RotBench: Evaluating Multimodal Large Language Models on Identifying Image Rotation

Anonymous ACL submission

Abstract

We investigate to what extent Multimodal Large Language Models (MLLMs) can accurately identify the orientation of input images rotated 0°, 90°, 180°, and 270°. This task demands robust visual reasoning capabilities to detect rotational cues and contextualize spatial relationships within images, regardless of their orientation. To evaluate MLLMs on these abilities, we introduce ROTBENCH – a 350-image manually-filtered benchmark comprising lifestyle, portrait, and landscape images. Despite the relatively simple nature of this task, we show that several state-of-the-art open and proprietary MLLMs, including GPT-5, o3, and Gemini-2.5-Pro, do not reliably identify rotation in input images. Providing models with auxiliary information – including captions, depth maps, and more – or using chain-of-thought prompting offers only small and inconsistent improvements. Our results indicate that most models are able to reliably identify right-side-up (0°) images, while certain models are able to identify upside-down (180°) images. None can reliably distinguish between 90° and 270°. Simultaneously showing the image rotated in different orientations leads to moderate performance gains for reasoning models, while a modified setup using voting improves the performance of weaker models. We further show that fine-tuning does not improve models' ability to distinguish 90° and 270° rotations, despite substantially improving the identification of 180° images. Together, these results reveal a significant gap between MLLMs' spatial reasoning capabilities and human perception in identifying rotation.[1]

1 Introduction

Advancements in Multimodal Large Language Models (MLLMs) have led to increased performance in complex visual tasks, such as image-text retrieval, image segmentation, and visual question answering (Li et al., 2025; Chen et al., 2023a; Ravi et al., 2024; Fu et al., 2024a; Chen et al., 2023b; Bai et al., 2025; OpenAI, 2025c; Gemini Team, 2025; Liu et al., 2023a; Deitke et al., 2024a). However, a growing body of recent work suggests that MLLMs are sensitive to simple image transformations (Anis et al., 2025), such as rotations, flips, and blurs, and they fail on tasks that are intuitive to humans (Fu et al., 2024b; Pothiraj et al., 2025; Tong et al., 2024). Downstream tasks involving a rotating camera – such as robotic arm manipulation or first-person extreme sports analysis – require MLLMs to demonstrate robust spatial reasoning, regardless of image orientation (Appendix C). Given these challenges, this work explores a fundamental question: can MLLMs identify image orientation?

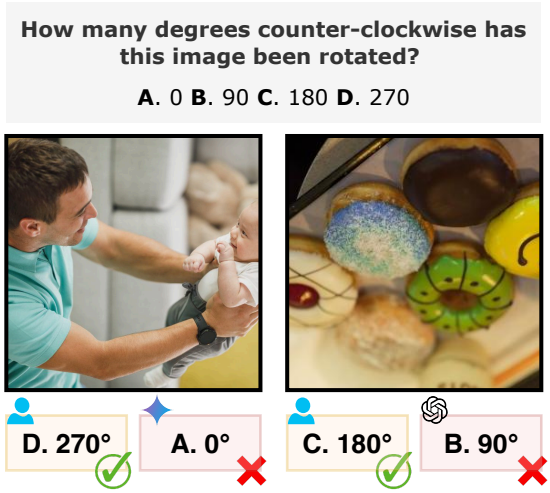

Figure 1: We present two ROTBENCH images: one (left) to Gemini-2.5-Pro, the other (right) to GPT-5. Humans can easily identify the correct rotation of the two images, but both models fail to do so.

Humans can quickly recognize whether an image has been rotated (Shepard and Metzler, 1971; Vandenberg and Kuse, 1978); for example, it is easy for us to recognize that the left image in Fig. 1 is not upright. The human viewer can use the orien-

[1] Code and data included in 'software' and 'data' fields, respectively.

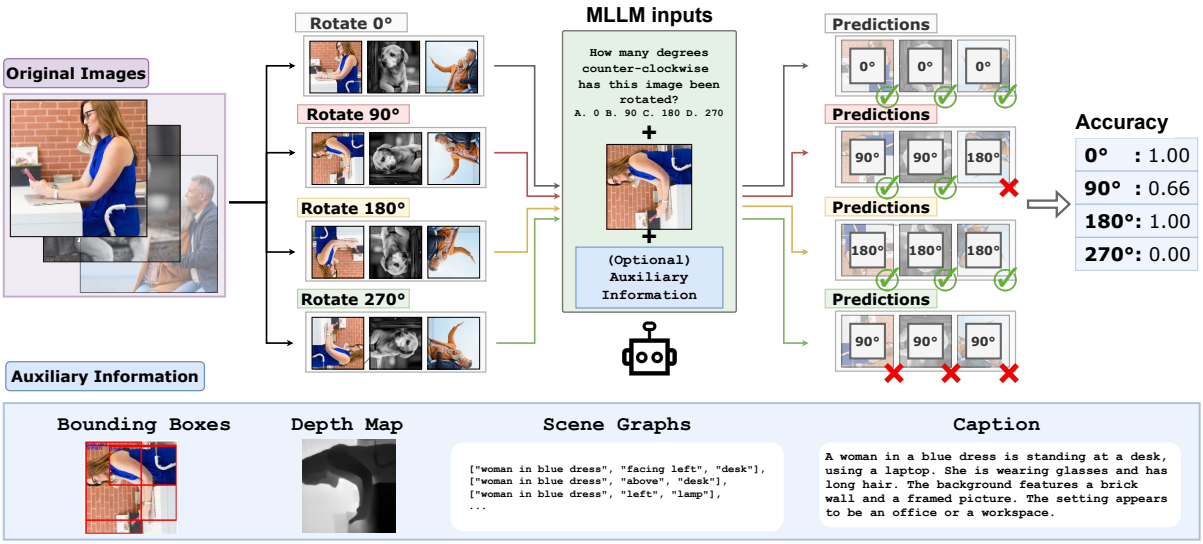

Figure 2: **RotBench** evaluation pipeline: for each image in RotBench, we rotate the image 0°, 90°, 180°, and 270° counter-clockwise. We represent the rotation estimation problem as a multiple-choice question answering problem (Appendix Q.5), and separately measure accuracy on each image orientation. We optionally provide different forms of auxiliary information to aid the model in identifying image rotation. We emphasize that all forms of auxiliary information are separately extracted for each rotation; the ground truth rotation is not marked.

tation of the couch in the background to infer that the father is actually lying on his back rather than standing up. The simple task of identifying image rotation requires reconciling the image's subjects, background, and semantics. Here, we show that identifying rotation remains a challenge, even in frontier MLLMs.

We introduce RotBench (Section 3), a benchmark for evaluating MLLMs' ability to recognize rotation in images. RotBench consists of images sampled from Spatial-MM (Shiri et al., 2024), and is comprised of two subsets: the 300-image RotBench-large and the 50-image RotBench-small. RotBench is carefully constructed to be challenging but fair, with a two-stage filtering procedure to remove images that are indistinguishable under different degrees of rotation. Section 3 describes our dataset construction.

Using RotBench, we explore whether frontier MLLMs can identify rotation in input images rotated 0°, 90°, 180°, and 270° (Fig. 2). We also evaluate whether providing various forms of auxiliary information or using chain-of-thought (Wei et al., 2022) prompting improves performance (Section 4.3). We find that models are able to consistently identify right-side-up (0°) images. However, only stronger models are able to identify upside-down (180°) images. All models fail to accurately distinguish between 90° and 270° images (Section 6.1, Section 6.2). We find adding auxiliary information offers minimal and inconsistent improvement, often improving performance on 270° images at the expense of 90° images.

Leveraging MLLMs' tendency to accurately identify 0° images, we attempt to improve performance by *further* rotating an input image 0°, 90°, 180°, 270° and presenting MLLMs with all rotations simultaneously. Using this approach, reasoning models (o3 and Gemini-2.5-Pro) show performance improvements, while weaker models see performance degradation. Further extending this idea, we utilize a voting approach to algebraically obtain a majority vote for the correct ground truth orientation. We obtain a model prediction for each further rotation by subtracting the added angle. While this approach does show significant performance improvements on weaker models, it lacks scalability as it requires multiple model calls for each prediction, and assumes *a priori* knowledge of all possible orientations (Appendix M).

Finally, fine-tuning on out-of-domain data (Section 6.3) significantly improved performance on identifying 180° images but not 90° and 270°. Interestingly, we find an oscillating pattern of performance changes between 90° and 270° as training progresses. Any performance improvement in 90° is matched with a degradation of 270° and vice versa. These results suggest the presence of two local optima hindering across-the-board progress.

Our findings demonstrate the tested MLLMs

significantly underperform compared to humans when it comes to spatial reasoning involving rotation detection, highlighting the need for integrating rotation-awareness into modern training pipelines.

2 Related Work

This section reviews the literature most closely related to our work. Additional relevant areas are discussed in Appendix B.

Image orientation estimation. Fine-tuning models to identify image orientation has been the focus of prior work (Xu et al., 2024). For example, Fischer et al. (2015) and Joshi and Guerzhoy (2017) focused on fine-tuning convolutional neural networks (CNNs) to estimate and identify image rotation. While our work tackles a similar task, we are instead interested in the problem as a test of general-purpose MLLMs' inherent reasoning abilities, i.e., whether they can estimate image rotation without extensive fine-tuning. Note that our work aligns more closely with the problem of *image*, not *camera*, orientation estimation. We further discuss camera orientation estimation in Appendix B.

Spatial reasoning in MLLMs. Beyond robustness, spatial relation understanding is a notable weakness of current MLLMs. Kamath et al. (2023) curate the What's Up benchmark to isolate "left-/right/above/below" relations, showing a significant gap in performance between humans and MLLMs. Shiri et al. (2024) further develop the Spatial-MM dataset and demonstrate that providing bounding boxes or scene graphs yields only modest gains. Both illustrate that MLLMs struggle with certain challenging cross-modal spatial reasoning tasks.

Gap between human perception and MLLMs. A growing body of work shows MLLMs exhibit fundamental gaps compared to human perceptual capabilities. Pothiraj et al. (2025) propose CAPTURe – a benchmark for occluded object counting – and report sharp drops in model accuracy on both synthetic and real images. Zhou et al. (2025) proposes MMVM, a benchmark for visual matching across images. Fu et al. (2024b) collect BLINK, a dataset comprised of visual tasks humans can solve in a 'blink,' such as identifying visual similarity and relative depth. Both Zhou et al. (2025) and Fu et al. (2024b) report low zero-shot accuracy on their respective tasks, suggesting MLLMs lack many of the intuitive reasoning mechanisms that underpin human visual perception. In this vein, our work provides a novel perspective to analyze and interpret the spatial reasoning capabilities of MLLMs, with results indicating that models struggle with this previously underexplored challenge.

3 RotBench

We introduce **RotBench**, a benchmark for evaluating models' ability to identify rotation in input images. RotBench is created using images from Spatial-MM (Shiri et al., 2024) and includes two subsets: the 300-image **RotBench-large** and the 50-image **RotBench-small**. While rotating an image is straightforward, not all images are meaningful under rotation. A rotated portrait of a human, such as the image shown in Fig. 2, will be easily noticed by human viewers. However, a top-down view of a simple tabletop does not significantly differ when rotated (Fig. 9). We use a two-stage filtering process to ensure different rotations of each image are clearly distinguishable. This section provides an overview of our filtering procedure. Appendix A describes our dataset procedure and statistics in further detail.

Stage 1. We randomly sample 300 images from Spatial-MM (Shiri et al., 2024). Stage 1 involves a single annotator. Depending on the amount of visual signals available, the Stage 1 annotator decides to either accept, discard, or flag each image. Flagged images then proceed to Stage 2. We provide further examples and details of accepted, flagged, and discarded images in Fig. 9.

Stage 2. Stage 2 involves a group of three human evaluators. Each flagged image is rotated 0°, 90°, 180°, and 270° counter-clockwise then presented to the evaluators as multiple-choice questions (Appendix A.4). During Stage 2, an evaluator will see each image four times, once per orientation. Any image that elicits an incorrect answer from two or more evaluators across all four orientations is discarded. Otherwise, the image is accepted. Stage 1 and 2 are repeated until a total of 300 images are accepted.

RotBench-large and RotBench-small. All images that have been accepted in Stage 1 and 2 are organized into RotBench-large. As each image is rotated in four orientations, obtaining human performance on RotBench-large is costly. Luckily, Stage 2 provides a human baseline for the subset of RotBench-large images that have been flagged (25 images). We expand this subset by further sampling images from Spatial-MM. From these additional images, we only select images that

fit the criteria for flagging to proceed to another round of Stage 2 evaluation. This process repeats until we reach a total of 50 images, organizing them into ROTBENCH-SMALL.[2]

4 Experiment Setup

4.1 Models

We evaluate various open-weight and proprietary MLLMs on ROTBENCH: Qwen-2.5-VL-7B-Instruct (Bai et al., 2025); GPT-4o (OpenAI et al., 2024), GPT-4.1 (OpenAI, 2025a), o3 (OpenAI, 2025c), GPT-5 (OpenAI, 2025b), Gemini-2.0-Flash (Hassabis and Kavukcuoglu, 2024), Gemini-2.5-Flash (Gemini Team, 2025), and Gemini-2.5-Pro (Gemini Team, 2025). Due to cost and resource limitations, we evaluate Gemini-2.5-Flash, Gemini-2.5-Pro, GPT-4.1, GPT-5, and o3 on ROTBENCH-SMALL. Responses are obtained through greedy decoding, while all chain-of-thought (Wei et al., 2022) responses are obtained with a temperature of 0.3.[3] In addition, we provide further results on Llama-3.2-11B-Instruct (Grattafiori et al., 2024), Qwen-2.5-VL-3B-Instruct (Bai et al., 2025), Phi-4-MM-Instruct (Microsoft et al., 2025), Gemma-3-12B-Instruct (Gemma et al., 2025), Molmo-7B-O (Deitke et al., 2024b), and Claude-4.1-Opus (Anthropic, 2025) in Appendix E.

4.2 Setup and Evaluation

We rotate each image in ROTBENCH-LARGE and ROTBENCH-SMALL by 0°, 90°, 180°, and 270° counter-clockwise, resulting in a total of 1000 and 200 images. Note that a 90° counter-clockwise rotation is equivalent to a 270° clockwise rotation. For each image and orientation, we provide the model with the image, a brief description of the task, and various forms of auxiliary information (Section 4.3). We frame this task as a four-way classification problem. To ensure robustness, the mapping between letter choice and degree of rotation is randomized for each prompt. We evaluate models on ROTBENCH-LARGE and report average accuracy and standard deviation across 3 runs in Table 1. We use the same procedure, albeit with only 2 runs, on ROTBENCH-SMALL and report results in Table 2. All prompts used are given in Appendix Q.

[2]The human evaluators all exhibit high accuracy, averaging > 0.97 for all rotations.

[3]We perform an ablation study where we vary the sampling temperature (Table 9). We use the default sampling temperature for proprietary models that do not expose a temperature parameter in their API interface.

We also present evaluations under a regression formulation and a multi-choice setup with increased granularity in Appendix K, all leading to inferior performance. We select our current setup for its simplicity and because it already challenges frontier models."

4.3 Auxiliary Information

Figure 6 illustrates all forms of auxiliary information provided to the model. Note that all auxiliary information is separately extracted for each rotation, ensuring our approach does not depend on prior knowledge of the image's orientation.

Captions. For each image and rotation, we instruct GPT-4o to provide a detailed caption (Appendix Q.2). We emphasize that each image is captioned four times, once per rotation.

Bounding Boxes. For each image and rotation, we first use GPT-4o to extract the primary subjects within the image (Appendix Q.1). Next, along with the image, the list of subjects is given to GroundingDINO (Liu et al., 2023b) to extract a set of normalized coordinates for each subject,[4] which is directly injected into the prompt.

Scene Graphs. A scene graph (Zhu et al., 2022) codifies relationships between objects in an image as a three-element tuple [object 1, predicate, object 2]. Using the extracted subjects from the previous section, we prompt GPT-4o to generate a scene graph for the image.

Depth Maps. We obtain depth maps for each image using ZoeDepth (Bhat et al., 2023). Rather than rotating the depth map obtained from 0°, we separately obtain depth maps for all four rotations.

Segmentation Maps. Using the previously extracted bounding boxes, we obtain a segmentation map of each image and orientation using SAM 2 (Ravi et al., 2024).

Chain-of-Thought. To evaluate whether our multiple-choice setup is hampering performance on this task, we modify the prompt to encourage the model to produce reasoning chains instead of a single letter choice.

Rotation Grid. We test if explicitly allowing models to "visualize" rotations aid performance by providing the input image along with three copies of the image further rotated 90°, 180°, 270°. We compose these four images into a single *rotation*

[4]Each set of coordinates is a four-element tuple, composed of $[x_min, y_min, x_max, y_max]$.

grid. Each image is captioned with the degree of further rotation, independent of the ground truth rotation. We provide a further experiment (*rotation grid guided*) where we explicitly prompt the model to identify an "anchor" image and algebraically calculate the original image's ground truth rotation. All rotation grid experiments use CoT prompting.

5 Main Results

Table 1 displays the results of evaluating Qwen-2.5-VL-7B-Instruct, Gemini-2.0-Flash, and GPT-4o along with various auxiliary information on ROTBENCH-LARGE. Table 2 displays results of evaluating Qwen-2.5-VL-7B-Instruct, GPT-4o, GPT-4.1, GPT-5, o3, Gemini-2.0-Flash, Gemini-2.5-Flash, Gemini-2.5-Pro on ROTBENCH-SMALL. Due to the high token cost of proprietary reasoning models, we only evaluate zero-shot and CoT prompts. However, we evaluate providing rotation grids to o3 and Gemini-2.5-Pro.[5] We provide qualitative examples of images that models answer correct and incorrectly in Appendix O.

MLLMs accurately identify right-side-up (0°) images. All evaluated models effectively recognize right-side-up (0°) images. Qwen-2.5-VL-7B-Instruct (Qwen) achieves an accuracy of 0.99 without supplemental data. Proprietary models (GPT-4o, GPT-4.1, o3, Gemini-2.5-Flash, Gemini-2.5-Pro, and Gemini-2.0-Flash) consistently exhibit near-perfect accuracy on identifying unrotated images. This outcome aligns with expectations, given these models likely encountered predominantly upright images during training, and thus 0° can be assumed to be the *default* option.

Proprietary models perform well on upside-down (180°) images. All models except Qwen demonstrate robust performance on images rotated 180°, with GPT-4o, GPT-4.1, o3, and Gemini-2.5-Pro all achieving accuracies notably above chance (> 0.7). However, Gemini-2.0-Flash and Gemini-2.5-Flash display relatively lower zero-shot performance, with accuracies around 0.5. This indicates that state-of-the-art proprietary models generally possess a reliable capability to recognize upside-down images, though there remains variability within the different model families.

[5]Table 5 further shows results obtained from evaluating these two models on ROTBENCH-SMALL with all available auxiliary information.

Identifying 90° and 270° is challenging for all evaluated models. All models exhibit substantial difficulties when distinguishing between 90° and 270°, while the poorest performance consistently emerged with 270° images. Confusion matrix analysis (Section 6.1) reveals frequent misclassifications between these two orientations, indicating a distinct challenge when identifying 0° and 180°.

Providing auxiliary inputs does not reliably improve performance. None of the auxiliary information results in meaningful, consistent performance gains across all tested models. Paradoxically, the introduction of additional information sometimes leads to marginal performance degradation. When including all forms of auxiliary information, Qwen's accuracy on 90° decreased from 0.51 to 0.26. Similarly, Gemini-2.0-Flash's accuracy on 270° decreased from 0.44 to 0.17. We further examine why scene graphs fails to consistently improve model performance in Appendix J.

Rotation grid improves performance only for reasoning models. Providing the rotation grid degraded performance on most models. Gemini-2.0-Flash accuracy on 90° decreased by nearly 0.5 compared to CoT prompting. Rotation grid guided, however, improved GPT-4o accuracy on 270° by around 0.1. At the same time, both o3 and Gemini-2.5-Pro saw performance improvements. Notably, Gemini-2.5-Pro's accuracy on 90° and 270° improved by 0.15. These results suggest the two reasoning models are much more effective at utilizing visual context. We further leverage the robust identification of 0° images and modify the rotation grid into a majority voting approach Appendix M. This setup results in gains even in weaker models, indicating MLLMs can achieve moderate performance with sufficient visual scaffolding.

CoT improves 180° performance, but results are mixed on 90° and 270°. Employing chain-of-thought (CoT) prompting yields mixed results, differing notably across models and orientations. Gemini models show improved accuracy on 90° rotations but decreased accuracy for 270°. On the other hand, GPT models displayed inverse trends. Yet, CoT consistently enhanced accuracy for 180° rotations across all models tested. These findings suggest CoT prompting can help models better reason about rotations to some extent, but does not universally resolve inherent challenges in distinguishing portrait orientations. Appendix H closely

Model	Accuracy on Different Degrees of Rotation			
	0°	90°	180°	270°
Qwen-2.5-VL-7B-Instruct				
Zero-shot	0.99±0.00	0.51±0.01	0.05±0.01	0.09±0.01
+ Caption	**1.00**±0.00	0.51±0.01	0.23±0.01	0.07±0.00
+ Bounding Box	0.90±0.00	0.48±0.01	0.01±0.00	0.11±0.00
+ Scene Graph	0.97±0.01	0.51±0.01	0.01±0.01	0.11±0.02
+ Depth Map	0.93±0.01	0.55±0.02	0.04±0.01	0.13±0.02
+ Segmentation Map	0.81±0.01	**0.63**±0.02	0.03±0.02	0.16±0.01
+ Chain-of-Thought	0.88±0.01	0.26±0.02	**0.34**±0.01	0.23±0.02
+ Rotation Grid	0.57±0.04	0.15±0.02	0.13±0.01	0.28±0.00
+ Rotation Grid Guided	0.59±0.01	0.12±0.01	0.13±0.00	0.30±0.02
+ *all above*	0.47±0.03	0.26±0.01	0.17±0.01	**0.33**±0.02
Gemini-2.5-Flash				
Zero-shot	**1.00**±0.00	0.30±0.00	0.72±0.00	0.44±0.01
+ Caption	**1.00**±0.00	0.38±0.00	**0.76**±0.00	0.44±0.01
+ Bounding Box	**1.00**±0.00	0.43±0.03	0.71±0.00	0.34±0.02
+ Scene Graph	**1.00**±0.00	0.41±0.00	0.71±0.00	0.33±0.02
+ Depth Map	**1.00**±0.00	0.30±0.01	0.69±0.01	**0.46**±0.01
+ Segmentation Map	**1.00**±0.00	0.28±0.01	0.73±0.02	0.45±0.00
+ Chain-of-Thought	**1.00**±0.00	**0.63**±0.00	**0.76**±0.01	0.19±0.1
+ Rotation Grid	**1.00**±0.00	0.07±0.01	0.57±0.01	0.07±0.01
+ Rotation Grid Guided	**1.00**±0.00	0.10±0.00	0.61±0.00	0.25±0.01
+ *all above*	**1.00**±0.00	0.1±0.01	0.67±0.01	0.17±0.01
GPT-4o				
Zero-shot	0.99±0.00	0.69±0.02	0.93±0.00	0.19±0.01
+ Caption	0.98±0.00	0.65±0.00	0.93±0.01	0.23±0.02
+ Bounding Box	0.98±0.01	0.59±0.01	0.91±0.00	0.31±0.04
+ Scene Graph	0.98±0.00	0.55±0.00	0.93±0.00	0.33±0.02
+ Depth Map	**1.00**±0.00	0.55±0.03	0.93±0.00	0.26±0.01
+ Segmentation Map	0.97±0.00	0.67±0.01	**0.95**±0.00	0.21±0.00
+ Chain-of-Thought	0.97±0.01	0.57±0.03	0.93±0.00	0.32±0.00
+ Rotation Grid	0.98±0.00	**0.71**±0.02	0.93±0.00	0.19±0.03
+ Rotation Grid Guided	0.98±0.00	0.46±0.03	0.93±0.00	**0.41**±0.00
+ *all above*	**1.00**±0.00	0.46±0.03	0.91±0.00	0.36±0.02

Table 1: Classification accuracy using various forms of auxiliary information and prompting for all rotations. All results are obtained from three runs on ROTBENCH-LARGE. Accuracy is scored on a four-way classification task.

examines a common failure of CoT.

6 Additional Analyses

6.1 Model Bias Towards 0° and 90°

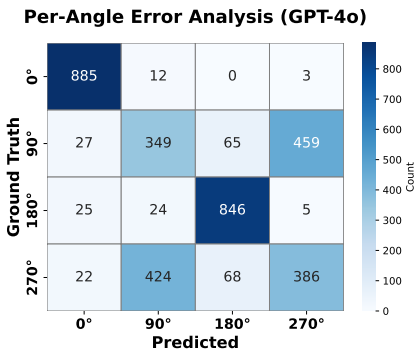

Figure 3: Confusion matrix of true vs. predicted rotations for GPT-4o using CoT prompting, summed across three runs on ROTBENCH-LARGE. Rows represent ground-truth labels, columns represent predicted labels.

To further elucidate the specific types of rotational errors made by models, we analyze the confusion matrix for GPT-4o. Figure 3 shows the confusion matrix obtained from summing predictions across three runs. We see GPT-4o predominantly struggles with differentiating between 90° and 270° rotations. Specifically, the model misclassifies 459 instances of 90° images as 270° and 424 instances of 270° images as 90°. Yet, 0° and 180° show significantly fewer misclassifications. This analysis underscores a critical shortcoming in model performance, suggesting either that the vision encoder is not providing sufficient signals to distinguish between clockwise and counter-clockwise rotations, or that the MLLM is not adequately incorporating the visual information into its reasoning.

Model	Accuracy on Different Degrees of Rotation			
	0°	90°	180°	270°
Human				
Zero-shot	0.99	0.99	0.99	0.97
Qwen-2.5-VL-7B-Instruct				
Zero-shot	**0.95**$_{\pm 0.01}$	**0.57**$_{\pm 0.04}$	0.03$_{\pm 0.02}$	0.15$_{\pm 0.05}$
+ Chain-of-Thought	0.86$_{\pm 0.00}$	0.20$_{\pm 0.04}$	**0.13**$_{\pm 0.03}$	**0.30**$_{\pm 0.04}$
GPT-4o				
Zero-shot	0.87$_{\pm 0.02}$	**0.65**$_{\pm 0.04}$	0.85$_{\pm 0.01}$	**0.21**$_{\pm 0.03}$
+ Chain-of-Thought	**0.91**$_{\pm 0.01}$	0.59$_{\pm 0.01}$	**0.86**$_{\pm 0.00}$	**0.21**$_{\pm 0.05}$
GPT-4.1				
Zero-shot	0.95$_{\pm 0.01}$	0.63$_{\pm 0.07}$	**0.85**$_{\pm 0.03}$	**0.19**$_{\pm 0.09}$
+ Chain-of-Thought	**0.98**$_{\pm 0.00}$	**0.88**$_{\pm 0.00}$	**0.85**$_{\pm 0.01}$	0.03$_{\pm 0.03}$
GPT-5				
Zero-shot	**1.00**$_{\pm 0.00}$	0.41$_{\pm 0.03}$	**0.81**$_{\pm 0.01}$	**0.59**$_{\pm 0.04}$
+ Chain-of-Thought	**1.00**$_{\pm 0.00}$	**0.53**$_{\pm 0.05}$	**0.81**$_{\pm 0.02}$	0.57$_{\pm 0.07}$
Gemini-2.0-Flash				
Zero-shot	**1.00**$_{\pm 0.00}$	0.25$_{\pm 0.04}$	0.48$_{\pm 0.01}$	**0.43**$_{\pm 0.07}$
+ Chain-of-Thought	**1.00**$_{\pm 0.00}$	**0.61**$_{\pm 0.01}$	**0.59**$_{\pm 0.01}$	0.25$_{\pm 0.01}$
Gemini-2.5-Flash				
Zero-shot	**1.00**$_{\pm 0.00}$	**0.23**$_{\pm 0.06}$	**0.50**$_{\pm 0.03}$	**0.47**$_{\pm 0.02}$
+ Chain-of-Thought	**1.00**$_{\pm 0.00}$	**0.23**$_{\pm 0.01}$	0.44$_{\pm 0.02}$	0.40$_{\pm 0.02}$
o3				
Zero-shot	**1.00**$_{\pm 0.00}$	**0.45**$_{\pm 0.04}$	0.70$_{\pm 0.03}$	0.48$_{\pm 0.03}$
+ Chain-of-Thought	**1.00**$_{\pm 0.00}$	0.36$_{\pm 0.00}$	0.74$_{\pm 0.04}$	0.57$_{\pm 0.01}$
+ Rotation Grid	0.99$_{\pm 0.01}$	0.23$_{\pm 0.05}$	**0.83**$_{\pm 0.05}$	**0.81**$_{\pm 0.01}$
+ Rotation Grid Guided	0.99$_{\pm 0.01}$	0.31$_{\pm 0.01}$	0.82$_{\pm 0.02}$	0.75$_{\pm 0.01}$
Gemini-2.5-Pro				
Zero-shot	**1.00**$_{\pm 0.00}$	0.50$_{\pm 0.04}$	0.72$_{\pm 0.00}$	0.40$_{\pm 0.06}$
+ Chain-of-Thought	**1.00**$_{\pm 0.00}$	0.46$_{\pm 0.02}$	0.71$_{\pm 0.01}$	0.49$_{\pm 0.01}$
+ Rotation Grid	0.99$_{\pm 0.01}$	0.58$_{\pm 0.00}$	0.67$_{\pm 0.03}$	0.59$_{\pm 0.03}$
+ Rotation Grid Guided	0.95$_{\pm 0.01}$	**0.71**$_{\pm 0.01}$	**0.73**$_{\pm 0.03}$	**0.74**$_{\pm 0.04}$

Table 2: Classification accuracy of different image rotations for various models across two runs on ROTBENCH-SMALL using zero-shot or chain-of-thought prompting. We also show results from o3 and Gemini-2.5-Pro using rotation grids. Accuracy is scored on a four-way classification task.

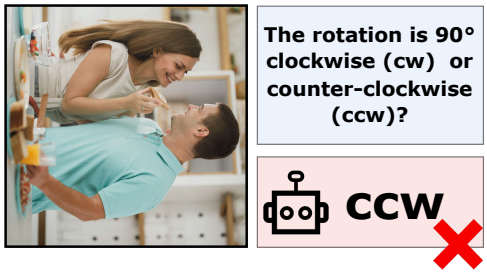

Figure 4: GPT-4o answers incorrectly when asked to identify whether the image has been rotated 90° clockwise or counter-clockwise.

6.2 Distinguishing Clockwise from Counter-clockwise Rotations

Setup. Following the analysis in Section 6.1, we examine whether MLLMs can distinguish clockwise (CW) and counter-clockwise rotations (CCW) by simplifying the four-way classification task into a binary classification task. Using the 90° and 270° images from ROTBENCH-LARGE, we prompt MLLMs to determine whether an image has been rotated 90° CCW or CW.[6] By asking models to explicitly differentiate the two directions, we hope to provide a clearer signal for directional understanding. Figure 4 shows an example question where the GPT-4o provides an incorrect answer.

Results. Table 3 tests CW vs. CCW classification on GPT-4o and Qwen-2.5-VL-7B-Instruct (we additionally provide similar smaller-scale tests on ROTBENCH-SMALL for GPT-5 in Appendix G). Table 3 indicates that GPT-4o has a significant bias toward identifying rotations as 90° counter-clockwise. GPT-4o correctly identified only 52 out of 300 clockwise rotations, whereas Qwen-2.5-VL-7B-Instruct correctly identified only 23 out of 300

[6]Note that 90° CW is equivalent to a 270° CCW.

GT \ Predicted	Counter-clockwise	Clockwise
GPT-4o		
Counter-clockwise	248	52
Clockwise	259	41
Qwen-2.5-VL-7B-Instruct		
Counter-clockwise	270	30
Clockwise	277	23

Table 3: Accuracy of different models in identifying 90° clockwise (CW) versus counter-clockwise rotation (CCW). Column indicates ground truth rotation (GT). A small number of responses are incorrectly identified as being right-side up. These responses are excluded in the table's statistics.

clockwise rotations. The models consistently default to labeling ambiguous or uncertain rotations as counter-clockwise, hinting towards a potential underlying perceptual bias (Appendix H). While we find that the most recent model tested, GPT-5, has less bias and is better able to identify clockwise rotations, this improvement is not reflected in Table 2, where the model performs on par with others; in other words, poor performance on ROTBENCH cannot be solely explained by a model's ability to distinguish clockwise from counterclockwise rotation. See Appendix G for further discussion.

These findings strongly indicate that the MLLMs tested have limitations in reliably distinguishing between CW and CCW rotational directions, offering a potential explanation for why distinguishing 90° and 270° CCW rotations presents such a challenging task. Further corroborating this idea, we show that evaluating Qwen-2.5-VL-7B-Instruct, GPT-4o, o3 on ROTBENCH-SMALL using clockwise angles does not result in significant performance differences (Appendix F).

6.3 Can Fine-tuning Solve ROTBENCH?

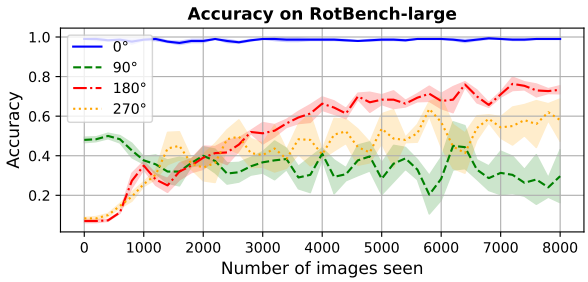

Figure 5: Qwen-2.5-VL-7B-Instruct's accuracy on different degrees of rotation as training progresses.

Setup. To assess whether specialized training can mitigate these performance issues, we conduct fine-tuning experiments using Qwen-2.5-VL-7B-Instruct. During data filtering, we find some images in Spatial-MM closely resemble each other (Fig. 11). If we fine-tune on images in Spatial-MM not selected for ROTBENCH, the existence of similar images in the training and testing sets may lead to inflated performance. To prevent such instances of overfitting, we train on 1000 images from MS COCO (Lin et al., 2015) and evaluate performance on ROTBENCH-LARGE. Appendix D provides further training details.

Results. Fig. 5 reveals a high and consistent accuracy for 0° throughout training, indicating robust recognition of upright images. Performance on 180° gradually improves, stabilizing around 0.8 after approximately 7000 images. However, accuracies for 90° and 270° exhibit substantial oscillations, suggesting that the model did not achieve stable improvement. The model appears to be caught in a cycle, alternating between accuracy gains and losses for these two rotations. This phenomenon is also reflected in our main results (Section 5), where using CoT prompting improved accuracy on 270° images at the expense of 90°. The unstable performance may result from potential representational constraints in current visual encoders that limit visual understanding capabilities, particularly regarding subtle rotational distinctions.

7 Conclusion

We evaluate whether MLLMs are able to identify input images rotated 0°, 90°, 180°, and 270° using ROTBENCH, a 350-image manually-filtered benchmark. Our results reveal that state-of-the-art MLLMs reliably identify images that are unrotated (0°) or upside-down (180°), but struggle with 90° and 270° rotations. Auxiliary information and chain-of-thought prompting provide limited improvements. Simultaneously providing all possible rotations improves performance only for frontier reasoning models, but a modified setup using majority voting improves performance for weaker models as well. Fine-tuning Qwen-2.5-VL-7B-Instruct shows that performance oscillates between 90° and 270°, suggesting the presence of two local optima. These results indicate a potential blind spot in MLLMs' spatial reasoning capabilities, motivating future directions in improving orientational understanding.

8 Limitations

Our work shows MLLMs struggle to accurately identify the rotation of input images rotated (0°, 90°, 180°, and 270°). In real-world scenarios, images are typically rotated by arbitrary angles. Therefore, our work may in fact overestimate model capabilities. Moreover, we did not evaluate larger open-weight models due to resource and hardware limitations. Rather, we evaluated the most recent and largest proprietary MLLMs through their APIs. Since these proprietary models dominate open-weight models on nearly all benchmarks (OpenAI, 2025c; Gemini Team, 2025), we expect larger open-weight models to perform worse or on par with the top proprietary models tested.

References

Bang An, Sicheng Zhu, Michael-Andrei Panaitescu-Liess, Chaithanya Kumar Mummadi, and Furong Huang. 2023. More context, less distraction: Visual classification by inferring and conditioning on contextual attributes. *CoRR*.

Ahmad Mustafa Anis, Hasnain Ali, and Saquib Sarfraz. 2025. On the limitations of vision-language models in understanding image transforms. *Preprint*, arXiv:2503.09837.

Anthropic. 2025. Claude 4.1 system card. Technical report, Anthropic. Accessed October 2025.

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, and 8 others. 2025. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. 2023. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint*.

Weijing Chen, Linli Yao, and Qin Jin. 2023a. Rethinking benchmarks for cross-modal image-text retrieval. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 1241–1251. ACM.

Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, Alexander Kolesnikov, Joan Puigcerver, Nan Ding, Keran Rong, Hassan Akbari, Gaurav Mishra, Linting Xue, Ashish Thapliyal, James Bradbury, and 10 others. 2023b. Pali: A jointly-scaled multilingual language-image model. *Preprint*, arXiv:2209.06794.

Taco S. Cohen and Max Welling. 2016. Group equivariant convolutional networks. *Preprint*, arXiv:1602.07576.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom, Kiana Ehsani, Huong Ngo, YenSung Chen, Ajay Patel, Mark Yatskar, Chris Callison-Burch, and 31 others. 2024a. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. *Preprint*, arXiv:2409.17146.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, Jiasen Lu, Taira Anderson, Erin Bransom, Kiana Ehsani, Huong Ngo, YenSung Chen, Ajay Patel, Mark Yatskar, Chris Callison-Burch, and 31 others. 2024b. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. *Preprint*, arXiv:2409.17146.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: efficient finetuning of quantized llms. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Zeyu Feng, Chang Xu, and Dacheng Tao. 2019. Self-supervised representation learning by rotation feature decoupling. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10356–10366.

Philipp Fischer, Alexey Dosovitskiy, and Thomas Brox. 2015. Image orientation estimation with convolutional networks. In *German Conference on Pattern Recognition*.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, Yunsheng Wu, and Rongrong Ji. 2024a. Mme: A comprehensive evaluation benchmark for multimodal large language models. *Preprint*, arXiv:2306.13394.

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

Google Gemini Team. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. https://storage.googleapis.com/deepmind-media/gemini/gemini_v2_5_report.pdf.

Gemma, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey

Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Demis Hassabis and Koray Kavukcuoglu. 2024. Introducing gemini 2.0: our new ai model for the agentic era. https://blog.google/technology/google-deepmind/google-gemini-ai-update-december-2024/.

Dan Hendrycks and Thomas Dietterich. 2019. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Ujash Joshi and Michael Guerzhoy. 2017. Automatic photo orientation detection with convolutional neural networks. In *2017 14th Conference on Computer and Robot Vision (CRV)*, page 103–108. IEEE.

Amita Kamath, Jack Hessel, and Kai-Wei Chang. 2023. What's "up" with vision-language models? investigating their struggle with spatial reasoning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9161–9175, Singapore. Association for Computational Linguistics.

Jinwoo Lee, Hyunsung Go, Hyunjoon Lee, Sunghyun Cho, Minhyuk Sung, and Junho Kim. 2021. Ctrl-c: Camera calibration transformer with line-classification. *Preprint*, arXiv:2109.02259.

Jinwoo Lee, Minhyuk Sung, Hyunjoon Lee, and Junho Kim. 2020. Neural geometric parser for single image camera calibration. *Preprint*, arXiv:2007.11855.

Jongmin Lee, Byungjin Kim, Seungwook Kim, and Minsu Cho. 2023. Learning rotation-equivariant features for visual correspondence. *Preprint*, arXiv:2303.15472.

Zongxia Li, Xiyang Wu, Hongyang Du, Fuxiao Liu, Huy Nghiem, and Guangyao Shi. 2025. A survey of state of the art large vision language models: Alignment, benchmark, evaluations and challenges. *Preprint*, arXiv:2501.02189.

Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2015. Microsoft coco: Common objects in context. *Preprint*, arXiv:1405.0312.

Zhiqiu Lin, Jin Sun, Abe Davis, and Noah Snavely. 2020. Visual chirality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12295–12303.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023a. Visual instruction tuning. *Preprint*, arXiv:2304.08485.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and 1 others. 2023b. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*.

Microsoft, :, Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, Dong Chen, Dongdong Chen, Junkun Chen, Weizhu Chen, Yen-Chun Chen, Yi ling Chen, Qi Dai, and 57 others. 2025. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras. *Preprint*, arXiv:2503.01743.

Agnieszka Mikołajczyk and Michał Grochowski. 2018. Data augmentation for improving deep learning in image classification problem. In *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, pages 117–122.

OpenAI. 2025a. Introducing GPT-4.1 in the api. https://openai.com/index/gpt-4-1/.

OpenAI. 2025b. Introducing gpt-5. https://openai.com/index/introducing-gpt-5/.

OpenAI. 2025c. Introducing OpenAI o3 and o4-mini. https://openai.com/index/introducing-o3-and-o4-mini/.

OpenAI, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, and 400 others. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Luis Perez and Jason Wang. 2017. The effectiveness of data augmentation in image classification using deep learning. *Preprint*, arXiv:1712.04621.

Atin Pothiraj, Elias Stengel-Eskin, Jaemin Cho, and Mohit Bansal. 2025. Capture: Evaluating spatial reasoning in vision language models via occluded object counting. *Preprint*, arXiv:2504.15485.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. *Preprint*, arXiv:2103.00020.

Cyrus Rashtchian, Charles Herrmann, Chun-Sung Ferng, Ayan Chakrabarti, Dilip Krishnan, Deqing Sun, Da-Cheng Juan, and Andrew Tomkins. 2023. Substance or style: What does your image embedding know? *arXiv preprint arXiv:2307.05610*.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. 2024. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*.

R N Shepard and J Metzler. 1971. Mental rotation of three-dimensional objects. *Science*, 171(3972):701–703.

Fatemeh Shiri, Xiao-Yu Guo, Mona Golestan Far, Xin Yu, Reza Haf, and Yuan-Fang Li. 2024. An empirical analysis on spatial reasoning capabilities of large multimodal models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21440–21455, Miami, Florida, USA. Association for Computational Linguistics.

Connor Shorten and Taghi M. Khoshgoftaar. 2019. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1).

Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. 2024. Eyes wide shut? exploring the visual shortcomings of multimodal llms. *Preprint*, arXiv:2401.06209.

Vishaal Udandarao, Max F Burg, Samuel Albanie, and Matthias Bethge. 2024. Visual data-type understanding does not emerge from scaling vision-language models. In *The Twelfth International Conference on Learning Representations*.

Muhammad Usama, Syeda Aishah Asim, Syed Bilal Ali, Syed Talal Wasim, and Umair Bin Mansoor. 2025. Analysing the robustness of vision-language-models to common corruptions. *Preprint*, arXiv:2504.13690.

S G Vandenberg and A R Kuse. 1978. Mental rotations, a group test of three-dimensional spatial visualization. *Percept. Mot. Skills*, 47(2):599–604.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Wenqi Xian, Zhengqi Li, Matthew Fisher, Jonathan Eisenmann, Eli Shechtman, and Noah Snavely. 2019. Uprightnet: Geometry-aware camera orientation estimation from single images. *Preprint*, arXiv:1908.07070.

Renjun Xu, Kaifan Yang, Ke Liu, and Fengxiang He. 2023. $e(2)$-equivariant vision transformer. *Preprint*, arXiv:2306.06722.

Ruijie Xu, Yong Shi, and Zhiquan Qi. 2024. Image orientation estimation based on deep learning - a survey. *Procedia Computer Science*, 242:1193–1197. 11th International Conference on Information Technology and Quantitative Management (ITQM 2024).

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. *Preprint*, arXiv:2303.15343.

Yikang Zhou, Tao Zhang, Shilin Xu, Shihao Chen, Qianyu Zhou, Yunhai Tong, Shunping Ji, Jiangning Zhang, Lu Qi, and Xiangtai Li. 2025. Are they the same? exploring visual correspondence shortcomings of multimodal llms. *Preprint*, arXiv:2501.04670.

Guangming Zhu, Liang Zhang, Youliang Jiang, Yixuan Dang, Haoran Hou, Peiyi Shen, Mingtao Feng, Xia Zhao, Qiguang Miao, Syed Afaq Ali Shah, and Mohammed Bennamoun. 2022. Scene graph generation: A comprehensive survey. *Preprint*, arXiv:2201.00443.

Appendix

A Dataset Details

A.1 Further details on Stage 1 and 2 Filtering

Stage 1. We randomly sample 300 images from Spatial-MM (Shiri et al., 2024). Spatial-MM is divided into two splits, *one-subject* and *two-subject*, respectively comprised of images that contain one and two primary subjects. As models may use image dimensions to infer orientation, we crop each image into a square before further processing. We sample 150 images from each split and show the resulting 300-image dataset to the Stage 1 annotator. For each image, the annotator can decide to either accept, discard, or flag. The annotator accepts an image if it (1) contains easily identifiable rotational visual cues (e.g., a person standing), and (2) the image has meaningful differences when rotated. Images that do not satisfy these criteria are discarded. Occasionally, some images have subtle visual cues, or require more detailed semantic understanding to correctly perceive orientation. These images tend to have primary subjects that do not vary significantly with rotation, or require

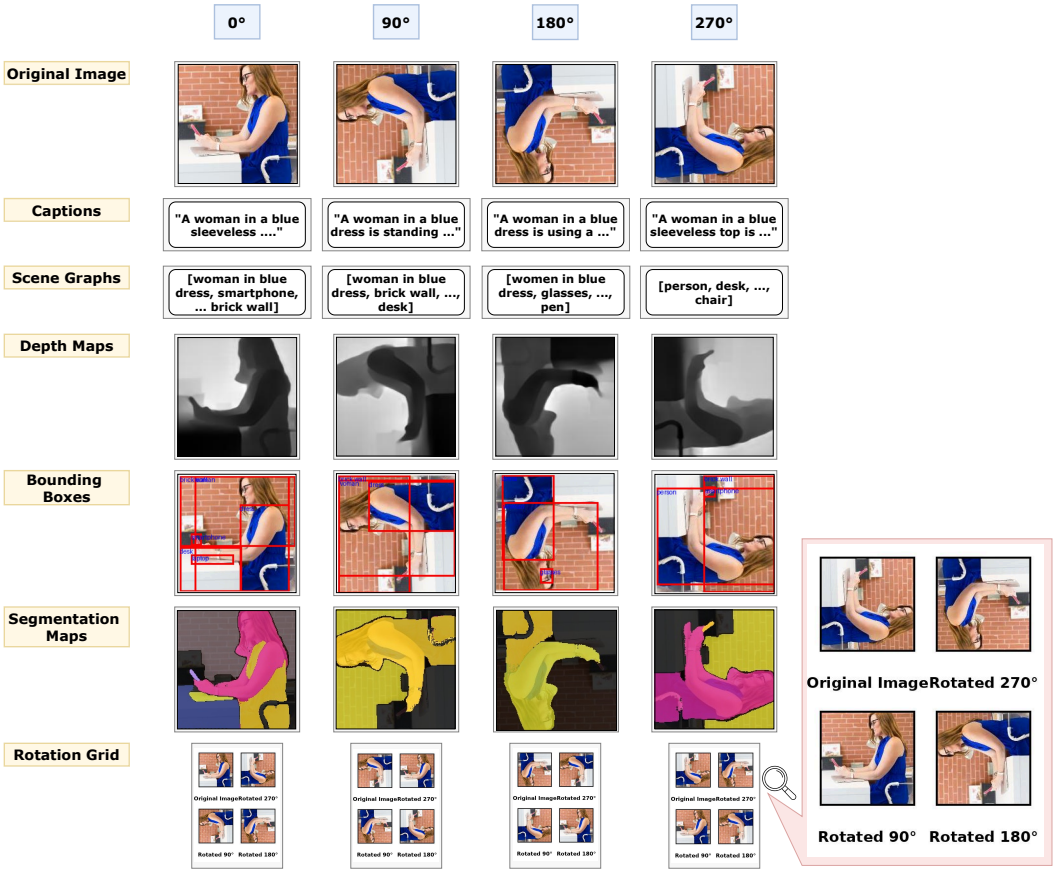

Figure 6: Examples of the different types of auxiliary information provided to the models.

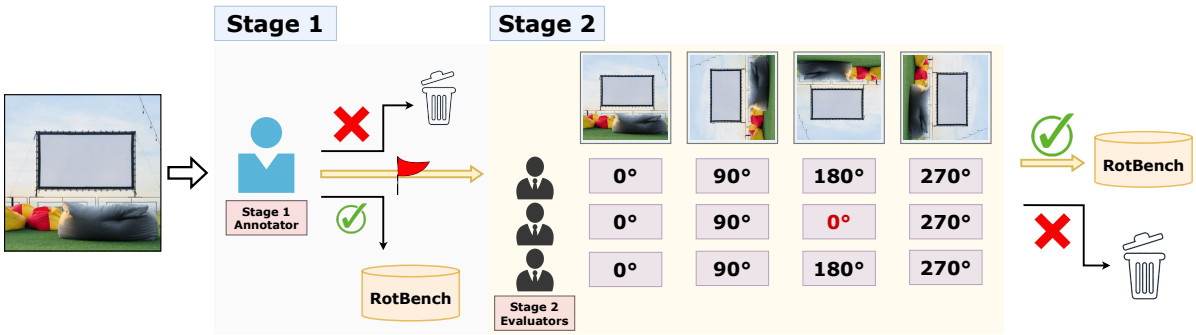

Figure 7: Figure describes our two-stage data filtering procedure. The example image is flagged during Stage 1, but subsequently accepted during Stage 2 as only one evaluator provided an incorrect response.

integrating background signals to identify rotation. The sample image shown in Fig. 7 is an example of a flagged image. Such images are flagged by the Stage 1 annotator to proceed to Stage 2. We provide further examples and details of accepted, flagged, and discarded images in Fig. 9. Note that the Stage 1 annotator is also tasked with ensuring no leaked personal information or offensive content is incorporated into ROTBENCH.

Stage 2. Stage 2 involves a group of three human evaluators. Each flagged image is rotated 0°, 90°, 180°, and 270°, producing four images. We shuffle all images – which now include rotated images – and present them to the evaluators as multiple-choice questions (Appendix A.4). Any image that elicits an incorrect answer from two or more evaluators across all four orientations is discarded. Otherwise, the image is accepted.

Of the first 300 sampled images, only 27 were flagged. Among the flagged images, only two are eventually rejected. We sample two additional images to replace the discarded images. These images form the 300-image ROTBENCH-LARGE benchmark.

A.2 Creating ROTBENCH-SMALL

As each image is rotated in four orientations (0°, 90°, 180°, and 270°), obtaining human performance on ROTBENCH-LARGE is costly. We therefore propose ROTBENCH-SMALL, a human-evaluated 50-image subset where we can establish a human baseline. Starting from the 25 non-discarded flagged images in Stage 2, we obtain ROTBENCH-SMALL by further sampling 25 more images from Spatial-MM that fit the criteria for flagging. We ensure an equal number of one and two primary subject images in the final 50-image dataset. We then repeat the Stage 2 procedure on this new set, obtaining a human baseline. Notably, none of the 25 new images were discarded from human evaluation.

A.3 Annotator Information

All three evaluators of Stage 2 data filtering are university students majoring in Computer Science, and have prior experience in Artificial Intelligence research. Two evaluators are undergraduate students who have volunteered to participate in the project, the third is a graduate student who also provided Stage 1 annotations and is an author of this work. All evaluators consented to have their data incorporated.

A.4 Annotator Interface

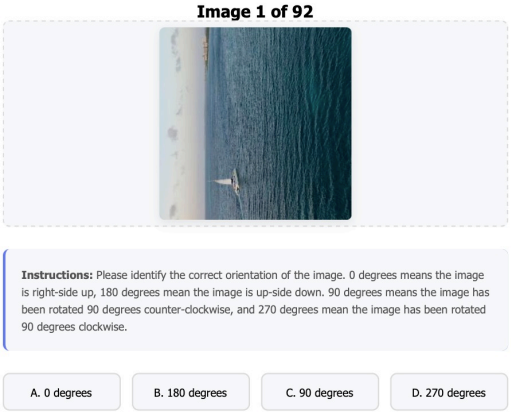

Figure 8: A screenshot of the custom interface shown to Stage 2 annotators.

Figure 8 shows the interface provided to Stage 2 annotators. The mapping between letter choice and rotation degree is shuffled from question to question.

A.5 Sample Images

Figure 6 displays the various forms of additional information provided to the model. Figure 9 provides examples of an image that is accepted, flagged, and discarded during Stage 1 of data filtering. The accepted image has a clear subject and is easily distinguishable when rotated 0°, 90°, 180°, and 270°. The flagged image is more difficult to identify, however, the slight tilt – as opposed to a directly top-down perspective – still enables accurate judgment of rotation. However, the discarded image has no meaningful signals to distinguish between the four orientations.

A.6 Further details on flagged images

Flagged images typically involve cases where there is a lack of primary subjects or the primary subject does not offer any distinguishable signals under rotation. These images tend to require deeper comprehension in order to make an accurate classification. For instance, Fig. 7 displays a blank projector canvas above some colorful couches on a patio. The orientation of a projector screen cannot be reliably identified by where its supporting structures are located – the screen can be hung (cable hangs down), pitched up (supporting stand on the ground), or attached to a wall (support frame located sideways of the screen). Therefore, correctly identifying the orientation of the image requires understanding the

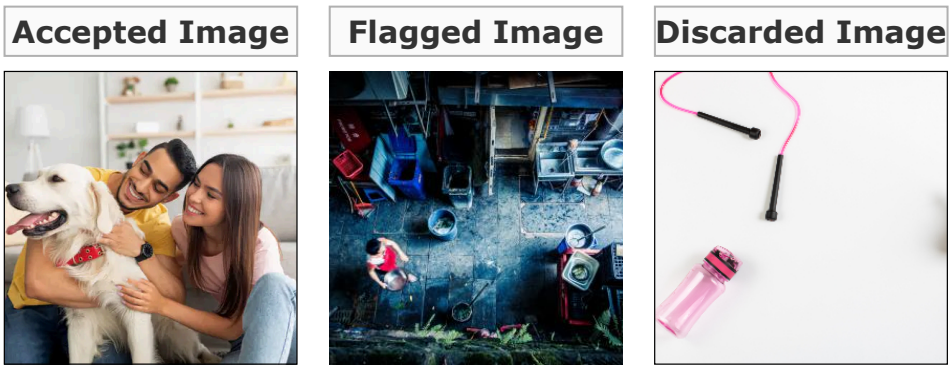

Figure 9: Samples of accepted, flagged, and discarded images during Stage 1 of data filtering.

correct orientation of the couches.

A.7 Motivation for Choosing Spatial-MM

There are multiple ways a model may reason about rotation. If an image has only one primary subject, rotation must be identified using the subject itself and background information. However, if there are multiple primary subjects, the spatial relationship between the two can also provide valuable information. We wish to curate a dataset that balances these two forms of reasoning. As previously mentioned, Spatial-MM's images already distinguish between these two categories, greatly facilitating this balancing when creating ROTBENCH. Furthermore, Spatial-MM is composed of images scraped from the internet, covering a wide range of lifestyle, portrait, and landscape images. This diversity leads to better generalization to other real-world scenarios.

B Other Relevant Work

Camera orientation estimation. Camera orientation estimation is also a well-studied task in computer vision (Xian et al., 2019). Instead of predicting the rotation of the image, camera orientation estimation seeks to predict the spatial location of the camera when capturing an image. Contemporary approaches of this task use deep networks to directly predict orientation parameters from image features in an end-to-end manner (Xian et al., 2019; Lee et al., 2021, 2020).

Sensitivity to visual perturbations. Previous work has shown that visual encoders and MLLMs are sensitive to simple image transformations. Anis et al. (2025) evaluate CLIP (Radford et al., 2021) and SigLIP (Zhai et al., 2023) on a suite of common image transformations – rotations, flips, noise, etc. – revealing substantial gaps between human and model understanding. Usama et al. (2025) finds MLLMs exhibit distinct failure patterns in scene-text and object reasoning tasks when applying ImageNet-C corruptions (Hendrycks and Dietterich, 2019) to image inputs. Taken together, these studies highlight that despite strong clean-image performance, visual encoders and MLLMs are highly sensitive to photometric and geometric distortions.

Robustness to image transformations. Past work has also examined various methods to ensure that image transformations do not affect downstream task performance. Mikołajczyk and Grochowski (2018), Shorten and Khoshgoftaar (2019), and Perez and Wang (2017) use image transformations as data augmentation methods to improve downstream classifier robustness. Other works instead proposed alternative architectures and training schemes to improve robustness to rotation (Xu et al., 2023; Cohen and Welling, 2016; Lee et al., 2023; Feng et al., 2019). While this line of work focuses on training models to ignore certain transformations and learn invariant features for downstream tasks, we instead focus on identifying and reasoning about the transformation itself.

Identifying visual perturbations. Past work has tested how well vision encoders and MLLMs can identify which perturbation or transformation was applied to an input image (Lin et al., 2020; An et al., 2023; Rashtchian et al., 2023). Udandarao et al. (2024) show that visual encoders and MLLMs often fail to classify which perturbation out of 27 perturbations (e.g., contrast, brightness, rotation, blur) was applied to an input image on automatically-created datasets. In contrast, we provide a fine-grained analysis of rotation specifically through a 4-way angle classification task on a curated dataset with rigorous human verification, offering novel insights into why state-of-the-art modern MLLMs

struggle with distinguishing specific rotation angles. Moreover, we test providing remedies such as chain-of-thought reasoning, various forms of auxiliary information (e.g., captions, bounding boxes, segmentation maps, depth maps), and guiding reasoning with algebraic calculations. We also analyze whether model performance is dependent on abilities such as identifying clockwise versus counter-clockwise rotation.

C Generalization and Potential Downstream Applications

While our work provides a detailed analysis into the task of identifying image rotation, showing that such a simple and intuitive task for humans remains challenging for frontier reasoning MLLMs. Despite its straightforward nature, image rotation estimation requires strong perception, spatial reasoning abilities, and general world knowledge. For each image, the model must recognize which objects are relevant to rotational understanding, identify their spatial relationships, and determine whether there exists any irregularities in such relationships. Furthermore, computer-vision tasks typically tend to emphasize the primary subject, or subjects at the forefront of an image (e.g., object extraction, counting, segmentation, etc). However, image rotation often requires the model to also interpret subtle background information.

These abilities are a fundamental prerequisite for higher-level spatial reasoning. Our findings expose systematic blind spots which can lead to failures in more complex downstream tasks, such as understanding relative positions ("left/right/above/below") or reasoning under rotated cameras.

In addition, we further elaborate on the two downstream applications outlined in Section 1:

- Robotics: AI tasks in manufacturing may require cameras mounted on robotic arms that can move in multiple axes. Researchers interested in training a robotic arm to perform complex maneuvers – such as orienting a drill bit to reach awkward angles – with a high degree of autonomy, would require a model that can accurately identify the current rotation of the arm.
- First-person Point-of-View (PoV) footage analysis: Analyzing footage from various extreme sports – such as cliff diving or sky diving – and aerial vehicles – such as camera drones – involves complex rotational movement across multiple axes of rotation. Researchers interested in training RL agents in such environments require the agent to exhibit strong spatial reasoning abilities, among which is the prerequisite ability to identify its current rotation from the image.

D Model Inference and Training Details

All proprietary models are accessed through first-party APIs provided in their respective official Python packages; the open-weight models are run on an Nvidia RTX A6000 Ada GPU (48 GB). When fine-tuning Qwen-2.5-VL-7B-Instruct, we employ 4-bit quantization with the Bits-And-Bytes (bnb) NF4 format (Dettmers et al., 2023) and apply Low-Rank Adaptation (LoRA) (Hu et al., 2021) with a rank 8 and an alpha parameter of 32. The fine-tuning procedure ran for 2 epochs, with a batch size of 32 and a learning rate of 2e-5.

E Additional Results

We provide experimental results from evaluating Llama-3.2-11B-Instruct on ROTBENCH-LARGE (Table 4), as well as o3, Gemini-2.5-Pro (Table 5),Qwen-2.5-VL-3B-Instruct, Claude-4.1-Opus, Phi-4-MM-Instruct, Gemma-3-12B-Instruct, and Molmo-7B-O on ROTBENCH-SMALL (Table 6. Llama-3.2-11B-Instruct seems to be the weakest model tested, showing inferior performance across all rotations. Without CoT, the model only achieves roughly 0.3 accuracy. The two frontier reasoning models (o3 and Gemini-2.5-Pro) show no significant performance improvement when prompted with auxiliary information. However, they respond positively to rotation grids. Gemini-2.5-Pro achieves the highest accuracy on 90° and 270° (both 0.7) using guided rotation grids, while o3 achieves its highest accuracy on 180° (0.83) using the rotation grid. These results indicate that strong reasoning models are able to utilize the grid to aid them in making a prediction, while weaker models struggle with it.

F Clockwise Prompting

To verify whether MLLMs show a directional preference towards either clockwise or counter-clockwise, we evaluate Qwen-2.5-VL-7B-Instruct, GPT-4o, and o3 on ROTBENCH-SMALL using *clockwise* angles. Results from all three models closely align with the results from using counter-clockwise angles (Section 5), indicating the choice of either direction is fully arbitrary.

Model	Accuracy on Different Degrees of Rotation			
	0°	90°	180°	270°
Llama-3.2-11B-Instruct				
Zero-shot	$0.28_{\pm 0.01}$	$0.14_{\pm 0.02}$	$0.53_{\pm 0.01}$	$0.30_{\pm 0.02}$
+ Caption	$0.36_{\pm 0.03}$	$0.11_{\pm 0.02}$	$\mathbf{0.64}_{\pm 0.02}$	$0.25_{\pm 0.03}$
+ Bounding Box	$0.22_{\pm 0.01}$	$0.20_{\pm 0.02}$	$0.43_{\pm 0.01}$	$0.26_{\pm 0.02}$
+ Scene Graph	$0.27_{\pm 0.01}$	$0.17_{\pm 0.02}$	$0.43_{\pm 0.02}$	$0.26_{\pm 0.01}$
+ Chain-of-Thought	$0.45_{\pm 0.01}$	$\mathbf{0.28}_{\pm 0.02}$	$0.31_{\pm 0.01}$	$0.34_{\pm 0.00}$
+ *all above*	$\mathbf{0.47}_{\pm 0.04}$	$0.16_{\pm 0.01}$	$0.41_{\pm 0.02}$	$\mathbf{0.40}_{\pm 0.00}$

Table 4: Average classification accuracy under different image rotation angles (0°, 90°, 180°, 270°) for Llama-3.2-11B-Instruct and auxiliary information across three runs on ROTBENCH-LARGE. Accuracy is scored on a four-way classification task. We did not evaluate providing depth maps or segmentation maps for Llama-3.2-11B-Instruct (Llama) as it only supports a single image input.

Model	Accuracy (↑)			
	0°	90°	180°	270°
Gemini-2.5-Pro				
Zero-shot	**1.00**	0.50	0.72	0.40
+ Caption	**1.00**	0.48	0.76	0.44
+ Bounding Box	0.98	0.60	0.74	0.34
+ Scene Graph	**1.00**	0.54	**0.78**	0.42
+ Depth Map	**1.00**	0.44	0.66	0.52
+ Segmentation Map	0.98	0.54	0.70	0.42
+ Chain-of-Thought	**1.00**	0.23	0.44	0.40
+ Rotation Grid	0.99	0.58	0.67	0.59
+ Rotation Grid Guided	0.95	**0.71**	0.73	**0.74**
+ *all above*	**1.00**	0.60	0.76	0.68
o3				
Zero-shot	**1.00**	**0.45**	0.70	0.48
+ Caption	**1.00**	0.32	0.70	0.44
+ Bounding Box	**1.00**	0.38	0.64	0.40
+ Scene Graph	**1.00**	0.40	0.60	0.48
+ Depth Map	**1.00**	0.34	0.58	0.40
+ Segmentation Map	**1.00**	0.44	0.68	0.40
+ Chain-of-Thought	**1.00**	0.36	0.74	0.57
+ Rotation Grid	0.99	0.23	**0.83**	**0.81**
+ Rotation Grid Guided	0.99	0.31	0.82	0.75
+ *all above*	**1.00**	0.28	0.78	0.78

Table 5: Average classification accuracy under different image rotation angles (0°, 90°, 180°, 270°) for o3 and Gemini-2.5-Pro and various forms of auxiliary information across a single run on ROTBENCH-SMALL. Accuracy is scored on a four-way classification task.

G Differentiating Clockwise and Counter-clockwise Rotations using GPT-5

In Table 8, we repeat the binary clockwise versus counter-clockwise classification experiment (Section 6.2) using GPT-5. GPT-5 achieves considerably higher accuracy on this binary classification task compared to GPT-4o and Qwen-2.5-VL-7B-Instruct. Notably, performance on 90° and 270° images are significantly higher in the binary task than in the four-way classification task.

Figure 10: Example image where GPT-4o's reasoning falsely distinguishes between two identical forms of rotation.

This performance difference reveals an important limitation: GPT-5 frequently misclassifies between portrait rotations (90° and 270°) and landscape rotations (0° and 180°). Despite strong performance on 0° and 180° rotations, this confusion demonstrates that the four-way classification task cannot be effectively reduced to two separate binary classification tasks (one distinguishing between 0° and 180°, and another between 90° and 270°).

H Chain-of-Thought Example

To further understand how GPT-4o confuses clockwise and counter-clockwise rotations, we examine the generated reasoning trace of an image in detail. Figure 10 has been rotated 270° counter-clockwise (or 90° clockwise). However, GPT-4o generates the following reasoning trace:

```
To determine the rotation of the
image, let's analyze the
orientation of the buildings and
other elements: [...] 3.
**Rotation Analysis**: - A
90-degree rotation
counter-clockwise would place
```

Model	Accuracy on Different Degrees of Rotation			
	0°	90°	180°	270°
Qwen-2.5-VL-3B-Instruct				
Zero-shot	**0.60**$_{\pm 0.00}$	**0.52**$_{\pm 0.00}$	0.16$_{\pm 0.00}$	0.22$_{\pm 0.00}$
+ Chain-of-Thought	0.18$_{\pm 0.00}$	0.44$_{\pm 0.00}$	**0.42**$_{\pm 0.00}$	**0.28**$_{\pm 0.00}$
Phi-4-MM-Instruct				
Zero-shot	**0.86**$_{\pm 0.00}$	0.10$_{\pm 0.00}$	0.26$_{\pm 0.00}$	0.02$_{\pm 0.00}$
+ Chain-of-Thought	0.34$_{\pm 0.00}$	**0.20**$_{\pm 0.00}$	**0.40**$_{\pm 0.00}$	**0.28**$_{\pm 0.00}$
Gemma-3-12B-Instruct				
Zero-shot	**0.96**$_{\pm 0.02}$	0.18$_{\pm 0.03}$	**0.56**$_{\pm 0.02}$	0.18$_{\pm 0.01}$
+ Chain-of-Thought	0.34$_{\pm 0.00}$	0.20$_{\pm 0.04}$	0.40$_{\pm 0.01}$	**0.28**$_{\pm 0.05}$
Molmo-7B-O				
Zero-shot	**1.00**$_{\pm 0.00}$	0.00$_{\pm 0.00}$	0.00$_{\pm 0.00}$	0.00$_{\pm 0.00}$
+ Chain-of-Thought	0.58$_{\pm 0.00}$	**0.28**$_{\pm 0.00}$	**0.52**$_{\pm 0.00}$	**0.32**$_{\pm 0.00}$
Claude-4.1-Opus				
Zero-shot	0.96$_{\pm 0.00}$	**0.38**$_{\pm 0.00}$	**0.48**$_{\pm 0.00}$	**0.30**$_{\pm 0.00}$
+ Chain-of-Thought	**0.98**$_{\pm 0.00}$	0.28$_{\pm 0.00}$	0.42$_{\pm 0.00}$	0.12$_{\pm 0.00}$

Table 6: Average classification accuracy under different image rotation angles (0°, 90°, 180°, 270°) for Qwen-2.5-VL-3B-Instruct, Phi-4-MM-Instruct, Gemma-3-12B-Instruct, Molmo-7B-O, Claude-4.1-Opus and various forms of auxiliary information across a single run on ROTBENCH-SMALL. Accuracy is scored on a four-way classification task.

Model	Accuracy on Different Degrees of Rotation			
	0°	90°	180°	270°
Qwen-2.5-VL-7B-Instruct				
Zero-shot	**0.91**$_{\pm 0.02}$	**0.58**$_{\pm 0.03}$	0.02$_{\pm 0.02}$	**0.17**$_{\pm 0.01}$
+ Chain-of-Thought	0.82$_{\pm 0.00}$	0.51$_{\pm 0.04}$	**0.13**$_{\pm 0.01}$	0.05$_{\pm 0.05}$
GPT-4o				
Zero-shot	0.87$_{\pm 0.02}$	**0.66**$_{\pm 0.02}$	0.83$_{\pm 0.02}$	0.20$_{\pm 0.00}$
+ Chain-of-Thought	**0.92**$_{\pm 0.02}$	0.63$_{\pm 0.05}$	**0.88**$_{\pm 0.04}$	**0.21**$_{\pm 0.05}$
o3				
Zero-shot	**1.00**$_{\pm 0.00}$	**0.38**$_{\pm 0.06}$	0.71$_{\pm 0.03}$	**0.50**$_{\pm 0.03}$
+ Chain-of-Thought	**1.00**$_{\pm 0.00}$	0.27$_{\pm 0.02}$	**0.72**$_{\pm 0.02}$	0.49$_{\pm 0.03}$

Table 7: Average classification accuracy under different *clockwise* rotation angles (0°, 90°, 180°, 270°) for Qwen-2.5-VL-7B-Instruct, GPT-4o, o3 across two runs on ROTBENCH-SMALL using zero-shot or chain-of-thought prompting.

GT \ Predicted	Counter-clockwise	Clockwise
GPT-5		
Counter-clockwise	41	9
Clockwise	13	37

Table 8: Accuracy of GPT-5 in identifying 90° clockwise (CW) versus counter-clockwise rotation (CCW) on ROTBENCH-SMALL. Column indicates ground truth rotation (GT). Results are obtained using a temperature of 0.3.

```
the sky on the right and the
ground on the left, which
matches the current orientation.
- A 270-degree rotation
clockwise would place the sky on
the left and the ground on the
right, which does not match.
Therefore, the image has been
rotated 90 degrees
counter-clockwise.
```

As a 90° counter-clockwise rotation is identical to a 270° clockwise rotation, the model falsely distinguishes between the two identical forms of rotation.

I Modifying Temperature

To ensure model performance does not significantly vary with sampling temperature, we examine zero-shot performance of GPT-4o and Qwen-2.5-VL-7B-Instruct (Qwen) on varying temperature settings (Table 9). We see that performance on all four rotations does not vary significantly for GPT-4o. Qwen does show slight sensitivity to temperature.

Temperature	Accuracy on Different Degrees of Rotation			
	0°	90°	180°	270°
GPT-4o				
0.0	**0.99**$_{\pm 0.00}$	**0.69**$_{\pm 0.02}$	0.93$_{\pm 0.00}$	0.19$_{\pm 0.01}$
0.2	0.98$_{\pm 0.00}$	**0.69**$_{\pm 0.02}$	0.93$_{\pm 0.00}$	0.2$_{\pm 0.03}$
0.5	**0.99**$_{\pm 0.00}$	0.68$_{\pm 0.02}$	0.93$_{\pm 0.00}$	0.2$_{\pm 0.03}$
0.7	0.98$_{\pm 0.01}$	0.68$_{\pm 0.01}$	**0.98**$_{\pm 0.00}$	0.23$_{\pm 0.02}$
1.0	0.98$_{\pm 0.00}$	0.65$_{\pm 0.02}$	0.92$_{\pm 0.00}$	**0.25**$_{\pm 0.01}$
Qwen-2.5-VL-7B-Instruct				
0.0	**0.99**$_{\pm 0.00}$	**0.51**$_{\pm 0.01}$	0.05$_{\pm 0.01}$	0.09$_{\pm 0.01}$
0.2	**0.99**$_{\pm 0.00}$	0.48$_{\pm 0.00}$	0.06$_{\pm 0.01}$	0.11$_{\pm 0.02}$
0.5	0.94$_{\pm 0.01}$	0.44$_{\pm 0.00}$	0.09$_{\pm 0.01}$	0.19$_{\pm 0.01}$
0.7	0.9$_{\pm 0.02}$	0.42$_{\pm 0.02}$	0.09$_{\pm 0.02}$	0.2$_{\pm 0.02}$
1.0	0.83$_{\pm 0.00}$	0.39$_{\pm 0.00}$	**0.13**$_{\pm 0.02}$	**0.24**$_{\pm 0.01}$

Table 9: Average classification accuracy under different image rotation angles (0°, 90°, 180°, 270°) for GPT-4o and Qwen-2.5-VL-7B-Instruct using various sampling temperatures. Accuracy is scored on a four-way classification task across three runs on ROTBENCH-LARGE.

	0°	90°	180°	270°
0°	-	0.33	0.16	0.36
90°	0.33	-	0.21	0.20
180°	0.16	0.21	-	0.20
270°	0.36	0.20	0.20	-

Table 10: Rate of overlapping scene graphs extracted from images rotated different orientations.

Model Name	MAAE
Qwen 2.5 VL 7B Instruct	88.72 ± 2.53
GPT-4o	61.09 ± 0.48

Table 11: MAAE comparison across models.

As temperature increases, performance on 0° and 90° decreases, while performance on 180° and 270° increases. These results support our primary conclusions, suggesting that the model distinguishes between landscape rotations (0° and 180°) and portrait rotations (90° and 270°). The model defaults to 0° for landscape rotations, and 90° for portrait rotations. With temperature increases, the model's responses begin to resemble random guessing.

J The Limitations of Scene Graphs

Our primary results (Section 5) show that including various forms of auxiliary information fails to consistently improve model performance. Our intuition behind providing scene graphs is that it would capture unnatural spatial relationships between objects (i.e., "horse above human"). However, strong language priors may override the image understanding, resulting in the model providing the same caption or scene graph regardless of image rotation. To understand why scene graphs fail to improve performance, we examine the overlap between scene graphs extracted from images rotated different orientations. We define two scene graphs as overlapping when both subjects and the predicate are identical, and when both the order of subjects are reversed and the spatial predicate is inverted (e.g., ["cow", "left", "farmer"] and ["farmer", "right", "cow"] are identical scene graphs).

Table 10 displays the rate of overlapping scene graphs between different image orientations. Noticeably, there is a much higher degree of overlap between 0° and 90°/270° (around 0.3) compared to between 0° and 180° (0.16). These results show the caption model is able to recognize upside-down image and caption accordingly, but fails to correctly capture vertically rotated images.

K Alternative Evaluation Methods

Our primary experiments (Table 1, Table 2) both follow the four-option multiple-choice setup widely used in previous literature (Anis et al., 2025; Joshi and Guerzhoy, 2017; Udandarao et al., 2024) to enable a more direct comparison and analysis with previous results. In this section, we explore model performance on alternative evaluation schemes. We see that all alternative setups show inferior performance. We select our current 4-option multi-choice setup for our primary experiments as it shows that, despite the leniency of the setup, frontier models still perform unsatisfactorily.

Regression. Given that the current setup already challenges frontier reasoning models, we expect evaluating the models on a regression task on the

Number of Bins	Accuracy
Qwen-2.5-VL-7B-Instruct	
4	0.26 ± 0.03
8	0.16 ± 0.01
10	0.15 ± 0.01
18	0.10 ± 0.01
GPT-4o	
4	0.38 ± 0.02
8	0.28 ± 0.01
10	0.23 ± 0.02
18	0.17 ± 0.02

Table 12: Classification accuracy on ROTBENCH-LARGE using different discretization levels. Rows represent the number of bins, which is equivalent to the number of choices in the multiple-choice question.

continuous interval [0°, 359°] will result in further degraded performance. In fact, doing so may actually limit the conclusions and insights we can draw from our experiments as it would be difficult to infer any failure cases without first discretizing the interval. To illustrate this, we provide some further experiments evaluated Qwen-2.5-VL-7B-Instruct and GPT-4o using a modified regression setup. Each image in ROTBENCH-LARGE is randomly rotated 0° - 359° counter-clockwise. The model is asked to output a single integer representing the degree of rotation. We then calculate the Mean Absolute Angular Error (MAAE) between the estimated and true rotations. MAAE is a metric that takes into account the cyclic nature of angular measurements:

$$\text{MAAE} = \frac{1}{n} \sum \min\Big(|\hat{\theta} - \theta|,\, 360 - |\hat{\theta} - \theta|\Big)$$

We find that both models are unable to provide accurate predictions of image orientation. GPT-4o performed significantly better than Qwen-2.5-VL-7B-Instruct, with a difference in MAAE of about 20. Nonetheless, both models averaged > 45 MAAE across the dataset, indicating a limited ability of current MLLMs to reason over continuous angular spaces.

Discretization at various granularities. Alternatively, we evaluate how classification accuracy changes when discretizing the continuous interval $[0, 359]$ into various granularities. For each image in ROTBENCH-LARGE, we randomly sample a rotation angle in [0°, 359°]. We then discretize the interval into a certain number of bins, assign a letter choice to each bin, and map the ground truth rotation angle to the correct letter choice. The models are then prompted to answer a multiple-choice question, where the number of choices equals the number of bins, and each choice represents a range of angles (e.g., "A. 0°-20°"). Doing so transformed our previous experiment from a regression task into a classification task. Note that our original experimental setup is identical to the four-bin setup.

We experiment with 4 bins (each representing 90°), 8 bins (45°), 10 bins (36°), and 18 bins (20°). As expected, we see accuracy worsen as we increase the number of bins Table 12. These results suggest that both evaluated models still lack the ability to reason about fine-grained image orientations.

Num. Images	0°	90°	180°	270°
0	0.99	0.51	0.05	**0.09**
8	**1.00**	0.43	0.04	0.04
24	0.97	0.49	**0.14**	0.03
40	0.91	**0.59**	0.09	0.06

Table 13: Qwen-2.5-VL-7B-Instruct's performance on each orientation when provided different numbers of in-context images.

L In-Context Learning

Setup. To further investigate whether the solution to identifying clockwise and counter-clockwise rotations is simply one of clarifying nomenclature, we implement an in-context learning (ICL) experiment using Qwen-2.5-VL-7B-Instruct. Recall ROTBENCH-LARGE and ROTBENCH-SMALL have an overlap of 25 images. We incrementally sample 2, 6, and 10 distinct in-context examples from the remaining 25 images in ROTBENCH-SMALL and evaluate on ROTBENCH-LARGE. As each image is rotated in four orientations, the model is shown a total of 8, 24, 40 images. We randomly shuffle the order of in-context images to ensure robustness. Moreover, we provide the ground truth rotation of each in-context example to the model, aiming to guide it towards improved performance.

Results. Table 13 shows the accuracy on each image orientation after injecting in-context examples. We do not see consistent performance improvement, regardless of the number of in-context examples provided. Together with results in Table 1, these experiments suggest that various forms of prompt modification are insufficient for identifying rotation. Rather, robust orientation identification

demands explicit parameter optimization through fine-tuning.

M Normalized Rotation Voting

Model	Accuracy (↑)			
	0°	90°	180°	270°
Qwen-2.5-VL-7B-Instruct				
Zero-shot	**0.95**	**0.57**	0.03	0.15
Chain-of-Thought	0.86	0.20	0.13	0.30
Normalized Rotation Voting	0.54	0.52	**0.56**	**0.52**
GPT-4o				
Zero-shot	0.87	0.65	0.85	0.21
Chain-of-Thought	**0.91**	0.59	**0.86**	0.21
Normalized Rotation Voting	0.86	**0.88**	0.80	**0.86**

Table 14: Average classification accuracy under different image rotation angles (0°, 90°, 180°, 270°) using normalized rotation voting on ROTBENCH-SMALL. Accuracy is scored on a four-way classification task.

Algorithm 1 Normalized Rotation Voting

Require: $images = [I_0, I_{90}, I_{180}, I_{270}]$
1: $rotations \leftarrow []$
2: **for** $i \leftarrow 0$ to $|images| - 1$ **do**
3: $\quad rot \leftarrow$ call_model($images[i]$)
4: $\quad rot_norm \leftarrow (rot - i \times 90) \bmod 360$
5: $\quad$ append rot_norm to $rotations$
6: **end for**
7: $final_rot \leftarrow$ majority_vote($rotations$)

Intuition and Setup. Our results show frontier MLLMs are able to reliably identify 0° and 180° images. Leveraging this pattern, we propose a voting approach to identify image rotation. Given an input image of unknown orientation, we can *further* rotate this image 0°, 90°, 180°, and 270°. We separately prompt the model to identify the rotation of each image, then normalize the predictions by subtracting the applied rotation (effectively shifting the angle into a common reference frame) (Algorithm 1). Regardless of the original rotation, two of the further rotated images would correspond to 0° and 180° in the ground-truth reference frame. If the model can correctly identify these two images, there is a high probability that the majority vote would reveal the ground-truth rotation. We evaluate GPT-4o and Qwen-2.5-VL-7B-Instruct on ROTBENCH-SMALL using this approach.

Results. Table 14 shows the accuracy on each image orientation using normalized rotation voting. In general, we see a much more even performance distribution across all four rotations compared to using zero-shot or chain-of-thought prompting. Qwen-2.5-VL-7B-Instruct achieves substantially stronger performance on 0° and 90° (0.95 and 0.57) compared to 180° and 270° (0.03 and 0.15). Normalized rotation voting achieves 0.5 performance on all orientations. GPT-4o achieves 0.8 accuracy on all orientations. In particular, performance on 270° improved substantially compared to zero-shot prompting (+0.85).

These results show that this voting approach provides a viable means of identifying image rotation using MLLMs. However, this approach has significant drawbacks. Firstly, it increases the compute required to predict a single image, as the model processes four images for each input image. Secondly, it relies on the assumption that we have prior knowledge of all possible rotations of the image. In real-world use cases, the rotation of an input image is most likely a continuous value. While it is possible to discretize continuous rotations to a level of granularity, this approach would quickly become impractical as finer granularity increases model calls.

Figure 11: A pair of images in Spatial-MM that closely resemble each other.

N Similar Images in Spatial-MM

During the data filtering process, we noted several images in Spatial-MM closely resemble each other (Fig. 11). This realization led to our decision to use MS COCO as the training dataset in our fine-tuning experiment.

During the data filtering process, we noted several images in Spatial-MM closely resemble each other (Fig. 11). This realization led to our decision to use MS COCO as the training dataset in our fine-tuning experiment.

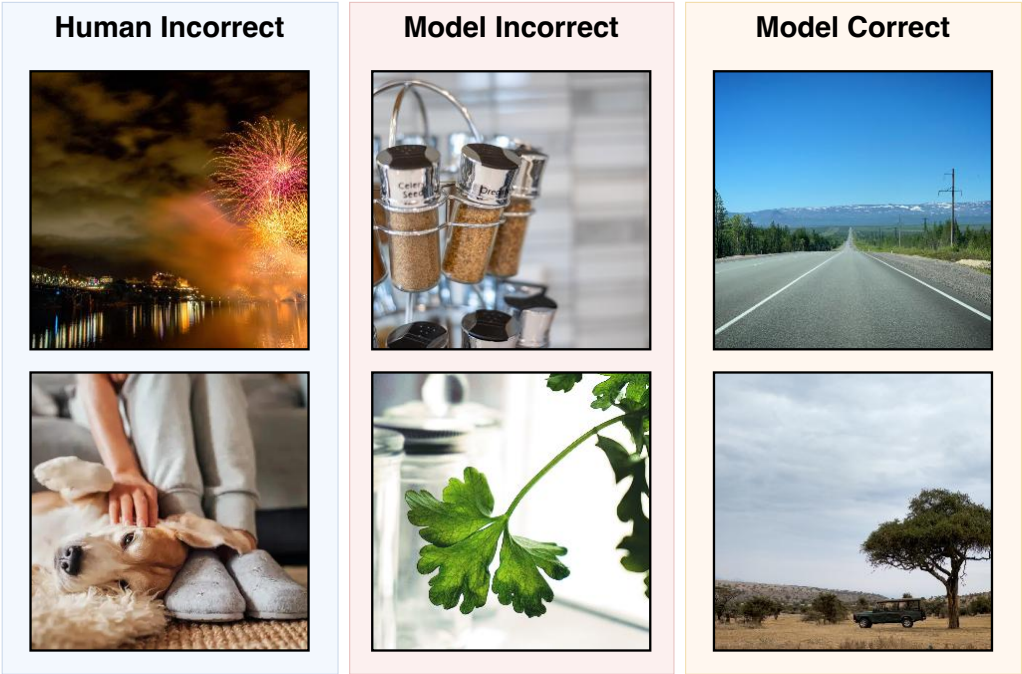

Figure 12: Qualitative examples of images that are difficult for humans (left), difficult for GPT-5 and Gemini-2.5-Pro (center), and easy for both models (right).

O Qualitative Examples of Easy and Difficult Images

This section provides a qualitative illustration of classification difficulty by presenting images that were misclassified by human annotators during Stage 2, alongside examples that both GPT-5 and Gemini-2.5-Pro classified correctly and incorrectly.

Human errors. On ROTBENCH-SMALL, humans achieve 0.99 accuracy on images rotated 0°, 90°, 180°, and 0.97 accuracy on images rotated 270°. These results indicate that for each rotation, all the annotators combined made fewer than 2 misclassifications. The images that led to misclassifications typically display subjects that can be logically oriented in multiple orientations and require incorporating background information to make an accurate judgment – such as the bottom-left image in Fig. 12.

Model errors. To better understand how the models perceive difficulty, we examined the set of images GPT-5 and Gemini-2.5-Pro answered correctly when rotated 90°, 180°, and 270° (easy images), and the images they answered incorrectly in all rotations (hard images). We include two example images from each category in the center and right columns of Fig. 12.

We find only 4 images in ROTBENCH-SMALL (50 images total) elicit incorrect classifications for both GPT-5 and Gemini-2.5-Pro across the three rotations. The images tend to have these characteristics:

1. The image features a prominent primary subject and blurred/subtle background. Moreover, the primary subject can be viable even when rotated in multiple orientations (bottom-center image in Fig. 12). Accurate identification of rotation requires incorporating the background into the reasoning.

2. The rotational cue is slight or subtle. The 'Flagged Image' example in Fig. 9 belongs to this category, as the sole rotational cue is the slight tilt of the camera. If the camera was directly looking down, the image would be viable in any rotation.

On the other hand, there are 5 images that both models got correct in all orientations. These images often feature a clear central character, or tend to be natural landscape images featuring clear color distinctions between sky and ground (right column in Fig. 12).

P License

We will publicly release our models, and include our code and data in the supplementary. We provide the following links to the standard licenses for the datasets, code, and models used in this project.

- **Spatial-MM:** No license specified (accessed October 6, 2025). Annotations from GitHub.
- **MS COCO:** Creative Commons Attribution 4.0 License.
- **Qwen-2.5-VL-7B-Instruct:** Apache 2.0.
- **Qwen-2.5-VL-3B-Instruct:** Qwen Research License.
- **Llama-3.2-11B-Instruct:** Llama 3.2 Community License.
- **GPT-4o, GPT-4.1, GPT-5, o3:** OpenAI Services Agreement and Service Terms.
- **Gemini-2.0-Flash, Gemini-2.5-Flash, Gemini-2.5-Pro:** Gemini API Additional Terms of Service.
- **Phi-4-MM-Instruct:** MIT.
- **Gemma-3-12B-Instruct:** Gemma Terms of Use.
- **Claude-4.1-Opus:** Anthropic Consumer Terms of Service.
- **Molmo-7B-O:** Apache 2.0.

Q Prompts

This section includes all prompts used in our experiments.

Q.1 Extract Image Subjects

Fig. 13 describes the prompt used for extracting primary subjects in images. The list of subjects extracted is later used to obtain bounding boxes, scene graphs, and segmentation maps .

Q.2 Captions

Fig. 14 describes the prompt used for captioning images.

Q.3 Scene Graphs

Fig. 15 describes the prompt used for extracting scene graphs from images.

Q.4 Clockwise Experiment

Fig. 16 describes the prompt used for clockwise vs counter-clockwise rotation experiment.

Q.5 Rotation Classification

Fig. 17 describes the system and user prompts for rotation classification. The mapping between letter choice and degrees is shuffled each prompt.

User Prompt

<Image encoded via. Base64>
Return a list of objects in this image. The list will later be passed to a bounding box model to extract bounding boxes for each detected object. Format your response as a Python list, surrounded with a Python markdown fence. For example: “‘python[’fedora’, ’woman in green dress’, ’man in red suit’, ...]“‘ Each object should have a distinct name. ENSURE YOUR RESPONSE FOLLOWS THE FORMATTING REQUIREMENTS!

Figure 13: Prompts used for extracting primary subjects in images.

User Prompt

<Image encoded via. Base64>
Generate a detailed caption for this image. Do not include any preceding text before the caption.

Figure 14: Prompts used for captioning images.

User Prompt

<Image encoded via. Base64>
Task: Given the image and key objects, generate a scene graph for this image. Represent each relationship as a three-element tuple with (’subject_id’, ’predicate’, ’object_id’). Extract a set of words describing the location, orientation, directions and spatial or positional relations between key objects in the image. Your answer should be a list of values that are in the format of (object1, relation, object2). The relation MUST be one of [left, right, above, below, facing left, facing right, front, behind]. You are to interpret the image literally. If you see a sky below a mountain, your scene graph must reflect that. Format your response as a Python list of tuples, surrounded by a markdown fence. Example formatting:
“‘python [("object1", "predicate1", "object2"), ("object2", "predicate2", "object3"), ...] “‘
Key objects in the image: <previously extracted image subjects>

Figure 15: Prompts used for extracting scene graphs from images.

System Prompt

You are an intelligent AI assistant that specializes in identifying rotation in images. You will be given an image that has been rotated 90 or 270 degrees. Specifically, a 90 degree rotation is a quarter-turn counter-clockwise (the same as 270 degrees clockwise); 270 is three quarter-turns counter-clockwise (the same as 90 degrees clockwise).

User Prompt

<Image encoded via. Base64>
Your task is to identify whether the image has been rotated 90 or 270 degrees counter-clockwise. Examine the image closely and identify the rotation. Let’s think step-by-step.

Figure 16: Prompts used for clockwise versus counter-clockwise rotation experiment.

System Prompt

You are an intelligent AI assistant that specializes in identifying rotation in images. You will be given an image and a multiple choice question. Each choice corresponds to the number of degrees the image has been rotated. A 90° rotation is a quarter-turn counter-clockwise; 270° is a quarter-turn clockwise. A 0° rotation indicates the image is right-side up; a 180° rotation indicates the image is upside-down.

User Prompt

<Image encoded via. Base64>
< (if included) Depth map encoded via. Base64>
< (if included) Segmentation map encoded via. Base 64>
Identify whether the image has been rotated. In addition, you have been provided some extra information about this image below.
< If caption >
The image is given the following caption: <caption>
< If bounding box>
Below is the normalized bounding box of objects in the image. Each object is bounded by four floats [xmin, ymin, xmax, ymax] (each float has been normalized between 0 and 1) <bounding boxes>
< If scene graph >
Below is a scene graph representing objects within the image and the relationship between them. <scene graph>
< If depth map >
Attached is also an estimated depth map of the image. The brighter the pixel, the further it is.
< If segmentation map > Attached is also a segmentation map of the image. Each object has been highlighted a different color.
< If CoT >
What is the rotation of this image? Let's think step-by-step.
< If rotation grid >
Attached is a grid showing the image rotated counter-clockwise in different orientations. The top-left image is the original image shown prior, the top-right image has been rotated 270° counter-clockwise, the bottom-right 180° counter-clockwise, and the bottom-left 270° counter-clockwise. Using these other images as an aid, what is the rotation of the original image? Let's think step-by-step
< If rotation grid guided >
Attached is a grid showing the image rotated counter-clockwise in different orientations. The top-left image is the original image shown prior, the top-right image has been rotated 270° counter-clockwise, the bottom-right 180° counter-clockwise, and the bottom-left 270° counter-clockwise. Use this three step procedure: (1) Carefully examine all four images shown in the grid. (2) Identify the image you are most familiar with, or which image most resembles your training data, as an anchor point. (3) Starting from that image, algebraically determine the rotation of the original image. Using these other images as an aid, what is the rotation of the original image? Let's think step-by-step
< Else >
Response with a SINGLE LETTER, either A, B, C, or D, representing the correct rotation. You must select one of these choices even if you are uncertain. DO NOT INCLUDE ANYTHING ELSE IN YOUR RESPONSE.
The rotation of the image is: A. 0 B. 270 C. 90 D. 180
Answer:

Figure 17: System and user prompts for rotation classification.