Multimodal Language Models Cannot Spot Spatial Inconsistencies

Om Khangaonkar[1], **Hadi J. Rad**[2,3], **Hamed Pirsiavash**[1]
[1]University of California, Davis [2]Shell [3]TU Delft
reachomk.github.io/spatial-incon

Abstract

Spatial consistency is a fundamental property of the visual world and a key requirement for models that aim to understand physical reality. Despite recent advances, multimodal large language models (MLLMs) often struggle to reason about 3D geometry across multiple views. Rather than asking models to describe scene attributes, we introduce a more challenging task: given two views of the same scene, identify the object that violates 3D motion consistency. We propose a simple and scalable method for generating realistic, spatially inconsistent image pairs from multi-view scenes, enabling systematic evaluation of this capability. Our results show that state-of-the-art MLLMs significantly underperform human observers and exhibit substantial variability across different scene attributes, revealing a fragile and incomplete understanding of 3D structure. We hope our findings underscore the need for approaches that develop a more deeply grounded understanding of the physical world.

1 Introduction

Spatial consistency is a defining property of the physical world. As a camera moves through a static scene, objects change appearance in predictable ways governed by geometry, depth, and viewpoint. Their shapes, scales, and occlusions vary predictably and coherently across views. Humans are sensitive to violations of these relationships: any configuration that cannot physically exist immediately feels "off". For any model that aims to understand or synthesize realistic scenes or videos, it is essential to be able to spot when views are *physically impossible* to reconcile as observations of a unique static scene.

We hypothesize that while multimodal large language models (MLLMs) can describe individual views with impressive detail and accuracy, they fail to recognize where two views of a scene are mutually inconsistent. If this is true, it suggests today's SOTA MLLMs demonstrate only surface-level 3D visual understanding and lack a robust internal model of spatial geometry. As MLLMs are increasingly used to percieve and judge the physical world, this limitation increasingly becomes a bottleneck.

To study this problem systematically, we focus on a simple but important question: *can a model spot a spatial inconsistency between two images of the same scene?* This task directly probes whether a model understands 3D structure rather than merely memorizing visual statistics. A key challenge, however, is obtaining high-quality pairs of images that differ only by a physically inconsistent object. Existing 3D rendering or view-synthesis methods, such as Zero123 (Liu et al., 2023) or SEVA (Zhou et al., 2025), can generate alternative viewpoints, but are computationally expensive and often hallucinate unrealistic geometry that confound evaluation. We instead develop a simple, lightweight, fully automatic procedure that uses real multi-view scenes to create natural yet inconsistent image pairs. The algorithm requires no manual annotation and operates at more than 5 pairs per second on a single 3090, serving as a scalable tool to enable this study and future works. Given three views of the same static scene, we select the first as a reference, segment an object from the second and third views, and perform three steps: (1) remove the selected object from the second view and inpaint the missing region using an off-the-shelf inpainting model, (2) copy and paste the

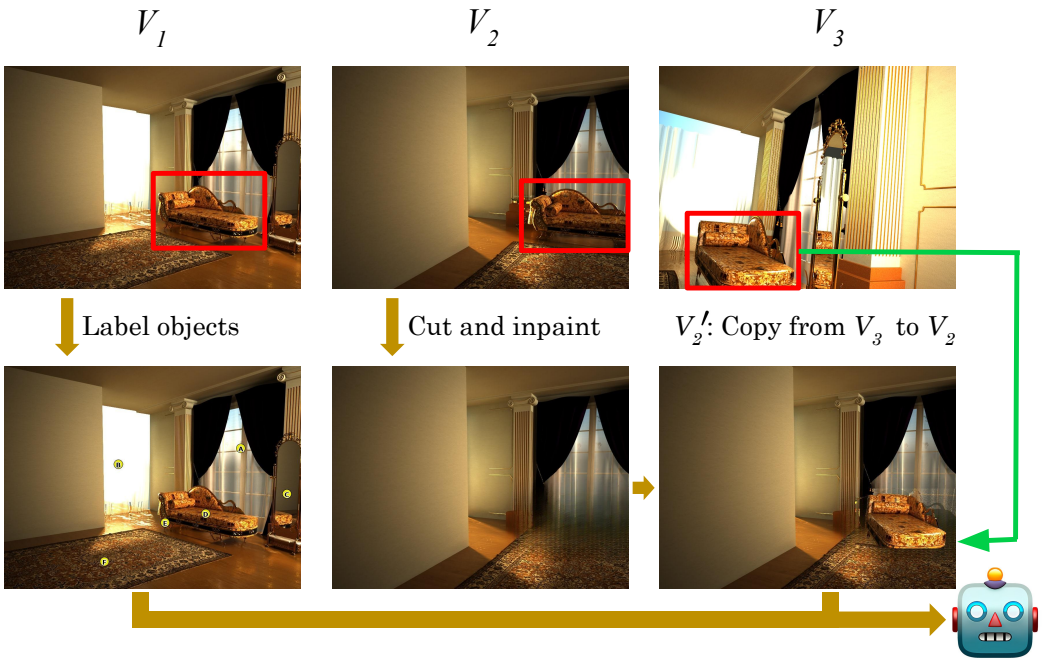

Figure 1: **Synthesizing spatially inconsistent image pairs from multi-view data.** Given three views (V_1, V_2, V_3) of the same static scene, we (1) select an object O visible in all views, (2) Erase O in view V_2 and inpaint to obtain a clean background, and (3) paste the instance of O from view V_3 back into V_2 at its original location. Because V_3 is captured from a different camera pose, the pasted object has an appearance that is incompatible with the 3D geometry implied by $V_1 \rightarrow V_2$, while all other objects remain consistent. This yields realistic image pairs with controlled spatial violations and no manual annotation. We label objects in V_1 for our forced-choice evaluation.

segmented object from the third view into the second, and (3) form a pair consisting of the first (reference) and the second (modified) view. Because the camera poses differ across the three views, the pasted object naturally acquires a 3D pose inconsistent with the reference view. The resulting pair preserves photorealism while introducing a controlled geometric violation, precisely the type of inconsistency that humans can easily detect but confuses most recent MLLMs.

Using this method, we construct a dataset of spatially inconsistent image pairs. Each pair includes ground truth annotations specifying which object violates consistency. We formulate a simple evaluation task: given two images of a scene, identify the inconsistent object. This task serves as a direct test of spatial reasoning ability in both humans and MLLMs. We evaluate several state-of-the-art models, including GPT-5, GEMINI 2.5 PRO, and QWEN3-VL, and find that human participants achieve significantly higher accuracy than all tested models.

More importantly, these failures are not merely a matter of lower average accuracy. We find that models' correct answers are highly unstable across scene attributes, unlike humans', which are very robust to these changes. For example, while human performance remains comparatively stable across viewing conditions, MLLM performance fluctuates sharply with factors such as object depth, lighting, physical plausibility, and object or scene category. We also find that while models often fail on similar pairs, the incorrect answers to these pairs vary widely, suggesting divergence in spatial reasoning across models. This suggests that current MLLMs lack a robust and human-like representation of spatial structure. These results demonstrate that our spatially inconsistent pairs provide a challenging and informative benchmark for diagnosing spatial reasoning gaps in current multimodal systems.

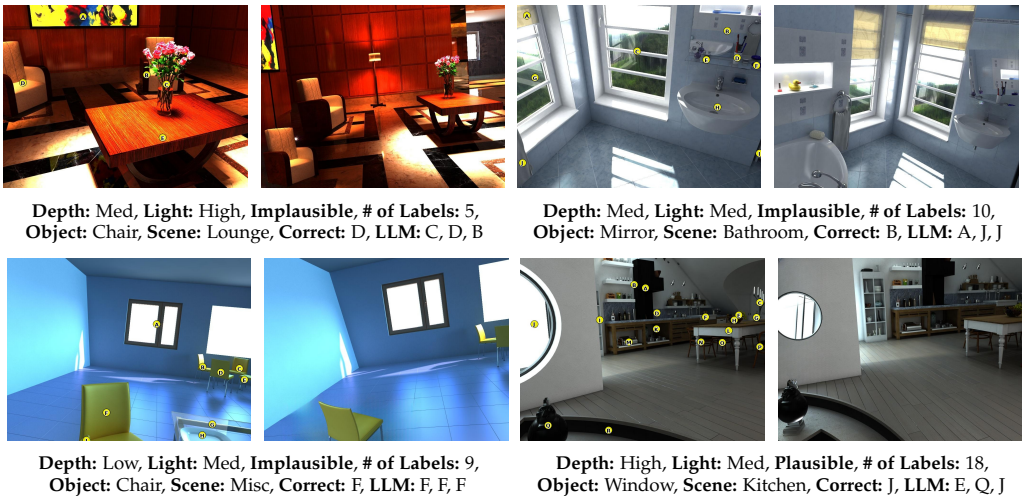

Figure 2: **Example spatial inconsistencies.** AI labels are ordered as GPT-5 (LR), GEMINI 2.5 PRO (MR) and QWEN3-VL 8B INSTRUCT. Zoom in to see labels in detail.

Thus, we present a simple, scalable algorithm to automatically generate spatially inconsistent image pairs, a dataset and evaluation task that reveal clear gaps between human and model reasoning, and an in-depth investigation of several factors that may effect their performance. We hope our findings inspire future work to build in spatial inconsistency identification as a core capability for the next generation of MLLMs.

2 Related Works

2.1 Evaluating Perception in Multimodal Language Models

Since the early days of VQA, many have questioned the perceptual limits of multimodal models. For example, the CLEVR dataset (Johnson et al., 2017) was introduced as a diagnostic to test whether models were learning superficial statistics or genuinely understanding low-level spatial and compositional relationships. This concern has re-emerged with general-purpose MLLMs, which excel at semantic tasks but often fail at core perception. A crucial line of work, exemplified by BLINK (Fu et al., 2024), found that many MLLMs "can see but not perceive." They showed that while models can solve high-level VQA, they fail at basic vision tasks like correspondence or object localization that cannot be solved with captions alone. These findings spurred a new generation of more rigorous benchmarks.

To probe a model's intuitive "world model," benchmarks like Physion++ (Tung et al., 2023) and PhysBench (Chow et al., 2025) evaluate understanding of physical properties and dynamics. Similarly, to test reasoning across multiple images, benchmarks such as MVBench (Li et al., 2024c), MMRB (Cheng et al., 2025), and MM-Spatial (Daxberger et al., 2025) evaluate a model's ability to aggregate spatial and semantic information from different viewpoints. More recently, works such as 3DSRBench (Ma et al., 2025) and 11Plus-Bench (Li et al., 2025a) further study spatial reasoning in multimodal models, introducing diagnostic tasks for evaluating geometric relationships and spatial reasoning capabilities across diverse visual contexts. However, these works ask models to describe physical scene properties (e.g., stability, mass) or temporal details (e.g., "what is the person doing after cleaning?"), while we isolate and evaluate a model's understanding of *projective geometry*. Additionally, we test not *what* is in a scene, but whether the scene's geometry is *fundamentally possible*.

2.2 Counterfactual Scene Understanding

A robust visual system should not only recognize what is present but also reject what is impossible, and counterfactual evaluation has become crucial to identify flaws in MLLMs

visual representations. Early foundational works in this domain, such as Winoground (Thrush et al., 2022) utilize semantic counterfactuals (e.g., swapping object-attribute bindings or word order) to demonstrate that models often ignore syntax and relational logic in favor of bag-of-words matching. This line of inquiry has since expanded to "hard negatives," with benchmarks like CREPE (Ma et al., 2023) and EqBen (Wang et al., 2023) testing a model's ability to distinguish between minimally different captions or images.

More recently, this paradigm has shifted toward visual and physical counterfactuals, where images are manipulated to contradict physical laws rather than just textual descriptions. HallusionBench (Guan et al., 2024) introduced a suite of visual illusions and consistency checks to probe whether MLLMs effectively hallucinate details when faced with confusing visual inputs. In the video domain, VideoHallu (Li et al., 2025b) and Vinoground (Zhang et al., 2024) generate synthetic videos that violate physical principles (e.g., gravity, object permanence) or temporal logic, identifying a critical gap in dynamic causal reasoning. Similarly, Visual Jenga (Bhattad et al., 2025) introduces a new task to evaluate a generative model's notion of scene stability and object dependence by asking the model to progressively remove objects from the scene without deforming it or creating a physical inconsistency. Recent works have also begun exploring counterfactual diagnostics for spatial and physical reasoning in multimodal systems, highlighting persistent failures when models must reason about physically grounded scene structure rather than semantic relationships alone.

However, these existing counterfactual benchmarks largely focus on semantic plausibility (is the correct object present?) or dynamic plausibility (did it move correctly over time?). We instead create multiview counterfactuals that violate the 3D motion of the physical world.

3 Synthesizing Spatial Inconsistencies

3.1 Problem Setting

Our goal is to construct pairs of views of a static scene in which a single object violates the spatial constraints implied by camera motion. Given multi-view imagery of a scene, we aim to generate two images that differ only in the 3D-consistency of one object. While one could synthesize such pairs using novel-view generation models (e.g., Zero123-style approaches (Liu et al., 2023; Zhou et al., 2025)), these methods are often slow, may introduce unintended artifacts, and may embed model-specific biases into the benchmark.

Instead, we adopt a lightweight cut-and-paste strategy that operates directly on observed views. This approach scales to large datasets, runs in almost 5 pairs per second on a single GPU, and enables us to operate on challenging compositional scenes. Most importantly, by sourcing the inserted object from a true novel view, our method produces a controlled 3D violation that alters only the object's pose without altering its structure or texture. The resulting pairs are photorealistic and contain a single, well-defined spatial inconsistency.

3.2 Source Data and Triplet Selection

We build on the Hypersim dataset (Roberts et al., 2021), which provides multi-view indoor scenes with tracked instance-level masks. We select Hypersim because of its large scale nature with many scenes from diverse categories. Additionally, because the scenes are selected for their hyper-realistic nature, the images appear very close to real-world scenes in their detail and complexity while giving us access to pixel-perfect annotations of segmentation, depth, intrinsic images, and more. However, as these properties can be inferred from other models, we can also extend our method to real-world multiview scenes, which we explore in the Appendix using DL3DV (Ling et al., 2024).

For each scene, we sample three frames (V_1, V_2, V_3) and select an object O visible in all three views. To avoid degenerate cases where the views are nearly identical, we impose that at most 75% of the objects overlap between (V_1, V_2) and between (V_2, V_3). We further restrict selection of O to objects occupying between 5% and 10% of the image area to ensure sufficient visual detail while avoiding large objects that lead to inpainting artifacts or small

Model	Overall	Model	Overall
GEMMA 3 12B	8.5	Random Chance	7.9
IDEFICS3 8B	8.9	Human	84.8
IDEFICS2 8B	11.9	GPT-5 NANO	15.3
QWEN2.5-VL 7B	15.9	GEMINI 2.5 FLASH	17.6
INTERNVL 3.5 8B	16.1	GPT-4O	19.0
LLAVA ONEVISION 1.5 8B	16.6	GEMINI 2.5 PRO (HR)	28.9
SPACEQWEN2.5-VL 3B	18.2	GEMINI 2.5 PRO (LR)	29.4
LLAMA 3.2 MULTIMODAL 11B	23.4	GEMINI 2.5 PRO (MR)	29.4
QWEN3-VL 4B	24.7	GPT-5 (HR)	30.2
QWEN3-VL 8B THINKING	25.2	GPT-5 (MR)	31.4
QWEN3-VL 8B INSTRUCT	27.6	GPT-5 (LR)	**34.2**
Ensemble (open source)	30.1	Ensemble (all models)	35.0

Table 1: **Overall accuracy (%) on the spatial inconsistency identification task.** We report overall accuracy for a random-choice baseline, humans, and a range of multimodal language models. We find that humans substantially outperform all tested models. Interestingly, higher reasoning budget does not necessarily yield higher accuracy. Ensembling open source models also allows them to perform at the level of large-scale proprietary models.

objects that lack textural cues. Additionally, we enforce the pixels of O in V_3 must project to at 40% of the area of O in V_2, ensuring that V_2 and V_3 have enough overlap.

3.3 Violating Spatial Consistency

Given a selected triplet (V_1, V_2, V_3, O), we construct an edited version V_2' in which only object O violates the geometric constraints consistent with the transition of camera view point $V_1 \rightarrow V_2$. The process consists of two steps:

1. **Removing O in V_2.** We erase O from view V_2 by inpainting its bounding box (enlarged by 5% to include context) using the LaMa inpainting model (Suvorov et al., 2021). This produces an object-free background that remains visually coherent. We inpaint the bounding box instead of the segmentation mask to ensure contour details of the object are also erased.

2. **Reinserting O from V_3.** We extract O from V_3 using its instance segmentation mask and paste it into the inpainted region of V_2'. The pasted object preserves its aspect ratio and is centered to match the size of O in V_2. All pixels belonging to other object instances within the inpainted area are restored to maintain continuity and correct occlusion ordering.

3.4 Benchmark Construction

We first auto-generate over 10,000 pairs. Following established practice to validate the benchmark (Fu et al., 2024), we manually inspect all pairs. Because of the time-consuming nature of manual filtering, we select 1-2 pairs per scene resulting in a set of 615 pairs. While this value is similar to prior works such as BLINK (Fu et al., 2024), for a larger-scale evaluation, one can use our whole auto-generated set. To probe MLLMs' spatial reasoning, we convert each pair into a forced-choice question: given views (V_1, V_2'), identify the object that violates 3D consistency.

Object labeling. We label all objects in view V_1 larger than 1000 pixels (assuming 1024×768 image size) with letter labels `A. . . Z` following prior work (Fu et al., 2024). This labeling scheme allows models to refer to objects unambiguously in natural language. If more than 26 objects satisfy the size threshold, we keep the largest 26 while ensuring that O is included. If too few objects qualify, we slightly relax the threshold but never include instances smaller than 300 pixels that lack meaningful visual detail. These labels define the answer space for both human and model evaluation.

Prompt and evaluation. At the test time, the model receives the labeled V_1 and the edited V_2' along with the prompt, which is provided in the appendix. We parse the final output and mark the item correct if the letter matches the single edited object. Our primary metric is the average accuracy across pairs.

Human evaluation protocol. We evaluate both humans and multimodal language models under the same forced-choice protocol. Given an image pair, participants and models must indicate which letter marks the 3D-inconsistent object. No additional textual hints about the manipulated object or category are provided. We measure accuracy over all 615 pairs. Section 4.2 reports detailed results broken down by depth, lighting, and physical plausibility.

4 Experiments

4.1 Models Evaluated

We evaluate a comprehensive suite of multimodal large language models (MLLMs) spanning various architectures, sizes, and training strategies. Our selection includes leading proprietary frontier models, specifically the GPT-5 family, GPT-4O, and GEMINI 2.5 (Pro and Flash) under a wide range of reasoning budgets. We also assess state-of-the-art open-weight models, including LLAMA 3.2 MULTIMODAL (Meta, 2024), QWEN 2.5, QWEN3-VL INSTRUCT, and QWEN3-VL THINKING (Bai et al., 2025b;a), GEMMA 3 (Team et al., 2025), IDEFICS2 AND 3 (Laurençon et al., 2024), LLAVA ONEVISION 1.5 (Li et al., 2024b), and INTERNVL 3.5 (Wang et al., 2025). Finally, to investigate the impact of domain-specific fine-tuning and inference-time reasoning, we include spatially specialized variants such as SPACEQWEN2.5 (Chen et al., 2024). For GPT-5 and GEMINI 2.5 PRO, we also report low (LR), medium (MR) and high (HR) reasoning levels.

4.2 How Many Spatial Inconsistencies Can MLLMs Catch?

As shown in Table 1, humans can easily spot spatial inconsistencies with relative ease at 84.8% accuracy. However, most models struggle with our task, with the very best (GPT-5 with "low" effort reasoning) achieves only 34.1%. This suggests that learning to spot inconsistencies still remains a perceptual gap in multimodal language models, even as accuracy in many 3D tasks such as depth or correspondence approaches human capabilities (Bai et al., 2025b). Surprisingly, models such as SPACEQWEN, which have been finetuned to excel at spatial reasoning tasks, still fail to spot inconsistencies. This may suggest that existing methods to teach models spatial understanding still leave large gaps in their 3D reasoning capabilities. Additionally, as is shown by the result corresponding to QWEN3-VL, GEMINI 2.5 PRO, and GPT-5, increasing test-time compute *does not* lead to improved performance. In fact, we consistently find that increasing reasoning effort often degrades model performance or leads to no further improvement. This finding is counterintuitive to the promise of visual reasoning and merits further investigation, which is out of the scope for this paper.

Because the camera poses differ across (V_1, V_2, V_3), the appearance of O as seen from V_3 will generally not be geometrically compatible with its appearance in V_1. This edit therefore introduces a precise 3D inconsistency while leaving all remaining objects unchanged. Models with strong 3D geometric understanding must detect mismatches in scale, orientation, shading, correspondence, and occlusion patterns across the two views.

We also report results after ensembling the models together. We assign each model votes based on its performance relative to the lowest performing model, GEMMA 3 12B. We weight i'th model's binary vote using $\frac{\text{accuracy of model } i}{\text{accuracy of GEMMA 3 12B}}$, then, choose the answer with highest summation of weighted votes. As shown in Table 1, we see a small but significant increase in performance when ensembling all models. Interestingly, ensembling just open-source models enables them to perform on par with proprietary models like GPT-5 (HR).

Depth and lighting factors. Because Hypersim dataset provides ground-truth depth and per-pixel illumination, we can directly examine how geometric and photometric factors

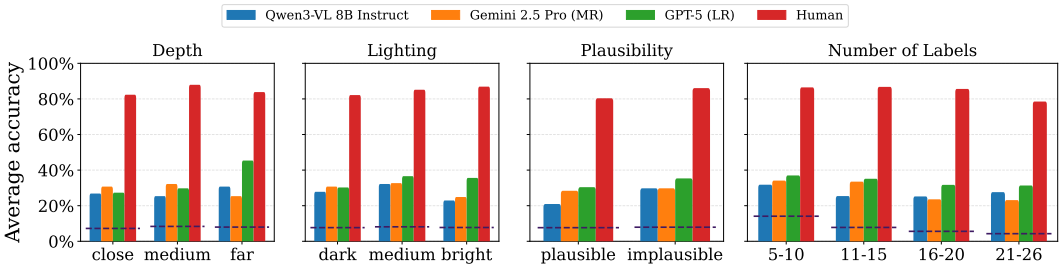

Figure 3: **Model accuracy varies across scene attributes, but much less across the number of labels per pair.** Left: We report accuracy for identifying the single spatially inconsistent object, stratified by inconsistent object depth (close/medium/far), average pair brightness (dark/medium/bright), and the augmented object's physical plausibility. While humans remain comparatively robust across conditions, models can show substantial sensitivity to changes in the pair's scene composition. Right: We partition the dataset based on the number of labels per pair and report accuracy. Interestingly, this has a minor effect on humans but not much of one on models. *Dashed line represents random chance.*

influence performance. For each inconsistent pair, we compute the depth of the modified object and the average scene brightness, then partition the dataset into 3 equally sized bins based on the empirical distributions: *close*, *medium*, and *far* for object depth, and *dark*, *medium*, and *bright* for lighting. For object depth, we use the ground-truth depth for frame V_1, and for lighting we use the luminance averaged across V_1 and V_2. This categorization enables systematic comparisons of human and model accuracy under different viewing conditions. Surprisingly, we find that most multimodal models perform substantially better on objects that are *far* relative to those that are *close*, with differences exceeding 15 percentage points, whereas human accuracy is highest for *medium*-distance objects and remains stable for *close* and *far* ones. This pattern suggests that current MLLMs exhibit geometric biases related to depth perception. Lighting reveals similar discrepancies: humans are consistently more accurate in *bright* scenes, while models tend to perform best under *medium* illumination, potentially reflecting biases inherited from training data.

Physical plausibility of poses. Not all spatially inconsistent poses are equally obvious. Some pose manipulations are obviously impossible in everyday scenes (e.g., a chair with two legs hovering above the ground), while others are subtle geometric violations that are harder to notice at a glance. To capture this variation, we categorize each pair by the *physical plausibility* of the inconsistent pose. We estimate the camera transformation between frames V_2 and V_3 of the underlying Hypersim sequence and use the induced roll to approximate whether the inserted object appears stably supported or not. Pairs with less than 5° roll are labeled as *plausible*, meaning the object appears roughly stable under gravity, while those with larger roll are labeled as *implausible*. This allows us to analyze how the physical compatibility of the inserted pose affects detection difficulty. We find that both humans and models more easily identify inconsistencies in implausible pairs, as these poses tend to be unusual in real-world settings.

Number of labels per pair. In Figure 3, we group our evaluation set by the number of labels per pair and plot the accuracy across models and groups. Surprisingly, we do not see a major decrease in accuracy, even when the number of choices doubles or quadruples. This suggests that the model struggles to select the best choice out of a shortlist, and that adding more "choices" does not necessarily make the task harder.

Inconsistent object category. We ask if models are equally strong at spotting inconsistencies across object types. We reuse the categories Hypersim annotates each object with. As some objects have an "unknown" category label, we group these into the "miscellaneous" category. We also add any pairs with inconsistent objects from categories with less than 10 pairs across the entire set into the "miscellaneous" category, as the individual sample size may be too small.

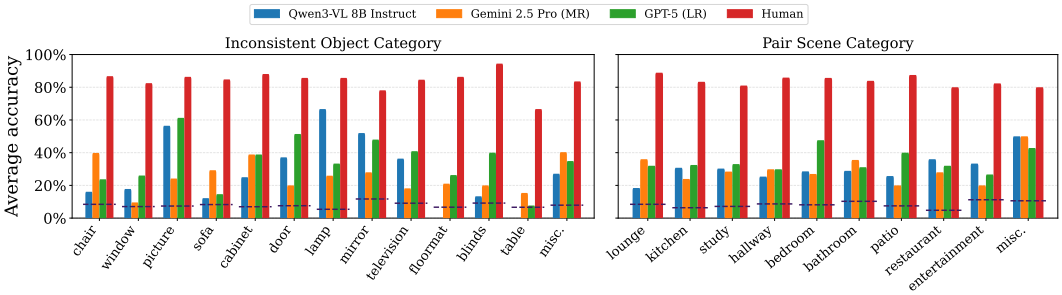

Figure 4: **Accuracy varies greatly across the inconsistent object or pair scene categories.** While humans are relatively consistent across all settings, the models show large amounts of variance. This suggests that their 3D understanding is brittle across our diverse visual world. *Dashed line represents random chance.*

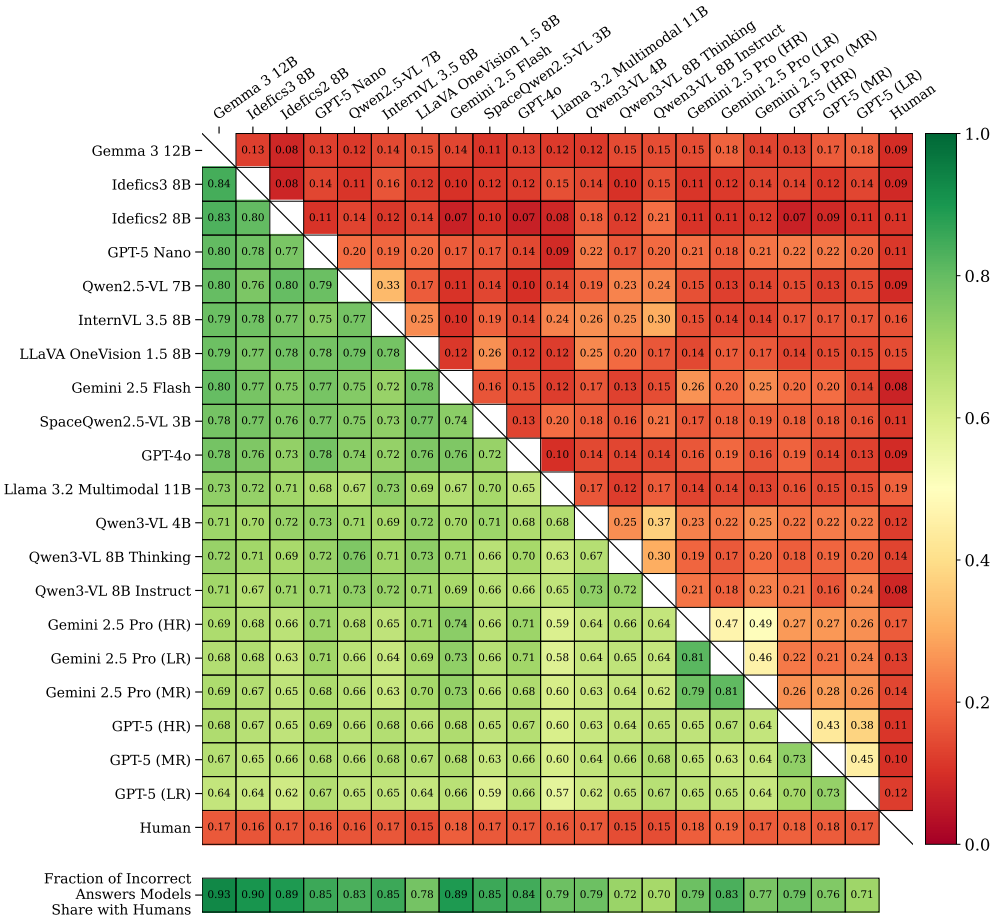

Figure 5: **Models get similar questions wrong, but don't pick similar answers.** On the bottom left, we report the IoU between two models' sets of incorrect questions. On the top right we take the intersection of incorrect questions between two models and report the fraction that they both produce the same wrong answer. From the last row, we find that most inconsistencies that trick the human, trick the AI models too. However, models rarely guess similar answers when wrong, suggesting large variance in 3D understanding.

In Figure 4, we plot the accuracy across each class type for our top models and find that, while humans are consistent across object types (with each class's accuracy within a few points of the others), MLLMs accuracy varies greatly between classes. For example, while QWEN3-VL 8B INSTRUCT has an average accuracy of 27.6, it ranges between 0 and 67% depending on the category. We also see that model rankings vary greatly across object

types, though none approach to human accuracy. For example, QWEN3-VL 8B INSTRUCT surprisingly strongly outperforms GPT-5 (LR) and GEMINI 2.5 PRO (MR) at lamps.

Scene category. To enable analysis across scene types, we group them into nine categories, plus a "miscellaneous" category. We first have QWEN3-VL 8B INSTRUCT caption the scene for each pair's initial unmodified pairs V_1 and V_2, then feed all the captions into GPT-5 and have it create and assign the 9 categories. We report results in Figure 4, where we observe a large amount of variance across categories. For example, QWEN3-VL 8B INSTRUCT mostly outperforms GEMINI 2.5 PRO (MR), but gets lower overall accuracy due to its weak performance on Lounge scenes. Similarly, GPT-5 (LR) gains most of its advantage from Bedrooms and Patios.

Overlapping incorrect answers. Figure 5 reveals a striking asymmetry between common wrong questions and common wrong answers. In the lower-left triangle, model–model pairs exhibit substantial overlap in the questions they answer incorrectly. In contrast, human–model IoU appears low, largely because humans make far fewer errors than the MLLMs, so the union is dominated by the model's failures. To account for this imbalance, we add an extra row reporting the fraction of humans' wrong questions that are also wrong for the model. These results suggest that humans and models often get similar questions wrong. However, in the upper-right triangle, humans and the same models rarely produce the same incorrect answer for the same subset of failures that is common. Thus, tend to agree on which examples are hard but not on how they are wrong. This suggests that current 3D failures are not driven by a single coherent alternative interpretation of the scene, but instead reflect unstable and model-specific breakdowns in spatial reasoning.

4.3 Are Models Spotting Inconsistencies or Artifacts?

While our cut-and-paste method offers a simple and scalable way to synthesize inconsistencies, it operates in pixel space. This raises a natural question: are the models actually spotting 3D inconsistencies, or are we just evaluating artifact spotting?

Model	Our Benchmark	Artifacts only (V_2 self-paste)	No inconsistency or artifacts (V_2 no change)
Random Chance	7.9	7.9	7.9
QWEN3-VL 4B	24.7	19.5	20.7
QWEN3-VL 8B	27.6	21.3	22.0
GEMINI 2.5 PRO (MR)	29.4	13.8	15.3
GPT-5 (LR)	34.2	15.1	13.7

Table 2: **Models do not succeed by detecting inpainting artifacts alone.** We compare performance on our full benchmark against two controls: a self-pasting baseline that preserves inpainting artifacts while removing the inconsistency, and a no-change baseline that contains neither artifacts nor inconsistency. Across all models, self-pasting performs similarly to the no-change baseline, indicating that artifact cues by themselves do not explain benchmark performance.

We investigate this through two lenses. First, we design a "self-pasting" baseline that contains inpainting artifacts but *no spatial inconsistency*. More specifically, given frames V_1, V_2, V_3, we remove the original object in V_2, expand and inpaint the bounding box and paste in the unmodified object from V_2 to V_2, instead of pasting a new view of the object from V_3. This allows us to ablate the effect of the artifacts in isolation from any spatial inconsistency. To contextualize our results, we compare against our original results (which have both artifacts and an inconsistency). Second, we use a "no change" baseline, which uses the unmodified V_2 image, and thus lacks both artifacts and an inconsistency.

Interestingly, in Table 2, "self-pasting" performs no better (and sometimes worse) than the "no change" baseline across all models. This suggests that inpainting artifacts are so small that they effectively go unnoticed by the model. Also, notice how the "no change" baseline performs higher than random chance, even though it lacks a spatial inconsistency. We

	# of self-pasted objects					
Model	None	1	3	5	10	all
QWEN3-VL 4B INSTRUCT	24.7	25.0	24.9	25.2	25.0	25.0
QWEN3-VL 8B INSTRUCT	27.6	27.3	26.7	26.0	25.0	25.4
GEMINI 2.5 PRO (MR)	29.4	28.1	29.8	28.9	26.7	24.7
GPT-5 (LR)	34.2	31.7	34.0	33.0	33.5	30.2

Table 3: **Self-pasting additional objects does not make our task harder.** Performance remains largely stable even when many or all objects are self-pasted, suggesting that models are looking for 3D structure and the task is not confounded by artifacts.

	Inpainting bbox expansion						
Model	5%	0%	10%	25%	50%	75%	100%
QWEN3-VL 4B INSTRUCT	24.7	25.9	26.3	27.6	31.9	35.8	37.6
QWEN3-VL 8B INSTRUCT	27.6	27.8	27.8	28.8	34.8	39.5	44.1
GEMINI 2.5 PRO (MR)	29.4	30.3	28.2	29.1	39.5	43.2	44.9
GPT-5 (LR)	34.2	31.7	30.6	34.0	43.3	42.8	48.0

Table 4: **Inpainting artifacts are too small to confound evaluation**. We gradually increase the bounding box for inpainting until the artifacts are too large to notice (which occurs around 25-50%). We use 5% when generating pairs, which is far smaller than this threshold.

hypothesize this is due to some common bias between the model and our object selection criteria, and is likely similar to why accuracy stays mostly consistent as choices increase.

To further investigate the role of artifacts, we ask if introducing new artifacts outside of the inconsistency will make it more challenging for the model to select the modified object. To evaluate this, we self-paste several additional objects (ranging just one to all objects) in the scene to artificially synthesize inpainting artifacts. If our model did not have any 3D understanding and was simply detecting artifacts, it would be unable to select the inconsistency due to the amount of confounding artifacts. However, in Table 3 we find that regardless of how many extra objects we self-paste, there is little to no effect on inconsistency detection. Even in the extreme case where *all* objects are self-pasted, there is only a very minor drop in accuracy.

Second, we ask at what threshold the artifacts actually start to make a difference on the model, and how close we are to that limit. Unlike self-pasting experiments where we hope for the accuracy to stay invariant to changes, this is a *destructive study* where we actively increase the inpainting artifacts until we see substantial increases in accuracy because of confounding artifacts. In Table 4, we observe that our bounding box expansion of 5% does not create artifacts large enough for the model to notice. If we do no expansion (0%), this can sometimes make it easier to spot as the cut-and-paste will not appear natural. When increasing the inpaint area we only start to see minor increases at around 25% expansion and it is not until 50% expansion till the increase becomes prominent. These results (along with the previous self-pasting experiments) suggest that an increase in accuracy on our benchmark implies an improved 3D understanding and is not due to confounding artifacts.

5 Conclusion

We introduced *spatial inconsistencies*, a form of evaluating MLLMs' 3D understanding by asking them to spot what is incorrect in a geometrically impossible scene. We show how to synthesize them using a simple and scalable cut-and-paste method and use them to evaluate MLLMs. Our results demonstrate that while this task is simple for humans, it remains challenging for MLLMs. Furthermore, we conduct an in-depth investigation of our results and find that MLLMs exhibit large variance across scenes, objects, and their attributes, indicating their 3D understanding is very brittle and inconsistent. We hope our work inspires others to look beyond training multimodal models to simply describe the visual world, and instead to teach them its fundamental physical and spatial constraints.

References

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025b. URL https://arxiv.org/abs/2502.13923.

Anand Bhattad, Konpat Preechakul, and Alexei A Efros. Visual jenga: Discovering object dependencies via counterfactual inpainting. *arXiv preprint arXiv:2503.21770*, 2025.

Boyuan Chen, Zhuo Xu, Sean Kirmani, Brian Ichter, Danny Driess, Pete Florence, Dorsa Sadigh, Leonidas Guibas, and Fei Xia. Spatialvlm: Endowing vision-language models with spatial reasoning capabilities. *arXiv preprint arXiv:2401.12168*, 2024. URL https://arxiv.org/abs/2401.12168.

Ziming Cheng, Binrui Xu, Lisheng Gong, Zuhe Song, Tianshuo Zhou, Shiqi Zhong, Siyu Ren, Mingxiang Chen, Xiangchao Meng, Yuxin Zhang, et al. Evaluating mllms with multimodal multi-image reasoning benchmark. *arXiv preprint arXiv:2506.04280*, 2025.

Wei Chow, Jiageng Mao, Boyi Li, Daniel Seita, Vitor Guizilini, and Yue Wang. Physbench: Benchmarking and enhancing vision-language models for physical world understanding. *arXiv preprint arXiv:2501.16411*, 2025.

Erik Daxberger, Nina Wenzel*, David Griffiths*, Haiming Gang, Justin Lazarow, Gefen Kohavi, Kai Kang, Marcin Eichner, Yinfei Yang, Afshin Dehghan, and Peter Grasch. Mm-spatial: Exploring 3d spatial understanding in multimodal llms. In *ICCV*, 2025. URL https://arxiv.org/abs/2503.13111.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see but not perceive. *arXiv preprint arXiv:2404.12390*, 2024.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, et al. Hallusionbench: an advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14375–14385, 2024.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2901–2910, 2017.

Hugo Laurençon, Léo Tronchon, Matthieu Cord, and Victor Sanh. What matters when building vision-language models?, 2024. URL https://arxiv.org/abs/2405.02246.

Baiqi Li, Zhiqiu Lin, Deepak Pathak, Jiayao Li, Yixin Fei, Kewen Wu, Tiffany Ling, Xide Xia, Pengchuan Zhang, Graham Neubig, and Deva Ramanan. Genai-bench: Evaluating and improving compositional text-to-visual generation. 2024a.

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

Chengzu Li, Wenshan Wu, Huanyu Zhang, Qingtao Li, Zeyu Gao, Yan Xia, José Hernández-Orallo, Ivan Vulić, and Furu Wei. 11plus-bench: Demystifying multimodal llm spatial reasoning with cognitive-inspired analysis. *arXiv preprint arXiv:2508.20068*, 2025a.

Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22195–22206, 2024c.

Zongxia Li, Xiyang Wu, Guangyao Shi, Yubin Qin, Hongyang Du, Fuxiao Liu, Tianyi Zhou, Dinesh Manocha, and Jordan Lee Boyd-Graber. Videohallu: Evaluating and mitigating multi-modal hallucinations on synthetic video understanding. *arXiv preprint arXiv:2505.01481*, 2025b.

Zhiqiu Lin, Deepak Pathak, Baiqi Li, Jiayao Li, Xide Xia, Graham Neubig, Pengchuan Zhang, and Deva Ramanan. Evaluating text-to-visual generation with image-to-text generation. *arXiv preprint arXiv:2404.01291*, 2024.

Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22160–22169, 2024.

Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object, 2023.

Grace Luo, Jonathan Granskog, Aleksander Holynski, and Trevor Darrell. Dual-process image generation. In *ICCV*, 2025.

Wufei Ma, Haoyu Chen, Guofeng Zhang, Yu-Cheng Chou, Jieneng Chen, Celso de Melo, and Alan Yuille. 3dsrbench: A comprehensive 3d spatial reasoning benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6924–6934, 2025.

Zixian Ma, Jerry Hong, Mustafa Omer Gul, Mona Gandhi, Irena Gao, and Ranjay Krishna. Crepe: Can vision-language foundation models reason compositionally? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10910–10921, 2023.

Meta. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Mike Roberts, Jason Ramapuram, Anurag Ranjan, Atulit Kumar, Miguel Angel Bautista, Nathan Paczan, Russ Webb, and Joshua M. Susskind. Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding. In *International Conference on Computer Vision (ICCV) 2021*, 2021.

Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. *arXiv preprint arXiv:2109.07161*, 2021.

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

Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5238–5248, 2022.

Hsiao-Yu Tung, Mingyu Ding, Zhenfang Chen, Daniel Bear, Chuang Gan, Josh Tenenbaum, Dan Yamins, Judith Fan, and Kevin Smith. Physion++: Evaluating physical scene understanding that requires online inference of different physical properties. *Advances in Neural Information Processing Systems*, 36:67048–67068, 2023.

Tan Wang, Kevin Lin, Linjie Li, Chung-Ching Lin, Zhengyuan Yang, Hanwang Zhang, Zicheng Liu, and Lijuan Wang. Equivariant similarity for vision-language foundation models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11998–12008, 2023.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

Jianrui Zhang, Cai Mu, and Yong Jae Lee. Vinoground: Scrutinizing lmms over dense temporal reasoning with short videos. *arXiv*, 2024. URL https://arxiv.org/abs/2410.02763.

Longtao Zheng, Ruiqing Wang, Deheng Ye, and Bo An. Controlling video generation with vision language models, 2026. URL https://openreview.net/forum?id=6SC61wyq8w.

Jensen (Jinghao) Zhou, Hang Gao, Vikram Voleti, Aaryaman Vasishta, Chun-Han Yao, Mark Boss, Philip Torr, Christian Rupprecht, and Varun Jampani. Stable virtual camera: Generative view synthesis with diffusion models. *arXiv preprint*, 2025.

A Extended Results

We had several figures evaluating our best models on various splits of the dataset. Here, we report accuracy on these same splits for all models.

Model	Object Depth			Scene Lighting			Plausibility	
	Close	Medium	Far	Dark	Medium	Bright	IP	P
Random Chance	7.2	8.0	8.4	7.7	8.1	7.8	7.9	7.7
Human	83.3	87.3	83.8	82.3	85.4	86.8	86.3	80.1
Gemma 3 12B	13.2	7.3	4.9	7.8	11.2	6.3	9.4	5.4
Idefics3 8B	8.8	10.7	7.3	8.8	9.8	8.3	7.7	12.8
Idefics2 8B	9.8	12.2	13.7	12.2	13.2	10.2	12.4	10.1
GPT-5 Nano	14.6	17.6	13.7	16.1	15.1	14.6	16.1	12.8
Qwen2.5 VL 7B	11.7	12.7	23.4	19.0	19.0	9.8	17.1	12.2
InternVL 3.5 8B	14.2	13.7	20.5	16.6	16.1	15.6	16.5	14.9
LLaVA OneVision 1.5 8B	11.2	14.6	23.9	18.1	19.0	12.7	17.3	14.2
Gemini 2.5 Flash	18.1	18.1	16.6	18.1	21.0	13.7	18.4	14.9
SpaceQwen2.5 VL 3B	23.4	16.6	14.6	20.5	19.5	14.6	19.3	14.9
GPT-4o	16.1	17.1	23.9	22.9	14.2	20.0	19.9	16.2
Llama 3.2 Multimodal 11B	30.7	23.4	16.1	21.0	25.4	23.9	24.2	21.0
Qwen3 VL 4B	31.2	24.9	18.1	23.9	25.4	24.9	25.9	21.0
Qwen3 VL 8B Thinking	18.1	24.4	33.2	31.2	27.8	16.6	26.8	20.3
Qwen3 VL 8B Instruct	26.8	25.4	30.7	27.8	32.2	22.9	29.8	21.0
Gemini 2.5 Pro (HR)	28.3	30.2	28.3	29.8	33.7	23.4	29.1	28.4
Gemini 2.5 Pro (LR)	**31.7**	29.3	27.3	30.2	31.7	26.3	30.6	25.7
Gemini 2.5 Pro (MR)	30.7	**32.2**	25.4	30.7	32.7	24.9	29.8	28.4
GPT-5 (HR)	28.3	25.9	36.6	28.8	32.2	29.8	30.6	29.1
GPT-5 (MR)	23.4	30.2	40.5	**32.2**	33.2	28.8	33.2	25.7
GPT-5 (LR)	27.3	29.8	**45.4**	30.2	**36.6**	**35.6**	**35.3**	**30.4**

Table 5: **Performance breakdown (% accuracy) by depth, lighting, and physical plausibility.** We report the full results for each model, of which a subset was shown in Figure 3.

Model	Chair	Window	Picture	Sofa	Cabinet	Door
Random Chance	8.4	7.0	7.4	8.3	7.0	7.6
Human	86.8	82.5	86.4	84.8	88.1	85.7
Gemma 3 12B	15.3	4.1	8.1	7.3	8.3	11.4
Idefics3 8B	8.5	1.4	4.8	17.1	11.1	11.4
Idefics2 8B	4.2	19.2	19.4	7.3	13.9	5.7
GPT-5 Nano	13.6	4.1	19.4	12.2	11.1	22.9
Qwen2.5 VL 7B	7.6	20.5	24.2	4.9	16.7	31.4
InternVL 3.5 8B	2.5	20.5	30.6	0.0	19.4	37.1
LLaVA OneVision 1.5 8B	15.3	11.0	19.4	12.2	19.4	31.4
Gemini 2.5 Flash	23.7	2.7	17.7	9.8	36.1	14.3
SpaceQwen2.5 VL 3B	22.0	1.4	12.9	**39.0**	2.8	14.3
GPT-4o	22.0	13.7	22.6	12.2	16.7	31.4
Llama 3.2 Multimodal 11B	28.8	13.7	27.4	19.5	16.7	17.1
Qwen3 VL 4B	32.2	5.5	29.0	26.8	11.1	31.4
Qwen3 VL 8B Thinking	18.6	17.8	37.1	17.1	33.3	54.3
Qwen3 VL 8B Instruct	16.1	17.8	56.5	12.2	25.0	37.1
Gemini 2.5 Pro (HR)	35.6	11.0	40.3	34.1	33.3	20.0
Gemini 2.5 Pro (LR)	**39.8**	9.6	25.8	31.7	33.3	20.0
Gemini 2.5 Pro (MR)	**39.8**	9.6	24.2	29.3	**38.9**	20.0
GPT-5 (HR)	22.0	24.7	41.9	9.8	36.1	42.9
GPT-5 (MR)	19.5	**28.8**	51.6	17.1	**38.9**	**57.1**
GPT-5 (LR)	23.7	26.0	**61.3**	14.6	**38.9**	51.4

Model	Lamp	Mirror	Television	Floormat	Blinds	Table	Misc.
Random Chance	5.4	11.7	9.1	6.7	9.2	6.6	7.9
Human	85.7	78.1	84.6	86.4	94.4	66.7	83.6
Gemma 3 12B	3.7	4.0	0.0	10.5	0.0	0.0	9.3
Idefics3 8B	3.7	8.0	4.5	5.3	6.7	15.4	14.0
Idefics2 8B	25.9	28.0	18.2	5.3	20.0	0.0	7.8
GPT-5 Nano	11.1	20.0	22.7	15.8	0.0	0.0	23.3
Qwen2.5 VL 7B	18.5	24.0	22.7	0.0	6.7	0.0	17.8
InternVL 3.5 8B	51.9	20.0	22.7	0.0	6.7	0.0	13.2
LLaVA OneVision 1.5 8B	14.8	16.0	27.3	0.0	6.7	7.7	19.4
Gemini 2.5 Flash	7.4	24.0	13.6	10.5	13.3	15.4	21.7
SpaceQwen2.5 VL 3B	44.4	8.0	22.7	5.3	13.3	**23.1**	23.3
GPT-4o	3.7	24.0	27.3	15.8	6.7	7.7	20.9
Llama 3.2 Multimodal 11B	59.3	20.0	18.2	10.5	13.3	7.7	25.6
Qwen3 VL 4B	40.7	28.0	40.9	10.5	6.7	7.7	27.1
Qwen3 VL 8B Thinking	18.5	**56.0**	31.8	0.0	6.7	0.0	24.8
Qwen3 VL 8B Instruct	**66.7**	52.0	36.4	0.0	13.3	0.0	27.1
Gemini 2.5 Pro (HR)	18.5	16.0	13.6	**26.3**	6.7	7.7	39.5
Gemini 2.5 Pro (LR)	22.2	24.0	22.7	**26.3**	20.0	7.7	**41.1**
Gemini 2.5 Pro (MR)	25.9	28.0	18.2	21.1	20.0	15.4	40.3
GPT-5 (HR)	55.6	28.0	40.9	5.3	**40.0**	15.4	34.1
GPT-5 (MR)	33.3	40.0	**54.5**	0.0	6.7	0.0	34.1
GPT-5 (LR)	33.3	48.0	40.9	**26.3**	**40.0**	7.7	34.9

Table 6: **Accuracy by inconsistent object class.** We report the full results for each model, of which a subset was shown in Figure 4.

Model	Lounge	Kitchen	Study	Hallway	Bedroom
Random Chance	8.4	6.3	7.1	8.7	8.1
Human	88.9	83.3	81.0	85.9	85.7
Gemma 3 12B	5.6	8.5	17.4	9.0	4.8
Idefics3 8B	8.8	6.8	9.2	9.0	7.9
Idefics2 8B	12.0	9.4	16.5	7.5	11.1
GPT-5 Nano	9.6	17.1	21.1	11.9	12.7
Qwen2.5 VL 7B	12.8	18.8	20.2	17.9	15.9
InternVL 3.5 8B	13.6	13.7	27.5	14.9	12.7
LLaVA OneVision 1.5 8B	19.2	14.5	14.7	23.9	15.9
Gemini 2.5 Flash	18.4	12.8	15.6	19.4	17.5
SpaceQwen2.5 VL 3B	19.2	12.8	16.5	23.9	12.7
GPT-4o	22.4	12.8	19.3	19.4	28.6
Llama 3.2 Multimodal 11B	25.6	19.7	24.8	17.9	31.7
Qwen3 VL 4B	20.8	18.8	26.6	19.4	28.6
Qwen3 VL 8B Thinking	19.2	22.2	33.0	**31.3**	20.6
Qwen3 VL 8B Instruct	18.4	30.8	30.3	25.4	28.6
Gemini 2.5 Pro (HR)	33.6	23.9	33.9	**31.3**	23.8
Gemini 2.5 Pro (LR)	32.8	21.4	**36.7**	**31.3**	22.2
Gemini 2.5 Pro (MR)	**36.0**	23.9	28.4	29.9	27.0
GPT-5 (HR)	31.2	25.6	33.9	25.4	31.7
GPT-5 (MR)	28.8	**32.5**	31.2	29.9	31.7
GPT-5 (LR)	32.0	**32.5**	33.0	29.9	**47.6**

Model	Bathroom	Patio	Restaurant	Entertainment	Misc
Random Chance	10.3	7.5	4.8	11.2	10.6
Human	83.9	87.5	80.0	82.4	80.0
Gemma 3 12B	6.7	2.9	4.0	6.7	7.1
Idefics3 8B	13.3	8.6	8.0	13.3	14.3
Idefics2 8B	17.8	8.6	4.0	26.7	7.1
GPT-5 Nano	22.2	17.1	12.0	6.7	21.4
Qwen2.5 VL 7B	13.3	5.7	20.0	20.0	0.0
InternVL 3.5 8B	11.1	8.6	24.0	20.0	7.1
LLaVA OneVision 1.5 8B	13.3	8.6	12.0	20.0	28.6
Gemini 2.5 Flash	20.0	22.9	20.0	26.7	21.4
SpaceQwen2.5 VL 3B	17.8	28.6	28.0	20.0	21.4
GPT-4o	17.8	17.1	4.0	20.0	28.6
Llama 3.2 Multimodal 11B	24.4	14.3	32.0	20.0	21.4
Qwen3 VL 4B	26.7	22.9	**48.0**	40.0	42.9
Qwen3 VL 8B Thinking	28.9	14.3	28.0	33.3	35.7
Qwen3 VL 8B Instruct	28.9	25.7	36.0	33.3	**50.0**
Gemini 2.5 Pro (HR)	33.3	20.0	20.0	20.0	35.7
Gemini 2.5 Pro (LR)	**35.6**	22.9	20.0	26.7	**50.0**
Gemini 2.5 Pro (MR)	**35.6**	20.0	28.0	20.0	**50.0**
GPT-5 (HR)	26.7	37.1	28.0	33.3	42.9
GPT-5 (MR)	**35.6**	34.3	20.0	**46.7**	35.7
GPT-5 (LR)	31.1	**40.0**	32.0	26.7	42.9

Table 7: **Accuracy by scene class.** We report the full results for each model, of which a subset was shown in Figure 4.

Model	**5–10**	**11–15**	**16–20**	**21–26**
Random Chance	12.3	7.9	5.7	4.1
Human	86.5	86.8	85.6	78.5
Gemma 3 12B	12.1	6.5	7.3	7.5
Idefics3 8B	13.3	9.7	6.5	4.5
Idefics2 8B	16.2	9.2	11.4	10.4
GPT-5 Nano	19.1	13.0	18.7	10.4
Qwen2.5 VL 7B	21.4	15.7	13.0	11.9
InternVL 3.5 8B	16.8	12.4	13.0	23.1
LLaVA OneVision 1.5 8B	22.0	14.1	13.8	15.7
Gemini 2.5 Flash	22.5	17.8	14.6	13.4
SpaceQwen2.5 VL 3B	24.9	16.2	10.6	19.4
GPT-4o	27.2	20.5	17.9	7.5
Llama 3.2 Multimodal 11B	28.3	17.8	20.3	27.6
Qwen3 VL 4B	30.6	18.4	24.4	26.1
Qwen3 VL 8B Thinking	31.8	25.9	20.3	20.1
Qwen3 VL 8B Instruct	31.8	25.4	25.2	27.6
Gemini 2.5 Pro (HR)	28.9	31.9	30.1	23.9
Gemini 2.5 Pro (LR)	32.9	34.1	26.0	21.6
Gemini 2.5 Pro (MR)	34.1	33.5	23.6	23.1
GPT-5 (HR)	34.1	30.8	26.8	27.6
GPT-5 (MR)	**38.2**	32.4	28.5	23.9
GPT-5 (LR)	37.0	**35.1**	**31.7**	**31.3**

Table 8: **Accuracy by number of labels.** We report the full results for each model, of which a subset was shown in Figure 3.

B Applications of Spatial Inconsistencies

B.1 Scaling Up to Real-World Data

We scale up the dataset by generating pairs using DL3DV dataset, which contains over 10K real-world multiview scenes. We obtain object tracks using SAM 3 (captioned by Qwen3-VL 8B) and metric depths with Depth Pro. We generated **39K pairs** from over **6,900 scenes** with diverse settings such as parks, malls, gyms and more. We show the results of the scaled up dataset (DL3DV) in Table 9. We observe that this task is still challenging for MLLMs. We see that MLLMs score higher than the Hypersim set, which we assume is due to the substantially lower number of objects per scene. This is both due to the expensive cost of SAM3 for tracking and DL3DV's selection of outdoor scenes, which have less objects. We evaluated Qwen models with the full 39K set, but had to use a fixed random sample of 6K for Gemini/GPT-5 due to API limits and cost.

Model	Accuracy on DL3DV inconsistencies
Random Chance	18.4
Qwen3-VL 4B	47.7
Qwen3-VL 8B	57.7
Gemini 2.5 Pro (MR)	52.9
GPT-5 (LR)	52.4

Table 9: When we scale up to real-world data, our task is still challenging for models.

Model	Base Model	Hypersim	DL3DV
Hypersim Inconsistencies	24.7	66.8±1.5	55.2
Multiview Reasoning	37.6	39.9±1.4	43.6
Visual Correspondence	80.2	81.7±0.7	82.2
Relative Depth	84.7	86.9±1.1	87.7

Table 10: We show that Qwen3-VL 4B finetuned with additional scene diversity perform on par or better than our initial models at 3D perception and perform well on the Hypersim evaluation set.

Model	Base Model	Finetuned
Our Benchmark, disjoint set	24.7	69.2

Table 11: **Fine-tuning on our spatial inconsistency task yields large gains on a disjoint test split.** Accuracy (%) of the base vision–language model versus the same model fine-tuned on our benchmark, evaluated on a held-out set of image pairs with no scene overlap. Performance jumps from 24.7% to 69.2%, showing that the task is learnable and that targeted training substantially improves spatial reasoning on new scenes.

B.2 Does Learning to Spot Inconsistencies Improve Broader 3D Understanding?

Given the poor zero-shot accuracy of state-of-the-art multimodal models, a natural question arises: is learning to spot spatial inconsistencies a "useful" task? We draw on ideas from visual representation learning and use our generated inconsistencies as a pretext task. If our task is sufficiently related to downstream 3D understanding tasks, we expect to see a small but present improvement on zero-shot accuracy over the base model. To answer this question, we fine-tune Qwen3-4B using our autogenerated set of 10K pairs. We apply LoRA (Hu et al., 2021) with rank 16. For Table 10, we train for one epoch using our scaled-up inconsistency sets.

Transfer to broader 3D benchmarks. The more important question is whether this training transfers beyond our task. We evaluate the fine-tuned model on the BLINK benchmark (Fu et al., 2024), focusing on categories most closely tied to spatial geometry: "Multiview Reasoning", "Visual Correspondence", and "Relative Depth". As summarized in Table 10, training on our inconsistency-spotting task yields consistent improvements across all three metrics. We observe gains of approximately 2.3 percentage points in both "Multiview Reasoning" and "Relative Depth", and 1.3 percentage points in "Visual Correspondence". We also finetune our model on our scaled-up DL3DV dataset and see similar gains. These improvements are notable because the model receives no direct supervision on depth estimation, correspondence matching, or camera motion. Instead, identifying *which* object breaks cross-view consistency implicitly encourages the model to learn depth (for scale compatibility), correspondence (for appearance matching), and also camera motion understanding. These findings suggest our simple cut-and-paste approach does provide signal to learn 3D understanding from.

In-domain generalization. We also retrain the model with 20% of scenes held-out to ensure that training on our task can generalize to novel scenes. As shown in Table 11, fine-tuning yields a substantial improvement in performance: Qwen3 VL-4B jumps from 24.7% to 69.2% accuracy on entirely unseen scenes. This signals that the task is learnable and that our synthetic pairs provide a consistent and meaningful signal for the model to acquire a generalizable notion of 3D violations, rather than memorizing scene layouts or object textures.

B.3 Do VQA Judges Notice Spatial Inconsistencies?

Despite struggling at directly identifying inconsistent objects, modern MLLMs are widely used as VQA-style judges for evaluating image and video generation. We therefore ask: do

Model	Pairwise	Pearson	Kendall
GPT–4o	50.1	-0.14	0.32

Table 12: **VQAScore evaluation of GPT–4o on our spatial inconsistency benchmark.** We report the pairwise accuracy (%) of VQAScore (Lin et al., 2024) when asked to score both the physically consistent image and the manipulated one, along with Pearson and Kendall correlations between VQAScore and human accuracy across image pairs. The near-chance pairwise score and weak correlations indicate that the SOTA VQA-based judge poorly aligns with human assessments of spatial consistency.

these judges notice our 3D inconsistencies when used in their standard Yes/No evaluation protocol? Given a set of frames and a textual description, the model is asked a constrained question such as: *"Does this video show* {`caption`}*? Please answer Yes or No."* The alignment score is then estimated from the probability assigned to the `Yes` token. In practice, this setup is used not only to measure semantic alignment, but also as a proxy for visual plausibility: if two candidates match the prompt, the expectation is that the judge will assign a higher `Yes` score to the one that is more realistic.

Recent benchmarks (e.g., GenAI-Bench (Li et al., 2024a)) support this intuition by showing that VQA-style judges can successfully rank images that all satisfy the prompt but differ in fidelity and artifacts, suggesting some sensitivity to realism beyond pure text alignment. We hypothesize that this sensitivity to realism does *not* extend to 3D understanding, and thus may be unable to distinguish videos where object poses are modified such that the frames are impossible in a static scene.

We use our constructed pairs to test this assumption directly. For each multiview set (V_1, V_2, V_2'):

1. We first generate a caption C_1 from frame V_1 alone (without labels) using GPT-5.
2. We treat (V_1, V_2) and (V_1, V_2') as the frames sampled from two candidate videos depicting the same static scene.
3. We compute the accuracy as the fraction of items where the unmodified pair (V_1, V_2) receives a higher `Yes` score than the manipulated pair (V_1, V_2').

Assuming GenAI-Bench's finding holds, a judge that truly understands scene geometry should consistently prefer (V_1, V_2), the unmodified pair over (V_1, V_2') which is spatially inconsistent. However, as shown in Table 12, GPT-4o (the current SOTA for VQAScore eval) performs near random chance. This suggests that VQA judges does not spot spatial inconsistencies when ranking alignment of generated videos and provided captions. Additionally, when utilizing VQA judges to *control* diffusion-based generators (Luo et al., 2025; Zheng et al., 2026), they may amplify these spatial inconsistencies due to their inability to identify them.