Tiny Scales, Great Challenges: The Limits of Multimodal LLMs in Scale Recognition

Jihang Jin[1], **Ronghao Chen**[2], **Hao Zhang**[3], **Ziyan Liu**[1], **Huacan Wang**[3], **Qi Ye**[1*], **Jingping Liu**[4*]
[1]East China University of Science and Technology, Shanghai, China,
[2]Peking University, Beijing, China,
[3]University of Chinese Academy of Sciences, Beijing, China,
[4]Sun Yat-sen University, Guangzhou, China
Correspondence: y30251116@mail.ecust.edu.cn, liujp68@mail.sysu.edu.cn

Abstract

Visual scale recognition is a fundamental aspect for humans to perceive physical quantities in the real world, and it is crucial for enabling human-like intelligence in multimodal large language models (MLLMs). However, existing benchmarks typically focus on a single type of quantity (e.g., time) or a specific format (e.g., dials), lacking a comprehensive evaluation of scale recognition capabilities. To address these problems, we propose ScaleBench, a visual scale recognition benchmark built using images from COCO, Open Images, and Flickr, designed to comprehensively evaluate the scale recognition capabilities of MLLMs. To ensure high data quality, we develop detailed annotation guidelines and procedures, resulting in a total of 6,574 annotated samples. Based on this benchmark, we evaluate multiple closed-source and open-source MLLMs. Experimental results reveal that the best-performing model achieves only 42.60% accuracy, far lower than the 97.40% of humans. Furthermore, we conduct in-depth experimental analyses and provide future research directions. Our benchmark and implementation codes are available at https://github.com/Sonder-hang/ScaleBench.

1 Introduction

Multimodal large language models have demonstrated significant value in processing and integrating information from various modalities, such as text and images. However, they continue to face challenges in handling complex and nuanced tasks (Liu et al., 2026), including multi-modal semantic understanding (Zhang et al., 2024) and spatial reasoning (Liu et al., 2025), highlighting the need for continued exploration of their capabilities.

An important aspect of evaluating MLLMs is their ability to recognize visual measurement scales. This task requires the model to: identify the measurement objects depicted in the image, interpret the scale's units and intervals, and recognize the specific value indicated by a pointer or marker. For instance, in Figure 1(a), the model needs to identify the thermometer displaying 37.3 degrees Celsius. This ability is important for machines because scale recognition is a fundamental way in which humans perceive and interpret physical quantities in the real world. Continuing with the previous example, if machines are to perform human-like tasks, they must be able to accurately read thermometer values, which are used to determine whether a fever is present.

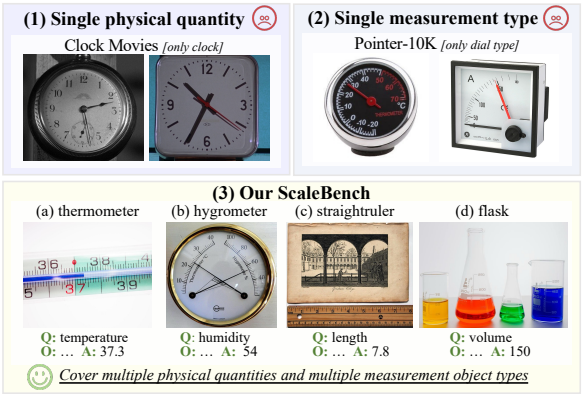

Figure 1: Samples from scale recognition benchmarks. "Q", "O", and "A" in our ScaleBench mean the question, options, and answer.

Despite the emergence of various benchmarks for evaluating visual scale recognition, current resources remain inadequate for comprehensively assessing MLLMs in this area. Existing benchmarks are typically categorized by the number of physical quantities they encompass. The first category comprises datasets focused on a single physical quantity, such as SynClock (Yang et al., 2022), Clock Movies (Yang et al., 2022), and Hou et al. (Hou et al., 2023). While useful for evaluating specific perceptual capabilities (e.g., SynClock and Clock Movies target time perception), their narrow scope limits the generalizability of find-

*Corresponding authors.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 40619–40632

Table 1: Comparison of scale recognition benchmark. "Q.", "C.", "S.", "U.", "P.", and "bbox" denote "Quantities", "measurement object categories", "single object style (Only our ScaleBench includes all three measurement object styles—dial, linear, and cylindrical—while other benchmarks include only the dial style)", "single scale units in an image", "single pointer in an image", and "bounding box", respectively. "SynData" (S.D.), "SelfCap" (S.C.), "W.S.", "SimEnv", "CO.", "O.I.", and "FLK" stand for "synthetic data", "self-captured", "web-sourced", "simulated environment", "COCO", "Open Images", and "Flickr", respectively.

Benchmark	Source	# Q.	# C.	w/o S.	w/o U.	w/o P.	w/o bbox	Size
SynClock (Yang et al., 2022)	SynData	1	1	×	×	✓	✓	-
Clock Movies (Yang et al., 2022)	Movie	1	1	×	×	✓	✓	1,244
Hou et al. (Hou et al., 2023)	SelfCap	1	1	×	×	×	×	2,300
Wu et al. (Wu et al., 2021)	SimEnv, S.C.	3	3	×	×	×	×	2,952
Meter Chall. (Shu et al., 2023)	SelfCap	4	4	×	×	×	×	1,296
Pointer-10K (Dong et al., 2021)	S.C., S.D., W.S.	5	5	×	×	✓	✓	10,000
Real-Gauges (Howells et al., 2021)	SelfCap	2	2	×	×	×	✓	2,700
Syn.-Gauges (Howells et al., 2021)	SynData	5	5	×	×	×	×	11,000
ScaleBench (ours)	CO., O.I., FLK	13	33	✓	✓	✓	✓	6,574

ings across broader physical properties. The second category includes benchmarks incorporating multiple quantities, such as Wu et al. (Wu et al., 2021), Meter Challenge (Shu et al., 2023), Pointer-10K (Dong et al., 2021), and Real-Gauges and Synthetic-Gauges (Howells et al., 2021). Although these offer greater diversity in quantities, they predominantly feature dial-based instruments (e.g., Pointer-10K's pressure gauges, ammeters, and voltmeters), thus constraining measurement modality variation and failing to reflect real-world instrument diversity. Consequently, existing benchmarks exhibit limitations in both the range of physical quantities and the types of measurement instruments, as depicted in Figure 1. This underscores a critical need for more comprehensive evaluation resources for visual scale recognition in MLLMs.

To this end, in this paper, we propose ScaleBench, a new multiple-choice benchmark designed to evaluate the scale recognition capabilities of MLLMs. ScaleBench is constructed from images collected from the COCO dataset (Lin et al., 2014), the Open Images dataset (Kuznetsova et al., 2020), and the public image platform Flickr,[1] comprising 5,371 images and 6,574 annotated samples. Unlike previous benchmarks, ScaleBench covers 13 commonly used physical quantities, spanning 33 types of measurement objects and 38 scale units, thereby aligning more closely with real-world application scenarios. A detailed comparison with existing benchmarks is provided in Table 1. To ensure the quality of the benchmark, we develop a clear annotation guideline and conduct three rounds of rigorous quality control involving professional annotators and reviewers. Based on this benchmark, we conduct a comprehensive evaluation of several representative MLLMs, including closed-source models (GPT-5[2] and Gemini 2.5 Pro (Team et al., 2023)) and open-source models (e.g., LLaVA (Liu et al., 2023a), Qwen-VL (Bai et al., 2025), and mPLUG-Owl (Ye et al., 2024)).

Contributions. Our contributions are summarized as:

- We introduce ScaleBench, a manually annotated, high-quality benchmark specifically designed to evaluate the visual scale recognition capabilities of MLLMs.
- ScaleBench covers 13 physical quantities, 33 measurement objects, and 38 scale units, effectively capturing a wide range of real-world scale recognition scenarios.
- We test several open-source and closed-source MLLMs on ScaleBench. Experiments show that the current state-of-the-art (SoTA) Gemini-2.5-pro and the fine-tuned mPLUG-owl3-7B only achieve 42.60% and 41.22% accuracy, which is far lower than the 97.40% of humans. In addition, we also provide future research directions for this benchmark.

2 Related Work

Visual scale recognition. Visual scale recognition is an important task in AI (Reitsma et al., 2024; Feng et al., 2025). Existing benchmarks

[1] https://www.flickr.com/

[2] https://openai.com/gpt-5-system-card/

in this area can be grouped into two categories: those focusing on a single physical quantity and those covering multiple quantities. In the first category, typical datasets include SynClock (Yang et al., 2022), Clock Movies (Yang et al., 2022), and Hou et al. (Hou et al., 2023). Both SynClock and Clock Movies target time measurement using dial clocks, with the former being synthetic and the latter collected from movie scenes. Hou et al. (Hou et al., 2023) focuses on pressure and uses images of dial pressure gauges. While useful, they lack diversity and fail to evaluate models across broader scenarios (Li et al., 2025, 2024). In the second category, examples include Real-Gauges and Synthetic-Gauges (Howells et al., 2021), Meter Challenge (Shu et al., 2023), Wu et al. (Wu et al., 2021), and Pointer-10K (Dong et al., 2021). Real-Gauges focuses on pressure and temperature, while Synthetic-Gauges adds current, speed, and volume. Meter Challenge and Wu et al. cover temperature, pressure, and current, and Pointer-10K extends to pressure, temperature, current, voltage, and volume. Although these datasets span multiple quantities, they mainly rely on dial-type instruments with limited scale diversity. Our ScaleBench addresses this limitation by including a broader range of scale types—circular (e.g., clocks, gauges), linear (e.g., rulers, thermometers), and cylindrical (e.g., measuring cylinders)—offering diversity and realism for evaluating MLLMs in real-world scenarios.

Multimodal large language model. In recent years, MLLMs have made rapid progress and become a key focus in visual-text understanding tasks (Wang et al., 2024; Fu et al., 2025). These models are usually either closed-source or open-source. Closed-source models like GPT-5 and Gemini lead in performance, mainly because they are large and trained on high-quality data. But since they are not open to the public, researchers can not directly fine-tune them for specific tasks. As a result, researchers often rely on ICL (Shukor et al., 2023; Liu et al., 2023b; Doveh et al., 2024) and prompt engineering (Chen et al., 2025a; Son and Lee, 2025) to adapt these models. By carefully crafting instruction templates and selecting relevant examples, these models can be guided to perform tasks such as image interpretation and cross-modal reasoning. In contrast, open-source models, such as LLaVA (Liu et al., 2023a), Qwen-VL (Bai et al., 2025), and mPLUG-Owl (Ye et al., 2024), are more flexible, but generally underperform due to smaller parameter sizes and limited training data. They also struggle more with following instructions for specific tasks (Ding et al., 2025). To address this, researchers use instruction fine-tuning—a way to teach models how to better follow task instructions. Techniques like LoRA (Hu et al., 2021) and prefix tuning (Li and Liang, 2021) help boost their performance without needing to retrain the whole model. Although MLLMs continue to improve, they still show performance bottlenecks on our ScaleBench benchmark, highlighting their current limitations in visual scale recognition tasks.

3 Problem Formulation

In this paper, we formulate scale recognition as a multiple-choice question-answering task.[3] Given a text question Q and an image I, where Q asks about the scale reading of an entity shown in I, the task is to select the correct answer from four given options. For example, as shown in Figure 1(a), given the question "What is the temperature shown by the thermometer in the image, in degrees Celsius?", the model should analyze the image and choose "37.3" from the candidate options as the correct answer.

4 ScaleBench Construction

In this section, we detail the construction process of ScaleBench, including image sources, annotation guidelines, and the annotation procedure.

4.1 Image Sources

In this paper, we collect images from three main sources: the COCO2017 dataset (Lin et al., 2014), the Open Images V7 dataset (Kuznetsova et al., 2020), and the public image-sharing website Flickr. First, we obtain images from the COCO dataset, which contains over 160,000 images depicting various scenes, including many objects with scale information (such as clocks and rulers). To extract relevant content efficiently, we remove COCO images by category tags and captions. Second, we gather images from the Open Images dataset, which consists of 9 million high-resolution images spanning a broad range of object categories, with many images containing clear scale structures (such as weighing scales and thermometers). We focus on categories with measurement-related objects and prioritize images with visible numerical or pointer-based indicators. To further diversify our dataset, we search

[3] Appendix A also reports results on a benchmark that includes both multiple-choice and fill-in-the-blank questions.

Flickr using scale-related keywords and collect additional images, often representing real-world scenes involving laboratory equipment, barometers, and pressure gauges.

After aggregating images from these three sources, we conduct a manual review to remove blurry, low-resolution, or unclear images related to scale. Ultimately, we curate a set of 9,000 candidate images for subsequent scale reading annotation. This curated set serves as the foundation for constructing the final benchmark, ensuring both diversity and clarity in visual scale recognition.

4.2 Annotation Guidelines

Next, we annotate each image with a question, candidate options, and the correct answer. To guide annotators in generating high-quality samples, we provide clear annotation guidelines, as well as both correct and incorrect examples for reference.

Each question is specifically tailored to align with the visual information in the image, clearly indicating which scale object or unit is to be identified. When images contain multiple entities—such as several gauges—or display different unit types (e.g., Celsius and Fahrenheit), the questions are carefully worded to specify the target entity or scale unit, eliminating ambiguity. For each question, there is only one correct answer, determined directly from the scale reading visible in the image, without any reliance on guesswork or estimation. Incorrect options are constructed to be plausible yet clearly incorrect, reflecting common errors that might arise when interpreting the image. For example, when an image has multiple pointers (Figure 1(b)), different units (Figure 1(c)), or several measuring objects (Figure 1(d)), other visible readings in the image are used as distractor options.

4.3 Annotation Procedure

To build a high-quality benchmark, we organize a team of three annotators, two checkers, and one reviewer. All team members receive professional training on scale recognition tasks and annotation guidelines. The procedure includes annotation, verification, and final review.

First, three college students served as annotators, with each assigned 3,000 images. For every image, they generate a question, four options, and the correct answer according to the guidelines. To improve efficiency and reduce workload, we design question templates tailored to different types of scale units. For example, a template for volume recognition is "What is the volume of *object* in the image, in *unit_type*?", where *object* and *unit_type* correspond to the specific object and its measurement unit. A complete list of templates is included in Appendix B.

Table 2: Statistics of our ScaleBench. "others" is composed of 4 physical quantities, including electric current, voltage, acidity, and sound.

	Train	**Dev**	**Test**	**Total**	**# Img**
ScaleBench	4,590	645	1,339	6,574	5,371
Physical quantity types					
angle	214	30	62	306	210
humidity	118	16	35	169	165
length	907	128	264	1,299	1,134
pressure	673	95	196	964	800
speed	268	37	80	385	312
temperature	1,314	187	378	1,879	1,358
time	450	64	130	644	638
volume	440	60	133	633	494
weight	83	11	25	119	99
others	123	17	36	176	161

Second, the samples are passed to two additional college students, who act as checkers. They verify the clarity and correctness of each question, option, and answer, as well as whether the *object* in the image is clearly identifiable. If either checker identifies a sample unsatisfactory, it is returned to the annotator for revision, along with a detailed explanation. This verification process is repeated until the batch reaches an accuracy of at least 95%.

Finally, all samples are submitted to the lead author for final review. The lead author randomly checks 20% of the batch. If any issues are found, the batch is sent back for further revision. This process repeats until the final review accuracy reaches 98%. Ultimately, we obtain 6,574 high-quality samples, forming the ScaleBench benchmark.

5 ScaleBench Analysis

ScaleBench statistics. As shown in Table 2, ScaleBench comprises 6,574 samples and 5,371 images, with 1,112 images associated with multiple questions. The benchmark is split into training, validation, and test sets in a 7:1:2 ratio. Moreover, we analyze the distribution of samples across the nine main types of physical quantities. Temperature and length recognition account for the majority, with 1,879 and 1,299 samples, respectively, whereas the humidity and weight recognition categories have comparatively few samples, each ranging between

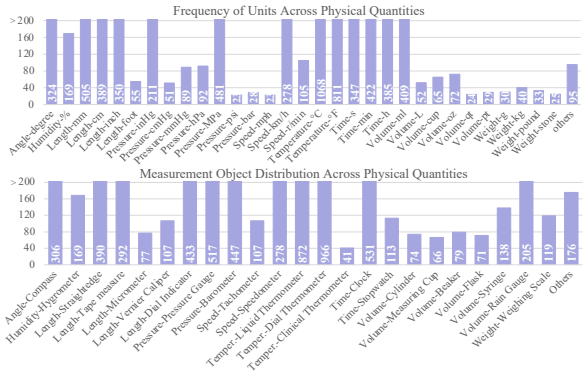

Figure 2: Distribution of units and measurement objects.

100 and 200. The "others" category, which includes electric current, voltage, acidity, and sound, contains 176 samples and helps increase the diversity of the dataset. This diverse coverage of physical quantities ensures that ScaleBench supports a broad evaluation of model capabilities across different measurement scenarios.

ScaleBench diversity. Our benchmark covers common units and measurement objects across various physical quantities, aiming to ensure a diverse set of questions. We report the frequency of different units and the number of measurement objects for each physical quantity, as illustrated in Figure 2. In "Frequency of Units Across Physical Quantities", a total of 38 units are used. Among them, 31 are listed individually, while 7 less frequent ones are grouped under the "Others" category. In "Measurement Object Distribution Across Physical Quantities", there are 33 measurement objects in total, with 23 shown explicitly and 10 classified as "Others". Thermometers appear most frequently in the benchmark, which leads to a high frequency of temperature-related units—up to 1,068 occurrences. Overall, our benchmark includes both commonly used units across physical quantities and a wide variety of measurement objects found in everyday life.

6 Experiments

In this section, we evaluate several SoTA MLLMs on ScaleBench and provide a detailed performance analysis. We also conduct an error analysis to better understand their limitations on the benchmark.

6.1 Baselines

We consider the following three baseline methods:

Open-source MLLMs. We select DeepSeek-VL-7B (Lu et al., 2024), MiniCPM-V-8B (Yao et al., 2024), LLaMA-3.2-11B-Vision (Grattafiori et al., 2024), LLaVA-v1.6-7B (Liu et al., 2023a), mPLUG-owl3-7B (Ye et al., 2024), Phi-3.5-vision-4B (Abdin et al., 2024), Janus-Pro-7B (Chen et al., 2025b), Qwen2.5-VL-7B (Bai et al., 2025), and InternVL3-8B (Chen et al., 2024). We evaluate them under two settings: (1) direct inference, where the model directly predicts an answer given a question, an image, and four options; and (2) parameter-efficient fine-tuning, where we use LoRA (Hu et al., 2021) to train the models using instruction data. This data is generated by transforming the input-output pairs from the training set.

Closed-source MLLMs. We evaluate Gemini-2.5-pro[4] and GPT-5[5] under two settings: zero- and few-shot reasoning. In the zero-shot setting, we provide the model with a question, an image, and options, along with a task prompt to guide the response. In the few-shot setting, we apply 1-shot, 2-shot, and 3-shot in-context learning (ICL), using the same task prompt. The examples are randomly selected from the training set and remain fixed across all test samples.

Other methods. We include two baselines: random guess, which uses a random function to select answers; and human evaluation, based on the average accuracy of three college students (not involved in annotation) on the randomly selected 500 samples: up to 50 per category (all included if fewer than 50), with the remaining randomly drawn from the rest.

6.2 Metrics and Implementations

We report four evaluation metrics: precision (P), recall (R), F1 score (F1), and accuracy (Acc). We train the model on four A800 80G GPUs using LoRA with a rank of R=32. The training is conducted over 6 epochs with a batch size of 4, using AdamW and a cosine learning rate scheduler. The learning rate is set to 2×10^{-5}. The task prompts of MLLMs are provided in Appendix C.

6.3 Main Results

We perform all baseline methods on the constructed ScaleBench, and the results are reported in Table 3. Overall, current MLLMs perform poorly on this new benchmark, with accuracies far below the human level of 97.40%. For instance, the best open-source model, mPLUG-owl3-7B fine-tuned

[4] gemini-2.5-pro-preview-05-06
[5] gpt-5-2025-08-07

Table 3: Model performance (%) on ScaleBench, averaged over three runs. We rule out the possibility that model errors stem from missing domain knowledge (e.g., not recognizing units or how to read them); see Appendix D.

Model	Setting	P	R	F1	Acc	Setting	P	R	F1	Acc
Open-source MLLMs										
DeepSeek-VL-7B	-	23.57	23.66	23.62	23.75	LoRA	37.39	33.63	35.41	33.01
MiniCPM-V-8B	-	28.10	27.45	27.77	27.78	LoRA	40.74	39.97	40.35	39.96
Llama-3.2-11B-Vision	-	23.39	23.38	23.39	23.67	LoRA	38.38	37.68	38.03	37.94
LLaVA-v1.6-7B	-	20.78	25.13	22.75	24.35	LoRA	40.62	40.56	40.59	40.55
mPLUG-owl3-7B	-	32.62	31.91	32.26	31.81	LoRA	41.14	41.12	41.13	**41.22**
Phi-3.5-vision-4B	-	23.82	23.81	23.82	23.82	LoRA	36.68	36.41	36.54	36.45
Janus-Pro-7B	-	46.66	28.09	35.07	28.60	LoRA	40.80	38.61	39.67	38.46
Qwen2.5-VL-7B	-	27.37	27.05	27.21	27.48	LoRA	39.36	38.39	38.87	38.39
InternVL3-8B	-	29.74	29.17	29.45	29.50	LoRA	39.94	39.74	39.84	39.66
Closed-source MLLMs										
Gemini-2.5-pro	0-shot	43.43	42.91	43.17	**42.60**	2-shot	39.42	39.24	39.33	38.80
	1-shot	41.00	40.80	40.90	40.40	3-shot	36.67	36.74	36.70	36.40
GPT-5	0-shot	36.65	36.05	36.35	36.00	2-shot	35.36	35.34	35.35	35.80
	1-shot	33.72	33.65	33.69	34.00	3-shot	38.54	38.68	38.61	39.00
Other Methods										
Random Choose	-	28.03	27.86	27.95	28.00	-	-	-	-	-
Human	-	97.44	97.42	97.43	**97.40**	-	-	-	-	-

with LoRA, achieves only 41.22% accuracy, while the best closed-source model, Gemini-2.5-Pro in the zero-shot setting, reaches just 42.60%. These results indicate that scale recognition remains a significant challenge for existing MLLMs. In addition, fine-tuned open-source models consistently outperform their untuned counterparts, generally improving accuracy by around 10%. For closed-source models, Gemini-2.5-pro performs best in the 0-shot setting (42.60%) but degrades as more ICL examples are added, suggesting limited ability to leverage multi-image examples. In contrast, GPT-5 shows an initial drop from 0-shot (36.00%) to 1-shot (34.00%) but improves steadily with more examples, indicating that while a single example may mislead the model, multiple examples provide more stable and robust guidance, enabling the model to adapt more effectively.

6.4 Detailed Analysis

Analysis of model performance across different physical quantities. To evaluate the performance of MLLMs on different physical quantities, we measure their accuracy across various categories. Table 4 shows the results for both fine-tuned open-source models and closed-source models in the 0-shot and best few-shot settings. Overall, all models perform substantially worse than humans on physical quantities, with accuracies generally below 60%, whereas human performance exceeds 94% across all quantities. In some cases, performance is barely above blind guessing—for example, open-source models on Volume. These results further demonstrate that current MLLMs remain far from achieving human-level performance in scale recognition. Among open-source models, LLaVA-v1.6-7B demonstrates balanced performance across categories. LLaMA-3.2-11B-Vision achieves the highest accuracy in "Humidity", outperforming all other models in that category. In contrast, DeepSeek-VL-7B performs poorly across most categories, sometimes falling below the level of random selection. For closed-source models, GPT-5 shows more uneven performance in scale recognition compared with Gemini-2.5-pro. For instance, its accuracy on "Angle" and "Humidity" is close to random guessing, while its performance on "Time" and "Volume" is only slightly better.

Impact of ICL example selection on model performance. To analyze the impact of different ICL examples on the performance of closed-source models, we classify the ICL examples into two categories: those aligned with the physical quantity category of the input question and those misaligned. The results are shown in Table 5. Models generally perform better when provided with aligned examples. For example, when Gemini-2.5-pro is given three aligned examples, its accuracy is 6.40% higher than when using three misaligned examples.

Table 4: Model performance (Acc %) across different physical quantity types on ScaleBench.

Model	Setting	Angle	Humid.	Length	Pressu.	Speed	Temper.	Time	Volume	Weight	Other
Open-source MLLMs											
DeepSeek-VL-7B	LoRA	20.97	31.43	40.15	38.27	43.75	28.04	28.46	33.08	24.00	25.00
MiniCPM-V-8B	LoRA	38.71	37.14	42.05	45.92	57.50	38.36	34.62	30.83	40.00	27.78
Llama-3.2-11B-V	LoRA	25.81	54.29	41.67	41.33	56.25	33.33	35.38	36.84	28.00	25.00
LLaVA-v1.6-7B	LoRA	35.48	45.71	44.70	38.78	55.00	37.30	38.46	34.59	52.00	47.22
mPLUG-owl3-7B	LoRA	25.81	37.14	46.97	39.29	55.00	42.06	35.38	35.34	44.00	41.67
Phi-3.5-vision-4B	LoRA	33.87	42.86	41.67	35.20	52.50	34.13	33.85	36.09	28.00	8.33
Janus-Pro-7B	LoRA	32.26	40.00	47.35	42.86	51.25	32.01	32.31	32.33	48.00	36.11
Qwen2.5-VL-7B	LoRA	27.42	42.86	47.73	33.67	56.25	33.86	30.77	37.59	60.00	33.33
InternVL3-8B	LoRA	35.48	34.29	45.45	42.35	57.50	37.30	33.08	33.08	32.00	33.33
Closed-source MLLMs											
Gemini-2.5-pro	0-shot	37.10	42.86	50.76	39.29	53.75	37.30	45.38	36.84	48.00	47.22
	1-shot	44.19	40.00	47.76	31.75	41.86	39.76	35.71	38.78	52.00	38.89
GPT-5	0-shot	30.23	20.00	29.85	25.40	46.51	36.14	46.43	53.06	44.00	30.56
	3-shot	23.26	37.14	34.33	41.27	41.86	36.14	50.00	46.94	48.00	33.33
Other Methods											
Random Choose	-	30.23	28.57	26.87	25.40	34.88	27.71	19.64	32.65	36.00	25.00
Human	-	97.67	97.14	97.01	96.83	97.67	98.80	98.21	95.92	96.00	94.44

Table 5: Model accuracy (%) with different ICL examples.

Model	Setting	Align.	Misalign.
Gemini-2.5-Pro	1-shot	40.20	39.80
	2-shot	40.80	37.40
	3-shot	41.40	35.00
GPT-5	1-shot	34.20	32.60
	2-shot	37.40	35.60
	3-shot	39.80	37.60

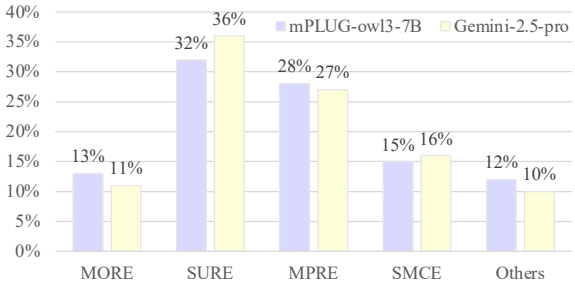

Figure 3: Error types of SoTA MLLMs on ScaleBench.

Furthermore, when using aligned examples, the performance of both models improves as the number of examples increases. However, the trend differs when using misaligned examples. For Gemini-2.5-pro, performance decreases as more misaligned examples are added. In contrast, GPT-5's performance still improves with more misaligned examples, though not as significantly as with aligned ones. This contrast suggests that GPT-5 may be more robust to suboptimal context or better at extracting relevant patterns even from less aligned examples compared to Gemini-2.5-pro.

Impact of model parameter size on performance. To evaluate how model performance varies with parameter size on ScaleBench, we test Qwen2.5-VL and InternVL3 families under two settings: without fine-tuning and with LoRA fine-tuning. The results are reported in Table 6. In general, larger models tend to perform better, but this is not always the case. For example, InternVL3-78B performs worse than InternVL3-38B in accuracy without fine-tuning. Furthermore, the best overall accuracy is only 42.57%. This suggests that simply increasing the number of parameters is not enough to solve the scale recognition task. When LoRA fine-tuning is applied, performance improves significantly for most models, especially smaller ones. For instance, Qwen2.5-VL-7B performs poorly without fine-tuning but achieves results close to the 32B and 72B models after LoRA is used. This highlights the importance of task-specific fine-tuning, particularly for smaller models.

6.5 Error Analysis

To support future research on scale recognition in MLLMs, we analyze 200 error cases from the test set, selected from predictions made by the best-performing open-source and closed-source

Table 6: Model performance (%) across different parameter sizes on ScaleBench.

Model	Setting	P	R	F1	Acc	Setting	P	R	F1	Acc
Qwen-VL Model Family										
Qwen2.5-VL-7B	-	27.37	27.05	27.21	27.48	LoRA	39.36	38.39	38.87	38.39
Qwen2.5-VL-32B	-	35.17	32.51	33.79	32.56	LoRA	45.47	39.24	42.13	38.69
Qwen2.5-VL-72B	-	33.86	32.37	33.10	32.79	LoRA	38.90	38.86	38.88	38.98
InternVL Model Family										
InternVL3-8B	-	29.74	29.17	29.45	29.50	LoRA	39.94	39.74	39.84	39.66
InternVL3-14B	-	32.51	32.01	32.26	32.41	LoRA	42.63	42.50	42.56	42.57
InternVL3-38B	-	35.14	35.27	35.21	35.32	LoRA	41.78	41.81	41.79	41.67
InternVL3-78B	-	32.54	32.63	32.58	32.94	LoRA	42.07	41.91	41.99	42.05

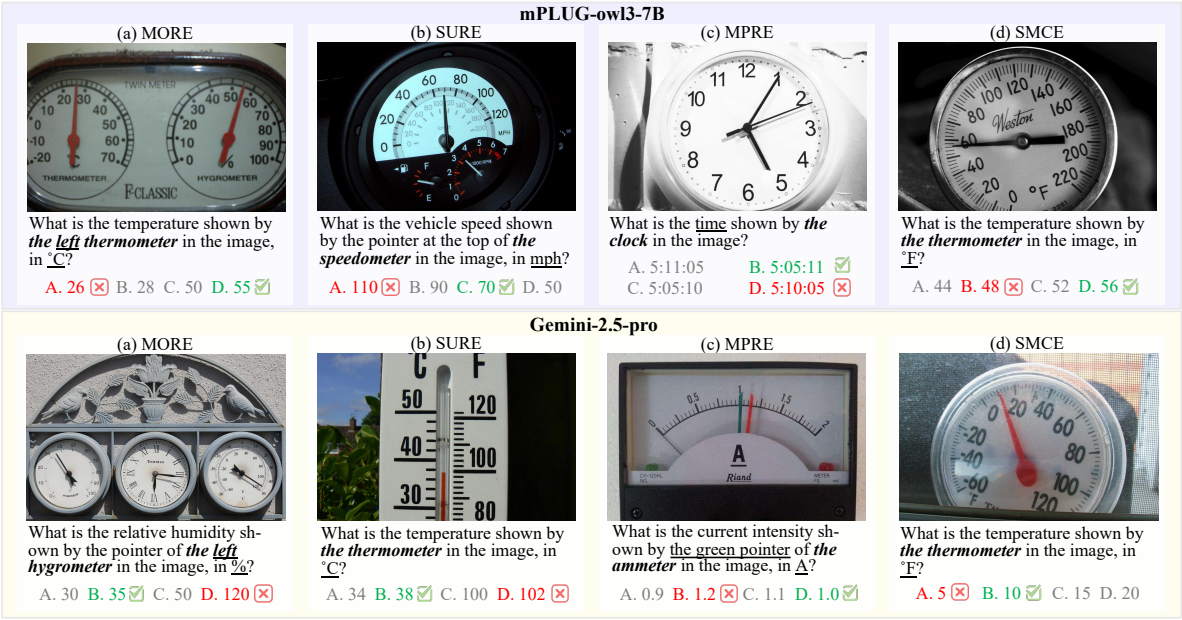

Figure 4: Error examples.

models in Table 3, namely mPLUG-owl3-7B and Gemini-2.5-Pro. After manual review, the errors are grouped into four main types and others: 1) Measurement Object Recognition Error (MORE), where the model fails to identify the correct object in the image, even though its position is clearly described in the question. 2) Scale Unit Recognition Error (SURE), in which the model confuses scale units when multiple are present, despite the unit being mentioned explicitly. 3) Multiple Pointer Recognition Error (MPRE), where the model cannot correctly interpret multiple pointers on tools like clocks or gauges. 4) Scale Meaning Comprehension Error (SMCE), where the model misunderstands what each unit or grid represents—for example, mistaking 2°F per division as 1°F. Other errors include random guesses or confusion about scale precision. As shown in Figure 3, SURE is the most frequent error type, followed by MPRE, while MORE and SMCE occur at similar rates. These results show that MLLMs struggle with different aspects of scale understanding. To illustrate each type more clearly, representative examples are provided in Figure 4.

7 Conclusion

In this paper, we propose ScaleBench, a manually annotated benchmark for scale recognition. To address the limitations of existing benchmarks, ScaleBench includes a broad range of physical quantities and measurement objects from everyday life, featuring diverse scale units and visual forms. Based on this benchmark, we evaluate a series of open-source and closed-source MLLMs and conduct extensive experimental analyses. Results show that existing MLLMs still struggle with ScaleBench, highlighting its difficulty and the need for further research in this area.

Limitations

Although ScaleBench is a valuable benchmark for evaluating the scale recognition ability of current MLLMs, it still has two limitations. First, the overall size of the dataset remains limited. To ensure high quality, we use a strict and detailed manual annotation process. However, this process required a lot of human effort, which limited the total number of samples. Second, the coverage of physical quantities and measurement objects is not yet complete. Although the dataset includes more common types than previous benchmarks, some less common physical quantities and measurement objects in real life are still missing.

Ethical Statement

We construct ScaleBench using data from the COCO dataset, the Open Images dataset, and the Flickr platform. Both COCO and Open Images are distributed under the Creative Commons Attribution 4.0 License, which allows redistribution and re-annotation with proper attribution. For images from Flickr, we carefully selected only those licensed under CC BY 4.0 or clearly free of copyright restrictions. Based on these sources, we release ScaleBench under the CC BY 4.0 license. In addition, we manually review all data to ensure that the dataset does not contain any harmful content, such as gender bias, racial discrimination, or other inappropriate material.

Acknowledgments

This paper was supported by the National Natural Science Foundation of China (No. 62306112) and Guangdong Basic and Applied Basic Research Foundation (No. 2026A1515010253).

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Banghao Chen, Zhaofeng Zhang, Nicolas Langrené, and Shengxin Zhu. 2025a. Unleashing the potential of prompt engineering for large language models. *Patterns*.

Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. 2025b. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198.

Shengyuan Ding, Shenxi Wu, Xiangyu Zhao, Yuhang Zang, Haodong Duan, Xiaoyi Dong, Pan Zhang, Yuhang Cao, Dahua Lin, and Jiaqi Wang. 2025. Mmifengine: Towards multimodal instruction following. *arXiv preprint arXiv:2504.07957*.

Zhipeng Dong, Yi Gao, Yunhui Yan, and Fei Chen. 2021. Vector detection network: An application study on robots reading analog meters in the wild. *IEEE Transactions on Artificial Intelligence*, 2(5):394–403.

Sivan Doveh, Shaked Perek, M Jehanzeb Mirza, Wei Lin, Amit Alfassy, Assaf Arbelle, Shimon Ullman, and Leonid Karlinsky. 2024. Towards multimodal in-context learning for vision and language models. In *European Conference on Computer Vision*, pages 250–267. Springer.

Jiajun Feng, Haibo Luo, and Rui Ming. 2025. Pointer meters recognition method in the wild based on innovative deep learning techniques. *Scientific Reports*, 15(1):845.

Pei Fu, Tongkun Guan, Zining Wang, Zhentao Guo, Chen Duan, Hao Sun, Boming Chen, Jiayao Ma, Qianyi Jiang, Kai Zhou, et al. 2025. Multimodal large language models for text-rich image understanding: A comprehensive review. *arXiv preprint arXiv:2502.16586*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Liqun Hou, Sen Wang, Xiaopeng Sun, and Guopeng Mao. 2023. A pointer meter reading recognition method based on yolox and semantic segmentation technology. *Measurement*, 218:113241.

Ben Howells, James Charles, Roberto Cipolla, et al. 2021. Real-time analogue gauge transcription on mobile phone. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2369–2377.

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. *arXiv preprint arXiv:2106.09685*.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. 2020. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International journal of computer vision*, 128(7):1956–1981.

Lin Li, Guikun Chen, Hanrong Shi, Jun Xiao, and Long Chen. 2024. A survey on multimodal benchmarks: In the era of large ai models. *arXiv preprint arXiv:2409.18142*.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597.

Zongxia Li, Xiyang Wu, Hongyang Du, Huy Nghiem, and Guangyao Shi. 2025. Benchmark evaluations, applications, and challenges of large vision language models: A survey. *arXiv preprint arXiv:2501.02189*, 1.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13*, pages 740–755. Springer.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023a. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.

Jingping Liu, Ziyan Liu, Zhedong Cen, Yan Zhou, Yinan Zou, Weiyan Zhang, Haiyun Jiang, and Tong Ruan. 2025. Can multimodal large language models understand spatial relations? *arXiv preprint arXiv:2505.19015*.

Jingping Liu, Xueyan Wu, Hanxuan Chen, Ziyan Liu, Zhangquan Chen, Ronghao Chen, and Huacan Wang. 2026. Easy for children, hard for ai: The limits of multimodal llms in early childhood learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 32078–32086.

Weihao Liu, Fangyu Lei, Tongxu Luo, Jiahe Lei, Shizhu He, Jun Zhao, and Kang Liu. 2023b. Mmhqa-icl: Multimodal in-context learning for hybrid question answering over text, tables and images. *arXiv preprint arXiv:2309.04790*.

Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, et al. 2024. Deepseek-vl: towards real-world vision-language understanding. *arXiv preprint arXiv:2403.05525*.

Maurits Reitsma, Julian Keller, Kenneth Blomqvist, and Roland Siegwart. 2024. Under pressure: learning-based analog gauge reading in the wild. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 14–20. IEEE.

Yan Shu, Shaohui Liu, Honglei Xu, and Feng Jiang. 2023. Read pointer meters based on a human-like alignment and recognition algorithm. In *CCF National Conference of Computer Applications*, pages 162–178. Springer.

Mustafa Shukor, Alexandre Rame, Corentin Dancette, and Matthieu Cord. 2023. Beyond task performance: Evaluating and reducing the flaws of large multimodal models with in-context learning. *arXiv preprint arXiv:2310.00647*.

Minjun Son and Sungjin Lee. 2025. Advancing multimodal large language models: Optimizing prompt engineering strategies for enhanced performance. *Applied Sciences*, 15(7):3992.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Jiaqi Wang, Hanqi Jiang, Yiheng Liu, Chong Ma, Xu Zhang, Yi Pan, Mengyuan Liu, Peiran Gu, Sichen Xia, Wenjun Li, et al. 2024. A comprehensive review of multimodal large language models: Performance and challenges across different tasks. *arXiv preprint arXiv:2408.01319*.

Xuang Wu, Xiaobo Shi, Yongchao Jiang, and Jun Gong. 2021. A high-precision automatic pointer meter reading system in low-light environment. *Sensors*, 21(14):4891.

Charig Yang, Weidi Xie, Andrew Zisserman, et al. 2022. It's about time: Analog clock reading in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2508–2517.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. 2024. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*.

Jiabo Ye, Haiyang Xu, Haowei Liu, Anwen Hu, Ming Yan, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. 2024. mplug-owl3: Towards long image-sequence understanding in multi-modal large language models. *arXiv preprint arXiv:2408.04840*.

Ming Zhang, Ke Chang, and Yunfang Wu. 2024. Multi-modal semantic understanding with contrastive cross-modal feature alignment. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 11934–11943.

Table 7: Model performance (Acc %) on ScaleBench_V.

Model	Setting	Choice	Blank	Total	Setting	Choice	Blank	Total
Open-source MLLMs								
DeepSeek-VL-7B	-	27.19	4.73	16.21	LoRA	**51.46**	6.56	29.50
MiniCPM-V-8B	-	30.26	20.61	25.54	LoRA	40.50	29.01	34.88
LLama-3.2-11B-Vision	-	27.49	18.02	22.85	LoRA	37.13	26.41	31.89
LLaVA-v1.6-7B	-	17.54	17.40	17.48	LoRA	44.88	27.48	36.37
mPLUG-owl3-7B	-	30.85	17.25	24.20	LoRA	44.44	23.21	34.06
Phi-3.5-vision-4B	-	20.61	19.39	20.01	LoRA	37.72	25.04	31.52
Janus-Pro-7B	-	31.29	3.36	17.63	LoRA	23.10	24.43	23.75
Qwen2.5-VL-7B	-	31.87	26.41	29.20	LoRA	43.27	29.47	36.52
InternVL3-8B	-	32.60	19.69	26.29	LoRA	36.70	25.50	31.22
Closed-source MLLMs								
Gemini-2.5-pro	0-shot	**44.20**	**35.00**	**39.60**	2-shot	44.00	**34.00**	39.00
	1-shot	43.40	31.20	37.30	3-shot	50.00	32.00	**41.00**
GPT-5	0-shot	38.00	29.60	33.80	2-shot	35.00	31.00	33.00
	1-shot	39.60	25.40	32.50	3-shot	42.00	30.00	36.00
Other Methods								
human	-	98.40	96.20	97.30	-	-	-	-

Table 8: Statistics of the benchmark variant containing both multiple-choice and fill-in-the-blank questions.

	Choice	Blank	Total
ScaleBench	3297	3,277	6574
Physical quantity types			
angle	153	153	306
humidity	85	84	169
length	651	648	1,299
pressure	483	481	964
speed	193	192	385
temperature	940	939	1,879
time	323	321	644
volume	318	315	633
weight	60	59	119
others	91	85	176

A Results on Mixed-Format Benchmark

We convert part of ScaleBench into fill-in-the-blank questions, creating a benchmark with 3,297 multiple-choice and 3,277 fill-in-the-blank questions, which we refer to as ScaleBench_V. In our paper, the "fill-in-the-blank" setting actually means an "open-ended QA" task. The training, validation, and test sets follow the same 7:1:2 split as in the previous experiments. We then evaluate all baseline methods on this benchmark, with results shown in Table 7, while detailed dataset statistics are also reported in Table 8. Overall, all models still perform poorly, with accuracies on multiple-choice and fill-in-the-blank questions far below the human levels of 98.40% and 96.20%. Moreover, most models, whether open-source (with or without fine-tuning) or closed-source MLLMs, consistently achieve higher accuracy on multiple-choice questions than on fill-in-the-blank ones, likely because the provided options act as cues that guide the models toward better answers.

B Question Templates

To improve the efficiency of benchmark annotation, we design a question template for each physical quantity. Using this template, annotators only need to fill in the corresponding quantity unit and measurement object based on the given image. The detailed template is presented in Table 9.

C Prompt used in LLMs

In the experiments, we use open-source MLLMs as baselines, with their task prompts shown in Table 10. We also evaluate closed-source MLLMs, and their prompts are listed in Table 11.

D Verification of Domain Knowledge

To rule out the possibility that models fail the scale recognition task due to insufficient domain knowledge (e.g., not recognizing measurement objects or understanding how to read them), we conduct a verification experiment using two models: mPLUG-owl3-7B and Gemini-2.5-Pro. Specifically, we randomly select 200 images from ScaleBench and apply a unified prompt for all queries, as shown in Fig-

Table 9: Question templates for each physical quantity.

Quantity	Question template
angle	What is the degree that *[measurement object]* in the image is pointing to?
humidity	What is the relative humidity shown by *[measurement object]* in the image, in %?
length	What is the length shown by *[measurement object]* in the image, in *[quantity unit]*?
pressure	What is the pressure shown by *[measurement object]* in the image, in *[quantity unit]*?
speed	What is the speed shown by the *[measurement object]* in the image, in *[quantity unit]*?
temperature	What is the temperature shown by *[measurement object]* in the image, in *[quantity unit]*?
time	What is the time shown by *[measurement object]* in the image?
volume	What is the volume of *[measurement object]* in the image, in *[quantity unit]*?
weight	What is the weight shown by *[measurement object]* in the image, in *[quantity unit]*?
current (I)	What is the current intensity shown by *[measurement object]* in the image, in *[quantity unit]*?
voltage	What is the voltage shown by *[measurement object]* in the image, in *[quantity unit]*?
acidity	What is the PH value shown by the *[measurement object]* in the image?
sound	What is the sound intensity shown by *[measurement object]* in the image, in *[quantity unit]*?

Table 10: Prompt used in open-source MLLMs.

Model	Prompt
DeepSeek-VL-7B MiniCPM-V-8B Llama-3.2-11B-V LLaVA-v1.6-7B mPLUG-owl3-7B Phi-3.5-vision-4B Janus-Pro-7B Qwen2.5-VL-7B InternVL3-8B	You are currently a senior expert in scale recognition. \n Given an Image, a Question and Options, your task is to identify the scale value and select the correct option. Note that you only need to choose one option from all options without explaining any reason. \n Input: Image: <image>, Question: {question}, Options: {options}. \n Output:

ure 5. Human evaluators then check whether each model provides the correct answers to determine if it possesses the necessary domain knowledge. The results show that mPLUG-owl3-7B and Gemini-2.5-Pro achieve accuracies of 94% and 98%, respectively, confirming that current models generally have sufficient domain knowledge to recognize and interpret different measurement objects.

E Results of Additional Models

We further report supplementary results for models across different parameter scales (4B–12B), including Qwen3-VL-4B, InternVL3.5-4B, Qwen3-VL-8B, and Pixtral-12B. Table 12 presents the results of additional models under zero-shot and LoRA settings. Overall, performance tends to improve as model scale increases within the QwenVL and InternVL families, and LoRA consistently leads to better results than zero-shot.

Prompt used in Verification

You will be shown an image containing a measuring instrument. Please complete the following two core tasks based on the image I provided, and provide a clear and detailed reasoning process for each task:

Task 1: Tool Identification

Determine the specific name of the graduated tool shown in the image (e.g., ruler, graduated cylinder, thermometer, vernier caliper, stopwatch, etc.), and explain the key basis for your judgment (such as the tool's shape, scale unit, structural features, etc.).

Task 2: Reading Method Explanation

If you can identify the tool type, fully elaborate on the standard reading steps of the tool (must include key points such as "how to confirm the measuring range", "how to find the division value", "requirements for line of sight", "whether estimation is needed and the estimation rules"); if you cannot identify it, explain the reason for the inability to judge.

Figure 5: Prompt used in the verification of domain knowledge.

F Effect of Input Resolution

We further examine Qwen3-VL-8B across input resolutions from 224×224 to 1120×1120. As shown in Table 13, the results remain relatively stable across resolutions, suggesting that the model is generally robust to input scaling. LoRA consistently outperforms zero-shot under all settings.

Table 11: Prompt used in Closed-source MLLMs.

Models	**Task prompt**
Zero-shot	You are currently a senior expert in scale recognition. \n Given an Image, a Question and Options, your task is to identify the scale value and select the correct option. Note that you only need to choose one option from all options without explaining any reason. \n Input: Image: <image>, Question: {question}, Options: {options}. \n Output:
Few-shot	You are currently a senior expert in scale recognition. \n Given an Image, a Question and Options, your task is to identify the scale value and select the correct option. Note that you only need to choose one option from all options without explaining any reason. \n Given the following 3 examples to learn the visual scale recognition task \n Example 1: Input: Image: <image>\n Question: What is the temperature shown by the thermometer in the image, in degrees Celsius? Options: 34; 35; 36; 37. \n Output: 36. \n Example 2: ... \n Example 3: Input: Image: <image>\n Question: {question}, Options: {options}. \n Output:

Table 12: Additional model performance on ScaleBench

Model	**Setting**	**P**	**R**	**F1**	**Acc**	**Setting**	**P**	**R**	**F1**	**Acc**
Qwen3-VL-4B	-	28.81	27.00	27.87	27.33	LoRA	33.73	33.94	33.83	34.20
InternVL3.5-4B	-	30.49	29.33	29.90	29.20	LoRA	38.98	35.42	37.11	35.32
Qwen3-VL-8B	-	30.94	29.10	29.99	29.65	LoRA	39.71	39.42	39.56	39.88
Pixtral-12B	-	33.08	30.70	31.84	29.65	LoRA	39.62	37.83	38.70	38.31

Table 13: Qwen3-VL-8B on ScaleBench under different input resolutions. "Default" denotes the main setting.

Resolution	**Setting**	**P**	**R**	**F1**	**Acc**	**Setting**	**P**	**R**	**F1**	**Acc**
224×224	-	31.76	29.15	30.40	29.65	LoRA	39.56	38.61	39.08	38.91
448×448	-	31.93	29.84	30.85	30.40	LoRA	40.24	39.70	39.97	40.10
672×672	-	31.36	28.39	29.80	28.90	LoRA	41.24	40.02	40.62	40.48
896×896	-	29.66	27.90	28.75	28.45	LoRA	40.64	39.96	40.30	40.48
1120×1120	-	30.54	28.61	29.55	29.13	LoRA	40.34	39.54	39.94	40.03
Default	-	30.94	29.10	29.99	29.65	LoRA	39.71	39.42	39.56	39.88