Zhang, Renjie Li, Junge Zhang, Runjin Chen, Hezhen Hu, Kevin Wang, Peihao Wang, Huaizhi Qu, Shijie Zhou, Dilin Wang, Zhicheng Yan, Hongyu Xu, Justin Theiss, Tianlong Chen, Jiachen Li, Zhengzhong Tu, Zhangyang Wang, and Rakesh Ranjan. VLM-3R: Vision-language models augmented with instruction-aligned 3D reconstruction. In *Proceedings of the IEEE/CVF Conference*

on Computer Vision and Pattern Recognition (CVPR), pages 31054–31065, 2026. 2

[19] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 13, 14

[20] Mohsen Gholami, Ahmad Rezaei, Zhou Weimin, Sitong Mao, Shunbo Zhou, Yong Zhang, and Mohammad Akbari. Spatial reasoning with vision-language models in ego-centric multi-view scenes. In *International Conference on Learning Representations (ICLR)*, 2026. 3

[21] Ziren Gong, Xiaohan Li, Fabio Tosi, Jiawei Han, Stefano Mattoccia, Jianfei Cai, and Matteo Poggi. Ov3R: Open-vocabulary semantic 3D reconstruction from RGB videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 34206–34216, 2026. 3

[22] Qiao Gu, Ali Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, Chuang Gan, Celso Miguel de Melo, Joshua B. Tenenbaum, Antonio Torralba, Florian Shkurti, and Liam Paull. ConceptGraphs: Open-vocabulary 3D scene graphs for perception and planning. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 5021–5028, 2024. 1, 2, 3, 6, 16, 18

[23] Yejie Guo, Yunzhong Hou, Wufei Ma, Meng Tang, and Ming-Hsuan Yang. Pursuing minimal sufficiency in spatial reasoning. In *International Conference on Learning Representations (ICLR)*, 2026. 3

[24] Tanmay Gupta and Aniruddha Kembhavi. Visual programming: Compositional visual reasoning without training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14953–14962, 2023. 3

[25] Derek Hoiem, Yodsawalai Chodpathumwan, and Qieyun Dai. Diagnosing error in object detectors. In *Computer Vision – ECCV 2012*, pages 340–353. Springer, 2012. 3, 7

[26] Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979. 13

[27] Peng-Hao Hsu, Ke Zhang, Fu-En Wang, Tao Tu, Ming-Feng Li, Yu-Lun Liu, Albert Y. C. Chen, Min Sun, and Cheng-Hao Kuo. OpenM3D: Open vocabulary multi-view indoor 3D object detection without human annotations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8688–8698, 2025. 3

[28] Jiacheng Hua, Yishu Yin, Yuhang Wu, Tai Wang, Yifei Huang, and Miao Liu. Unleashing spatial reasoning in multimodal large language models via textual representation guided reasoning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 13616–13637, 2026. 1, 2, 3, 5, 6, 16

[29] Ting Huang, Zhenyu Zhang, Wenyuan Huang, Jian Yang, and Hao Tang. ConsiSpace: Learning geometric consistency matters for video spatial reasoning. In *European Conference on Computer Vision (ECCV)*, 2026. 2, 5

[30] Weikai Huang, Jieyu Zhang, Sijun Li, Taoyang Jia, Jiafei Duan, Yunqian Cheng, Jaemin Cho, Matthew Wallingford, Rustin Soraki, Chris Dongjoo Kim, Shuo Liu, Donovan Clay, Taira Anderson, Winson Han, Ali Farhadi, Bharath Hariharan, Zhongzheng Ren, and Ranjay Krishna. WildDet3D: Scaling promptable 3D detection in the wild. *arXiv preprint arXiv:2604.08626*, 2026. 1, 2, 6, 15, 16

[31] Harold W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1–2):83–97, 1955. 13

[32] Dingming Li, Hongxing Li, Zixuan Wang, Yuchen Yan, Hang Zhang, Siqi Chen, Guiyang Hou, Shengpei Jiang, Wenqi Zhang, Yongliang Shen, Weiming Lu, and Yueting Zhuang. ViewSpatial-Bench: Evaluating multi-perspective spatial localization in vision-language models. In *European Conference on Computer Vision (ECCV)*, 2026. 2

[33] Haoyuan Li, Qihang Cao, Tao Tang, Kun Xiang, Zihan Guo, Jianhua Han, Hang Xu, and Xiaodan Liang. Thinking with geometry: Active geometry integration for spatial reasoning. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2026. 2, 5

[34] Hongxing Li, Dingming Li, Zixuan Wang, Yuchen Yan, Hang Wu, Wenqi Zhang, Yongliang Shen, Weiming Lu, Jun Xiao, and Yueting Zhuang. SpatialLadder: Progressive training for spatial reasoning in vision-language models. In *International Conference on Learning Representations (ICLR)*, 2026. 2, 4, 5, 16

[35] Pengteng Li, Pinhao Song, Wuyang Li, Huizai Yao, Weiyu Guo, Yijie Xu, Dugang Liu, and Hui Xiong. See&Trek: Training-free spatial prompting for multimodal large language model. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. 3

[36] Zongzhao Li, Zongyang Ma, Mingze Li, Songyou Li, Yu Rong, Tingyang Xu, Ziqi Zhang, Deli Zhao, and Wenbing Huang. STAR-R1: Multi-view spatial transformation reasoning by reinforcing multimodal LLMs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12041–12051, 2026. 2

[37] Benlin Liu, Yuhao Dong, Yiqin Wang, Zixian Ma, Yansong Tang, Luming Tang, Yongming Rao, Wei-Chiu Ma, and Ranjay Krishna. Coarse correspondences boost spatial-temporal reasoning in multimodal language model. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 3783–3792, 2025. 3

[38] Fangfu Liu, Diankun Wu, Jiawei Chi, Yimo Cai, Yi-Hsin Hung, Xumin Yu, Hao Li, Han Hu, Yongming Rao, and Yueqi Duan. Spatial-TTT: Streaming visual-based spatial intelligence with test-time training. In *European Conference on Computer Vision (ECCV)*, 2026. 2

[39] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding DINO: Marrying DINO with grounded pre-training for open-set object detection. In *Computer Vision – ECCV 2024*, pages 38–55. Springer, 2024. 6

[40] Yifan Liu, Fangneng Zhan, Kaichen Zhou, Yilun Du, Paul Pu Liang, and Hanspeter Pfister. Abstract 3D perception for spatial intelligence in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. 3

[41] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10012–10022, 2021. 15

[42] Zuntao Liu, Yi Du, Taimeng Fu, Shaoshu Su, Cherie Ho, and Chen Wang. Vision-language memory for spatial reasoning. In *European Conference on Computer Vision (ECCV)*, 2026. 2

[43] Zhanpeng Luo, Ce Zhang, Silong Yong, Cunxi Dai, Qianwei Wang, Haoxi Ran, Guanya Shi, Katia P. Sycara, and Yaqi Xie. pySpatial: Generating 3D visual programs for zero-shot spatial reasoning. In *International Conference on Learning Representations (ICLR)*, 2026. 3

[44] Chenyang Ma, Kai Lu, Ta-Ying Cheng, Niki Trigoni, and Andrew Markham. SpatialPIN: Enhancing spatial reasoning capabilities of vision-language models through prompting and interacting 3D priors. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 3

[45] 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 (ICLR)*, 2023. 2

[46] Xianzheng Ma, Tao Sun, Shuai Chen, Yash Sanjay Bhalgat, Jindong Gu, Angel X. Chang, Iro Armeni, Iro Laina, Songyou Peng, and Victor Adrian Prisacariu. Do 3D large language models really understand 3D spatial relationships? In *International Conference on Learning Representations (ICLR)*, 2026. 2

[47] 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. 3

[48] Yongsen Mao, Junhao Zhong, Chuan Fang, Jia Zheng, Rui Tang, Hao Zhu, Ping Tan, and Zihan Zhou. SpatialLM: Training large language models for structured indoor modeling. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. 1, 2, 3, 6, 8, 15, 16, 18

[49] Damiano Marsili, Rohun Agrawal, Yisong Yue, and Georgia Gkioxari. Visual agentic AI for spatial reasoning with a dynamic API. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 19446–19455, 2025. 3

[50] George A. Miller. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41, 1995. 13

[51] Phuc Nguyen, Tuan Duc Ngo, Evangelos Kalogerakis, Chuang Gan, Anh Tran, Cuong Pham, and Khoi Nguyen. Open3DIS: Open-vocabulary 3D instance segmentation with 2D mask guidance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4018–4028, 2024. 3

[52] OpenAI. GPT-6 Astra system card. https://deploymentsafety.openai.com/gpt-6-astra, 2026. 2, 4, 5, 6, 15, 16

[53] Jonah Philion, Amlan Kar, and Sanja Fidler. Learning to evaluate perception models using planner-centric metrics. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14055–14064, 2020. 3

[54] Sundar Pichai, Demis Hassabis, and Koray Kavukcuoglu. A new era of intelligence with Gemini 3. https://blog.google/products-and-platforms/products/gemini/gemini-3/, 2025. 4, 13, 14, 15, 16

[55] Zhangyang Qi, Zhixiong Zhang, Ye Fang, Jiaqi Wang, and Hengshuang Zhao. GPT4Scene: Understand 3D scenes from videos with vision-language models. In *International Conference on Learning Representations (ICLR)*, 2026. 3

[56] Krishan Rana, Jesse Haviland, Sourav Garg, Jad Abou-Chakra, Ian Reid, and Niko Suenderhauf. SayPlan: Grounding large language models using 3D scene graphs for scalable robot task planning. In *Proceedings of The 7th Conference on Robot Learning (CoRL)*, pages 23–72. PMLR, 2023. 3

[57] David Rozenberszki, Or Litany, and Angela Dai. Language-grounded indoor 3D semantic segmentation in the wild. In *European Conference on Computer Vision (ECCV)*, 2022. 15

[58] Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H. Chi, Nathanael Schärli, and Denny Zhou. Large language models can be easily distracted by irrelevant context. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, pages 31210–31227. PMLR, 2023. 3

[59] Aaditya Singh et al. OpenAI GPT-5 system card. *arXiv preprint arXiv:2601.03267*, 2025. 4, 5, 15

[60] Dídac Surís, Sachit Menon, and Carl Vondrick. ViperGPT: Visual inference via python execution for reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11888–11898, 2023. 3

[61] Ayça Takmaz, Elisabetta Fedele, Robert W. Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. OpenMask3D: Open-vocabulary 3D instance segmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 3

[62] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 5294–5306, 2025. 3

[63] Qineng Wang, Baiqiao Yin, Pingyue Zhang, Jianshu Zhang, Kangrui Wang, Zihan Wang, Jieyu Zhang, Keshigeyan Chandrasegaran, Han Liu, Ranjay Krishna, Saining Xie, Jiajun Wu, Li Fei-Fei, and Manling Li. Spatial mental modeling from limited views. In *International Conference on Learning Representations (ICLR)*, 2026. 2, 4, 5, 16

[64] Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. In *International Conference on Learning Representations (ICLR)*, 2026. 2, 3, 4, 14

[65] Abdelrhman Werby, Chenguang Huang, Martin Büchner, Abhinav Valada, and Wolfram Burgard. Hierarchical open-vocabulary 3D scene graphs for language-grounded robot navigation. In *Proceedings of Robotics: Science and Systems (RSS)*, 2024. 3

[66] Diankun Wu, Fangfu Liu, Yi-Hsin Hung, and Yueqi Duan. Spatial-MLLM: Boosting MLLM capabilities in visual-based spatial intelligence. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. 2

[67] Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a variety of vision tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4818–4829, 2024. 3, 13, 14

[68] Mi Yan, Jiazhao Zhang, Yan Zhu, and He Wang. MaskClustering: View consensus based mask graph clustering for open-vocabulary 3D instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 28274–28284, 2024. 3

[69] 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 Computer Vision and Pattern Recognition Conference (CVPR)*, pages 10632–10643, 2025. 1, 2, 3, 4, 5, 6, 14, 16

[70] Jihan Yang, Zifan Zhao, Xichen Pan, Shusheng Yang, Junyi Zhang, Hu Xu, Shang-Wen Li, and Saining Xie. Cambrian-P: Pose-grounded video understanding. In *European Conference on Computer Vision (ECCV)*, 2026. 2

[71] Sihan Yang, Runsen Xu, Yiman Xie, Sizhe Yang, Mo Li, Jingli Lin, Chenming Zhu, Xiaochen Chen, Haodong Duan, Xiangyu Yue, Dahua Lin, Tai Wang, and Jiangmiao Pang. MMSI-Bench: A benchmark for multi-image spatial intelligence. In *International Conference on Learning Representations (ICLR)*, 2026. 2, 4, 5, 8, 16

[72] Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis Brown, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhan Wang, Rob Fergus, Yann LeCun, Li Fei-Fei, and Saining Xie. Cambrian-S: Towards spatial supersensing in video. In *International Conference on Learning Representations (ICLR)*, 2026. 2, 5

[73] Yung-Hsu Yang, Luigi Piccinelli, Mattia Segu, Siyuan Li, Rui Huang, Yuqian Fu, Marc Pollefeys, Hermann Blum, and Zuria Bauer. 3D-MOOD: Lifting 2D to 3D for monocular open-set object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7429–7439, 2025. 1, 2, 3, 6, 7, 8, 15, 16, 17, 18

[74] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. ScanNet++: A high-fidelity dataset of 3D indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. 4, 13

[75] Jiangye Yuan, Gowri Kumar, and Baoyuan Wang. Boosting MLLM spatial reasoning with geometrically referenced 3D scene representations. *arXiv preprint arXiv:2603.08592*, 2026. 3

[76] Hanxue Zhang, Haoran Jiang, Qingsong Yao, Yanan Sun, Renrui Zhang, Hao Zhao, Hongyang Li, Hongzi Zhu, and Zetong Yang. Detect anything 3D in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5048–5059, 2025. 3

[77] Jiahui Zhang, Yurui Chen, Yanpeng Zhou, Yueming Xu, Ze Huang, Jilin Mei, Junhui Chen, Yu-Jie Yuan, Xinyue Cai, Guowei Huang, Xingyue Quan, Hang Xu, and Li Zhang. From flatland to space: Teaching vision-language models to perceive and reason in 3D. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2025. 2

[78] Fangrui Zhu, Hanhui Wang, Yiming Xie, Jing Gu, Tianye Ding, Jianwei Yang, and Huaizu Jiang. Struct2D: A perception-guided framework for spatial reasoning in MLLMs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. 3

[79] Lojze Zust, Yohann Cabon, Juliette Marrie, Leonid Antsfeld, Boris Chidlovskii, Jerome Revaud, and Gabriela Csurka. PanSt3R: Multi-view consistent panoptic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5856–5866, 2025. 1, 2, 3, 6, 15, 16, 18

Structured 3D Scene Representations Improve VLM Spatial Reasoning

Supplementary Material

This supplement gives the details behind the paper: how we build the list (Appendix A), how we evaluate (Appendix B), the paper's tables with 95% intervals (Appendix C), how we ran the other methods (Appendix D), and the controlled error study with tests on real lists (Appendix E). Section, table and figure numbers without a letter or an S refer to the paper.

A. Building the list

Table S1 lists every setting of the pipeline.

The fixed vocabulary of Sec. 3.2 is: chair, armchair, office chair, stool, bench, sofa, ottoman, table, desk, coffee table, side table, nightstand, dresser, cabinet, cupboard, wardrobe, shelf, bookshelf, drawer, tv stand, bed, crib, mattress, headboard, television, monitor, computer, laptop, keyboard, computer mouse, computer tower, printer, speaker, telephone, tablet, projector, router, fan, heater, radiator, air conditioner, refrigerator, microwave, oven, stove, range hood, dishwasher, washing machine, dryer, kettle, toaster, coffee maker, blender, vacuum cleaner, lamp, floor lamp, table lamp, ceiling light, chandelier, light switch, power outlet, power strip, smoke detector, toilet, sink, bathtub, shower, mirror, towel, toilet paper, soap dispenser, bath mat, trash can, laundry basket, counter, kitchen island, cutting board, pot, pan, plate, bowl, cup, mug, bottle, jar, dish rack, paper towel, pillow, cushion, blanket, curtain, blinds, rug, clothes, jacket, hat, box, basket, bin, bag, backpack, suitcase, crate, container, shoe rack, coat rack, coat hanger, picture, painting, poster, clock, plant, potted plant, vase, candle, calendar, whiteboard, bulletin board, sign, door, window, book, magazine, paper, notebook, toy, ball, guitar, piano, umbrella, shoes, boots, helmet, bicycle, ladder, toolbox, fire extinguisher, tray, stand, mat, treadmill, exercise ball, easel, file cabinet, mailbox, cart, bucket, broom, iron, ironing board, sewing machine, globe, aquarium, pet bed, stroller, car seat, baby gate. For each scene, we add the 60 object names that occur most often in its Florence-2 [67] captions. A name is the main noun of an object phrase in a caption, or its last two words if they form a name of the vocabulary. Words for structure, such as wall, floor or room, are skipped.

When masks overlap, each pixel goes to the smallest mask, so that a large mask in the background does not take over the objects in front of it. Two masks of the same frame are merged only if they touch in 3D. For the merge rules of Sec. 3.2, two names count as the same if they have the same base form or the same main noun, such as office chair and chair; we use no synonym list. After the merging, small fragments inside an object join it, and so do overlapping objects with the same name. No step uses appearance features or the VLM.

Boxes that end just above the floor are extended down to it, because the bottom of an object is usually hidden.

We developed the pipeline on VSI-Bench, so we also measured the gain on the 220 scenes that played no part in any decision. The gain on these 220 scenes is 5.5 points, against 6.0 on all scenes.

Building a list takes about three minutes on one A100 GPU per VSI-Bench video, plus CPU time. Answers with our list are also cheaper (Sec. 4.2). Compared with the video alone, Gemini 3.8 Flash [54] thinks for 8.1k instead of 17.5k tokens per question. The list adds 7.7k input tokens, but an answer still costs about 40% less, \$0.0145 instead of \$0.0254 at batch prices, over 1,585 VSI-Bench questions.

B. Evaluation

Table S2 gives the settings of every VLM. The 800 VSI-Bench questions (100 per task) were drawn at random from 1,587 questions that we sampled beforehand. None of them comes from the ten scenes we used to write the prompt.

For the intervals of Sec. 4.1, the bootstrap [17] resamples scenes within each source dataset of VSI-Bench, 10,000 times (2,000 times for Tab. 1). Each condition was run once, and even at temperature 0 an answer can change between runs. The intervals therefore show the effect of which questions were sampled, not of this run-to-run change. Each interval is for its own result alone. Only the nine hypotheses of the error study were tested together, with Holm's method [26]: it makes each test stricter, so that the chance of calling any of the nine supported by mistake stays below 5%.

Tables S10 and S11 compare a list with the ground-truth list of its scene. Both lists are gravity-aligned, so they differ only by a rotation about the vertical axis, a horizontal shift and a scale. We fit these with RANSAC [19] on pairs of objects with matching names (thresholds in Tab. S1). A name matches a ground-truth (GT) category if the two are equal after normalisation, if the category is the last word or words of the name, or if their most common WordNet [50] senses agree. Within a category, we pair list and ground-truth objects with the Hungarian method [31] on centre distance. The fitted scale also measures the scale error of our reconstruction. Its median is below 6% on each source dataset of VSI-Bench: ScanNet, ARKitScenes [2] and ScanNet++ [74].

C. The paper's tables with intervals

Tables S3 to S6 repeat Tables 1, 2, 4, 5 and 6 of the paper with 95% intervals. In Tabs. S3 and S4, bold marks the better result of each comparison.

Some methods give no list for some scenes. There the VLM answers from the video alone (Sec. 4.3). A method that ran but found no objects gives an empty list instead.

In Tab. S5, *names only* also drops the floor area, and its prompt says that the list has no positions or sizes. Names

Setting	Value
Geometry (Pi3X [64])	
Input frames	two sharpest per second, at most 106
Pixel confidence	$>0.6^{\dagger}$, not on a depth edge
Relative rule	also $\geq$ the median, if $<1/2$ of pixels pass
Later stages	pixel confidence >0.5
Names and segmentation	
Fixed vocabulary	157 categories
Caption names	60 most frequent (Florence-2 [67])
SAM 3 [8] mask score	≥ 0.5
NMS across names	mask IoU ≥ 0.7
Boxes and floor	
Box fit	gravity-aligned, 0.5% of the extent cut per side
Floor snap	bottom <0.3 m above the floor
Floor area	5 cm voxels, 10 cm grid, largest region, holes filled

Setting	Value
Lifting and association	
Depth outliers	>3 MAD from the mask's median depth
Pixels per mask	≥ 50
Voxel size	5 cm
Link	$\geq 1/4$ of the smaller voxel set, centres ≤ 1.5 m apart
Same-name merge	box overlap $\geq 1/2$
Object kept	≥ 100 points in ≥ 2 frames, or ≥ 300 points
Matching to ground truth (analyses only, Appendix B)	
Alignment	RANSAC [19] on objects with matching names
Inliers	within 0.5 m; ≥ 4 from ≥ 3 categories, or ≥ 8 from 2
Scale	between 0.5 and 2
Object found	centres ≤ 0.5 m apart, or centre inside the box

Table S1. **Every setting of the pipeline** and of the matching to ground truth. $^{\dagger}0.4$ on the ARKitScenes [2] videos of VSI-Bench [69].

```
### SYSTEM INSTRUCTION
You are an expert in 3D Spatial Intelligence.
[32 frames]
Answer each question by correlating the **video walkthrough** (the frames above, in temporal order) with the
**3D object list** below.

COORDINATE SYSTEM: metres, gravity-aligned. z is up and the floor is at z = 0; x and y are horizontal.
Each object has: label; center [x, y, z]; size [length, width, height] (length and width are horizontal, height is
vertical).
room_footprint_m2 = floor area covered by the reconstruction of the scene.
Distances between objects are measured between their closest points (box surfaces), not between centers.
The list is machine-generated: it can miss objects, contain extra ones, or split one object into several entries.
Verify with the frames.

### SCENE METADATA (JSON)
{"room_footprint_m2":18.92,"objects":[
{"label":"air conditioner","center":[2.17,-1.63,1.36],"size":[0.44,0.12,0.54]},
{"label":"armrest","center":[-1.17,0.64,0.31],"size":[0.72,0.21,0.61]},
...
{"label":"lamp","center":[1.46,1.97,1.15],"size":[0.43,0.16,0.65]},
...
{"label":"window","center":[2.2,-0.86,1.55],"size":[1.15,0.21,1.19]},
{"label":"window","center":[-1.75,-0.17,1.54],"size":[0.21,0.02,0.31]}]}

### TASKS
Answer each question below. For each question, output your answer in <answer>...</answer> tags.
Use exactly one <answer> block per question, in the same order as the questions.
For multiple-choice questions, answer with the option letter only.
For numeric questions, answer with a single number only (no units, no words).

### QUESTION 1
Measuring from the closest point of each object, what is the distance between the lamp and the window (in meters)?

### OUTPUT
CRITICAL: Provide exactly 1 <answer>...</answer> blocks in order: first <answer> for QUESTION 1,
second for QUESTION 2, etc. Do not reorder.
```

Figure S1. **The prompt with our list** for the question of Fig. 1 (ScanNet [14] scene0616_01), shortened. The list is sorted by label and sent as one line; only part of it is shown. The output instructions are generic; every request holds one question. With this prompt, Gemini 3.8 Flash [54] answers 2.11 m; the ground truth is 2.2 m.

are mapped to VSI-Bench categories with the rule of Appendix B. The *stricter detection threshold* drops the quarter of the masks with the lowest SAM 3 scores.

D. Compared methods

Each method's list keeps its own metric scale when we place it in our frame (Sec. 4.3). For methods that output points, our box fitter fits a box to each of their objects. The VLM sees each method's own names, with our prompt and frames. Figure S2 shows the lists of all methods for one question.

VLM	Model	Settings
Gemini 3.5 Flash-Lite [13]	`gemini-3.5-flash-lite`	temperature 0, top-p 1, thinking level high, default output limit
Gemini 3.1 Pro [13]	`gemini-3.1-pro-preview`	temperature 0, top-p 1, thinking level high, default output limit
Gemini 3.8 Flash [54]	`gemini-3.8-flash`	temperature 0, top-p 1, thinking level high, default output limit
GPT-5 [59]	`gpt-5-2025-08-07`	reasoning effort medium, at most 32,000 output tokens
GPT-6 Astra [52]	`gpt-6-astra`	reasoning effort medium, at most 32,000 output tokens
Qwen3-VL-8B-Thinking [1]	released weights, bf16	its own sampling (temperature 1.0, top-p 0.95, top-k 20), at most 8,192 new tokens
Qwen3-VL-32B-Instruct [1]	released weights, bf16	greedy decoding, at most 64 new tokens, no thinking

Table S2. **VLM settings.** Every VLM receives 32 frames at native resolution, evenly spaced over a video, or all images of an image set, and answers one question per request. Qwen3-VL-8B-Thinking draws one seeded sample; its answer is read after its reasoning, and a request without an answer counts as wrong.

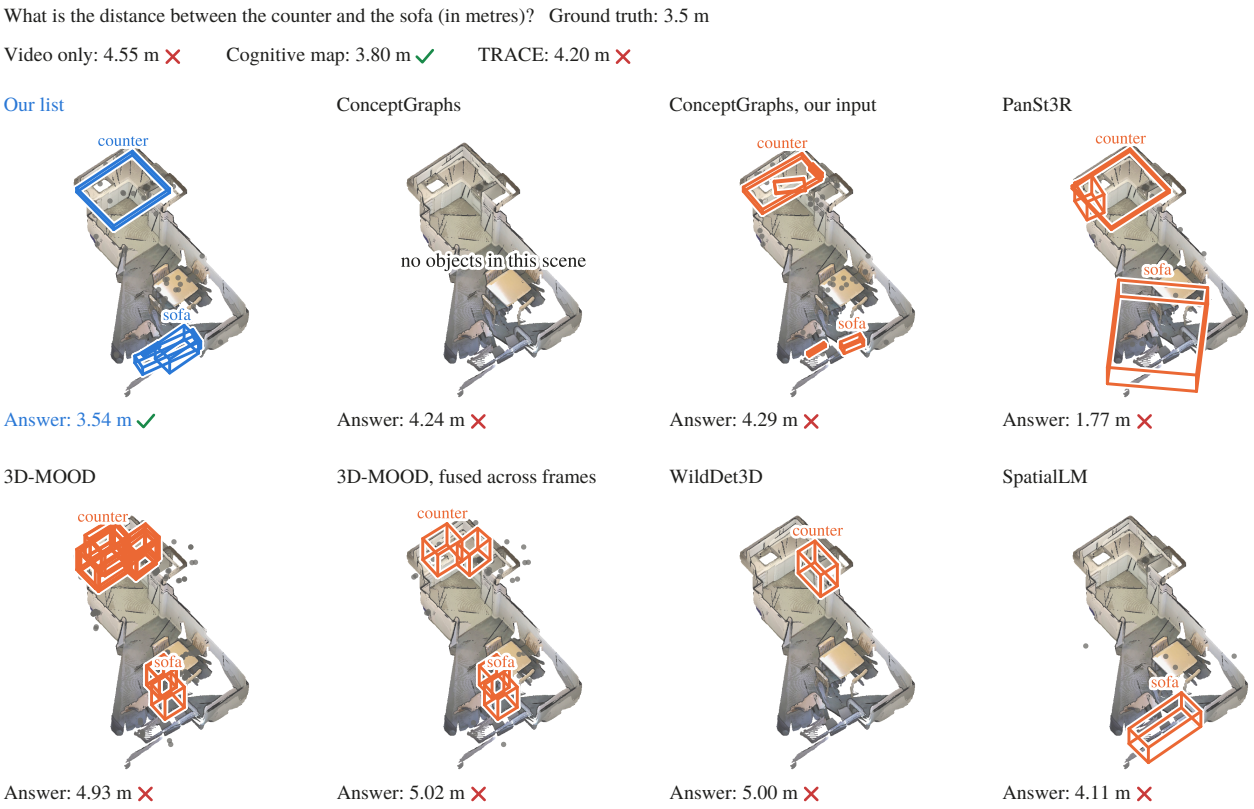

Figure S2. **The lists of all compared methods for one question.** ScanNet scene0651_02, Gemini 3.8 Flash with the same 32 frames. Each list is drawn on our reconstruction: entries named counter or sofa as boxes, all other entries as grey dots. ConceptGraphs ran on this scene but found no objects, so its list is empty. A check marks an answer within 10% of the ground truth. We chose this example for legibility among the questions where every method ran.

PanSt3R [79] ran with its released weights on our frames. We keep its points and panoptic segments, except walls, floor and ceiling.

3D-MOOD [73] ran with its Swin-B [41] checkpoint on the same frames, with the ScanNet200 [57] class names as prompts, and we keep detections with a score of at least 0.3. We place its boxes from each frame with Pi3X's camera poses. The fused variant merges overlapping boxes with the same name (3D IoU of at least 0.25, or one centre inside the other box), which leaves about half of the boxes.

WildDet3D [30] ran with its released code and checkpoint on the same frames. Its boxes are merged across frames with the rule of the fused 3D-MOOD variant. SpatialLM [48] (SpatialLM1.1-Qwen-0.5B) ran with its repository's settings on our Pi3X point cloud, in our frame and at our scale.

The cognitive map uses the prompt of Thinking in

```
The images above are frames from one video walkthrough of an indoor scene, in temporal order.
### TASK
Build a structured 3D representation of this scene. It will later be used to answer spatial
questions about the scene, which are not known yet, so include every object that someone might
ask about: furniture, appliances, fixtures, decorations and small objects. Give one entry per
physical object, counting each object once even if it appears in many frames.
Use one metric coordinate frame for the whole scene, in metres: z points up, the floor is at
z = 0, and x and y are horizontal. For each object give a short common-noun label, the center
[x, y, z] of its 3D bounding box and its size [length, width, height] along the box's own axes.
Also give the floor area of the room in square metres as room_footprint_m2.
### OUTPUT
Output only the JSON, with no other text, in exactly this format:
{"room_footprint_m2": <number>, "objects": [{"label": "<label>", "center": [x, y, z],
 "size": [l, w, h]}, ...]}
```

Figure S3. **The request with which GPT-6 Astra builds its own list** (Sec. 4.3 of the paper). It is sent once per scene after the frames, without any question.

	Video	+ our list	Gain [95% interval]
Table 1: benchmarks			
VSI-Bench [69]	67.5	**73.6**	6.0 [4.2, 7.9]
VSI-Bench-Debiased [4]	65.3	**71.6**	6.3 [4.0, 8.8]
MMSI-Bench [71]	64.1	**72.2**	8.1 [5.0, 11.4]
SPBench-MV [34]	72.8	**78.0**	5.2 [0.2, 9.8]
MindCube-Tiny [63]	92.4	**94.3**	1.9 [−0.1, 3.9]
Table 2: VSI-Bench by task			
Room size	54.2	**75.2**	21.0 [16.7, 25.4]
Abs. distance	52.2	**69.7**	17.5 [14.0, 21.0]
Route plan	79.9	**91.8**	11.9 [6.2, 18.0]
Rel. direction	76.4	**84.4**	8.0 [3.2, 12.9]
Appear. order	73.0	**74.3**	1.3 [−2.7, 5.2]
Counting	**56.9**	56.0	−0.8 [−4.6, 2.8]
Rel. distance	**71.3**	68.2	−3.1 [−9.2, 3.0]
Object size	**76.5**	69.1	−7.4 [−9.7, −5.0]

Table S3. **Tables 1 and 2 of the paper with intervals.** Gemini 3.8 Flash with the same frames: accuracy (%) and gain in points with its 95% interval. Bold: the higher accuracy.

Method	Gain over video	Our lead	Abs. dist.
Cognitive map [69]	0.0 [−2.3, 2.4]	5.9 [2.8, 9.0]	6.7
TRACE [28]	−1.7 [−3.9, 0.5]	7.6 [4.4, 10.8]	1.9
ConceptGraphs [22]	0.9 [−1.5, 3.3]	5.1 [2.0, 8.2]	3.2
PanSt3R [79]	1.9 [−0.7, 4.6]	4.0 [1.0, 7.0]	2.8
3D-MOOD [73]	1.1 [−1.5, 3.7]	4.8 [1.6, 8.0]	9.3
3D-MOOD, fused across frames	−0.1 [−2.7, 2.4]	6.1 [2.9, 9.4]	7.9
WildDet3D [30]	2.5 [−0.0, 5.0]	3.5 [0.5, 6.5]	9.0
SpatialLM [48]	3.5 [0.8, 6.2]	2.4 [−0.9, 5.7]	9.8
ConceptGraphs, our input	5.0 [1.9, 8.0]	0.9 [−1.4, 3.2]	24.4
Our list	**5.9** [3.0, 9.0]	–	**28.9**
Ground-truth list	26.2 [23.3, 29.3]	−20.3 [−23.3, −17.4]	49.7

Table S4. **Table 4 of the paper with intervals.** Gemini 3.8 Flash on the 800 VSI-Bench questions, in points, with 95% intervals. *Our lead*: accuracy with our list minus accuracy with the method; it is negative where the method is ahead. *Abs. dist.*: gain over the video on the 100 absolute-distance questions. Bold: best, not counting the ground-truth list.

Space [69] for the objects named in the question: the VLM draws the map and then answers, in one call. TRACE [28] uses its own prompts with our VLM and

Change to our list	Accuracy change [95% interval]
Names only, without positions or sizes	−8.6 [−11.7, −5.4]
Only names that map to VSI-Bench categories	−2.6 [−4.6, −0.5]
Half of the objects removed at random	−3.8 [−6.4, −1.2]
Fixed vocabulary only, no caption names	−0.5 [−2.2, 1.3]
Stricter detection threshold	−0.5 [−2.3, 1.4]
Only objects seen in at least three frames	−0.2 [−1.9, 1.5]
List only, without the frames (text only)	−11.8 [−14.8, −8.9]
No list (video alone)	−5.9 [−9.0, −2.9]

Table S5. **Table 5 of the paper with intervals.** Gemini 3.8 Flash on the 800 questions: change in accuracy against the full list (74.5%), in points.

Change to the GT list	Lost by the VLM [95% interval]
Missing	17.7 [15.0, 20.6]
Duplicated	6.3 [4.5, 8.0]
Size ×2	23.9 [22.2, 25.6]
Shifted 1 m	14.5 [12.2, 16.9]
Shifted 0.5 m	7.6 [5.7, 9.5]
Unrelated objects added	1.7 [0.3, 3.1]
No list	26.2 [22.7, 29.7]

Table S6. **Table 6 of the paper with intervals.** Accuracy lost by Gemini 3.8 Flash against the intact ground-truth list, in points over the six tasks, on the 592 questions that ask about specific objects (572 for unrelated objects). The detection columns of Tab. 6 use the lists of the same questions.

frames, so its result differs from the one it reports with Gemini 3 Pro [54]. GCA [11] ran from its public code with its MMSI-Bench [71] settings.

Figure S3 shows the request with which GPT-6 Astra [52] builds its own list (Sec. 4.3).

E. The controlled error study

The larger study of Sec. 5.1 used Gemini 3.5 Flash-Lite on all 1,587 sampled questions, with more errors and more levels of each. Table S7 defines every error, Tab. S8 gives the nine hypotheses and Tab. S9 the cost of every error. Each error changes one object named in the question, and all errors of a question change the same object. For counting,

Error	Change to the ground-truth list	Levels
The asked-about object		
Missing	removed; for route planning, appearance order and answer options, every object of its category	–
Duplicated	a copy 0.62 m away in a random horizontal direction (the median offset of duplicates in our lists)	–
Shifted	centre moved horizontally in a random direction	0.1, 0.25, 0.5, 1 m
Size	all three sides scaled about the centre	$\times$0.5, 0.8, 1.25, 2
The rest of the scene		
Unrelated objects	categories in neither the scene nor the question, at their median size, at random free spots on the floor	+50%
Half of the other objects	objects the question does not name, removed at random	50%
Global scale	the whole list scaled about its centre (10th and 90th percentiles of our lists' scale error)	$\times$0.925, 1.078
Random removal	objects removed at random, whatever the question asks	20%, 50%
Another room	the ground-truth list of another scene of the same dataset	–

Table S7. **The errors of the study.** Each request applies one error at one level to the intact ground-truth list of the question's scene. Table 6 uses a missing, duplicated, doubled or shifted (0.5 and 1 m) asked-about object and unrelated objects; the larger study uses every row and level.

ID	Hypothesis (frozen wording)	Statistic and test (frozen)	Estimate [95% CI]	Decision
H1	MISS costs more than every other core referent state	For each of the 5 other core referent states: paired difference cost(MISS) $-$ cost(state), macro over T6 minus that state's answer-setting cells (fixed before the freeze as the cells where a literal solver loses at least as much as with MISS: DUP on counting, SIZE $\times$2 on object size); one-sided; intersection–union	smallest of the five: vs. SIZE $\times$0.5 11.4 [8.8, 14.0]	supported ($p < 0.001$; Holm threshold 0.0056)
H2	MISS costs more than DUP	Paired difference, macro over T6 minus counting; one-sided	23.0 [19.9, 26.0]	supported ($p < 0.001$; Holm threshold 0.0063)
H3	Removing half of the unnamed objects costs little (CTX-half)	TOST $\pm$3, macro over T6	0.3 [$-$1.3, 2.0]	supported ($p < 0.001$; Holm threshold 0.0167)
H4	Size errors cost little outside the size task	TOST $\pm$3 for SIZE $\times$0.5 and $\times$2.0, macro over T6 minus object size and absolute distance; intersection–union	$\times$0.5: 3.2 [0.8, 5.7]; $\times$2.0: 2.2 [$-$0.3, 4.7]	**not supported** (p = 0.560; Holm threshold 0.0250)
H5	0.25 m is tolerated; 1.0 m is costly (LOC)	TOST $\pm$3 at 0.25 m on the solver-selected tasks; "costly" test at 1.0 m on T6; intersection–union	0.25 m: 0.0 [$-$1.7, 1.7]; 1 m: 10.7 [8.6, 12.8]	supported ($p < 0.001$; Holm threshold 0.0125)
H6	A list with a missing referent scores below no list	Paired difference MISS $-$ RGB, macro over T6 minus counting; one-sided (< 0)	4.1 [0.8, 7.4]	**not supported** (p = 0.993; Holm threshold 0.0500)
H8r	The main costs replicate on Pro	With Gemini 3.1 Pro: cost of MISS and of LOC 1.0 m on T6, each "costly" (one-sided test of cost > 0 rejects and estimate ≥ 3); intersection–union	MISS 26.1 [18.4, 34.3]; LOC 1 m 15.0 [9.5, 21.1]	supported ($p < 0.001$; Holm threshold 0.0071)
H10	Gemini follows a list of another room	Paired difference X-scene $-$ RGB, macro over T6; one-sided (< 0)	$-$6.2 [$-$8.9, $-$3.5]	supported ($p < 0.001$; Holm threshold 0.0083)
H11	Gemini follows a wrong list over the video	List-following rate over the direction-known cells (SIZE on size questions, MISS and DUP on counting) where the solver's answer changes; one-sided test of rate > 0.5	88.8% [86.5, 91.1]	supported ($p < 0.001$; Holm threshold 0.0100)

Table S8. **Seven of the nine hypotheses are supported.** Hypotheses and tests as fixed before the study (Gemini 3.5 Flash-Lite; H8r: Gemini 3.1 Pro), decided by Holm's method at $\alpha = 0.05$. H9 was dropped and H7 made exploratory before the study; H8 was replaced by H8r. Estimates are in points, except for H11 (share of answers). MISS: the asked-about object missing; DUP: duplicated; LOC: shifted; SIZE: resized; CTX-half: half of the other objects removed; X-scene: the list of another room; RGB: no list; T6: the six tasks whose questions name objects.

it changes one object of the counted category, so a missing or duplicated object changes the true count by one. The changes are random but seeded, so they can be repeated exactly.

List-following (Sec. 5.2) is measured where we know which way an error moves the answer: a doubled box on object-size questions, and a missing or duplicated object on counting questions. We use the questions where a program that reads the list literally gives different answers for the intact and the changed list. An answer follows the list if it is closer to the program's answer for the changed list than to its answer for the intact list. H11 in Tab. S8 is the same measure for Gemini 3.5 Flash-Lite.

F1 at 3D IoU 0.25 (Sec. 5.2) pairs list and ground-truth boxes one to one within a category, highest IoU first. We compute the 3D IoU exactly: for gravity-aligned boxes, it is the overlap of the two footprints times the overlap of their heights. The open detection score of 3D-MOOD [73] is $\text{ODS} = [3\,\text{AP}_{\text{dist}} + (1-\text{TE}) + (1-\text{SE}) + (1-\text{OE})]/6$, with F1 in place of AP_{dist}. It pairs boxes by the distance of their

Error	Qs.	VLM cost	F1@.25	ODS
Asked-about object				
Missing	1,165	29.8 [27.3, 32.3]	9.1	4.9
Duplicated	1,165	10.9 [8.8, 12.9]	6.1	3.0
Shifted 0.1 m	592	−0.8 [−2.9, 1.5]	1.9	0.6
Shifted 0.25 m	1,165	1.6 [0.1, 3.3]	5.5	1.8
Shifted 0.5 m	592	2.9 [0.4, 5.3]	9.8	4.0
Shifted 1 m	1,165	10.7 [8.7, 12.8]	12.7	6.4
Size ×0.8	592	5.9 [3.6, 8.1]	0.0	1.1
Size ×1.25	592	8.3 [6.1, 10.6]	0.0	1.1
Size ×0.5	1,165	18.4 [16.7, 20.3]	13.6	2.0
Size ×2	1,165	23.0 [21.2, 24.8]	13.6	2.0
Rest of the scene				
Half the other objects removed	1,008	0.3 [−1.4, 1.9]	20.6	10.3
Unrelated objects added	572	−1.4 [−3.5, 0.8]	20.2	10.1
Scale ×0.925	600	2.9 [0.9, 5.0]	33.6	12.8
Scale ×1.078	600	−0.2 [−2.7, 2.1]	32.5	13.1
20% of all removed	572	12.9 [9.7, 16.2]	–	–
50% of all removed	594	27.2 [24.0, 30.5]	–	–
Reference				
No list	1,187	32.2 [29.4, 35.0]	–	–
List of another room	1,187	38.5 [35.7, 41.3]	–	–

Table S9. **The larger study: for Gemini 3.5 Flash-Lite, a missing asked-about object costs most, while a 1 m shift costs F1 more.** VLM cost: accuracy lost against the intact ground-truth list, in points over the six tasks, with 95% intervals. F1@.25, ODS: points lost by the same lists in F1 at 3D IoU 0.25 and in the open detection score. Qs.: the number of questions each error was run on. Errors with about 600 questions were run on the paper's 800 questions only. An error of the asked-about object needs a question that names an object.

	Detection metrics				Object	All asked-about	Gain over
List	F1@.25	P@.25	R@.25	ODS	recall	objects found	video (Tab. 4)
Ours	0.32	0.27	**0.40**	**0.59**	**0.68**	**31%**	**5.9**
ConceptGraphs [22]	**0.33**	0.50	0.28	**0.59**	0.40	5%	0.9
PanSt3R [79]	0.24	0.23	0.28	0.56	0.64	26%	1.9
3D-MOOD [73]	0.12	0.08	0.38	0.43	0.57	13%	1.1
SpatialLM [48]	0.29	**0.52**	0.22	0.55	0.30	6%	3.5

Table S10. **Detection metrics do not tell which list helps the VLM.** The lists of Tab. 4 on the 31 VSI-Bench scenes where all five align to the ground truth. P, R: precision and recall at 3D IoU 0.25. Object recall: share of ground-truth objects found. All asked-about objects found: share of questions whose named objects are all in the list. Gain: Gemini 3.8 Flash on the 800 questions (Tab. 4). Bold: best in each column.

centres, at 0.5 to 1.0 times half the diagonal of the ground-truth box. TE is that distance relative to the threshold, SE is one minus the IoU after aligning centre and orientation, and OE is the angle between the boxes divided by π.

	Points [95% interval]
Gap of the list to the ground-truth list	24.1 [19.4, 29.0]
Gain from all five fixes together	13.4 [8.2, 18.6]
Lost when one fix is left out	
Missed objects added	8.5 [4.3, 12.6]
Boxes snapped to ground truth	4.6 [−0.2, 9.1]
Duplicates removed	1.9 [−2.0, 5.9]
Background objects removed	1.2 [−2.6, 5.2]
Labels corrected	0.0 [−4.1, 4.1]

Table S11. **Fixing real lists.** Only the restored missing objects are needed for a significant gain.

Tables S10 and S11 test these findings on real lists (Sec. 5.2). Table S10 gives the detection metrics of five lists of Tab. 4. The oracle fixes of Tab. S11 start from an earlier version of our list. With the ground truth, they add the missed objects, remove duplicates, correct names, snap boxes to the ground truth and remove background objects.