

A Free-form Generation

As mentioned in the **Limitations** paragraph (Sec. 5), multiple-choice questions can elicit proactiveness since proactive options are already listed among others. This section evaluates multimodal LLMs on ProactiveBench using open-ended answers, avoiding biasing models from proposing proactive suggestions. Therefore, we only provide the MLLM with the image frame and the question that the model should answer.

Evaluation protocol. As evaluating free-form answers is challenging, we follow previous works [16, 40, 46, 47, 53, 58, 66] and employ LLM-as-a-judge to provide a score to each answer. In particular, we use OpenAI’s o4-mini [54] and prompt it to spot proactive suggestions and category predictions. The following system and user prompt were used to query the judge:

System Prompt:

You are an automatic evaluation system.

You will receive:

- A user prompt (describing the task or question)
- A list of correct answers (accepted correct outputs)
- A system output (the model’s generated answer)

Your task is to evaluate whether the system output includes at least one of the correct answers clearly and correctly.

Guidelines:

- Minor wording differences (e.g., paraphrasing) are acceptable as long as the meaning is preserved.
- If a correct answer is present but accompanied by incorrect or irrelevant content, still consider it correct (score = 1).
- If none of the correct answers are present, or the output is incorrect, mark it as wrong (score = 0).

Respond with a comma-separated list of 0s and 1s, one for each correct answer in the list.\n\n

User Prompt:

Input:

User Prompt: {user_prompt}

Correct Answers: {correct_answer}

System Output: {generated_answer}

Your Response: (Only output comma-separated list of 0s and 1s)

Thus, as answers are usually long, the LLM-as-a-judge is tasked to spot whether in the answer there are correct proactive suggestions and correct category predictions, respectively defined in {correct_answer}, returning comma-separated values, with one digit for each correct answer, containing both proactive suggestions and the right category. Evaluating multi-turn conversations with free-form answers, however, is highly impractical since it requires alternating the answer generation with the MLLM and the evaluation with LLM-as-a-judge to update the environment state. Therefore, we limited this evaluation to single-turn conversations. Furthermore, to reduce the evaluation cost, we subsample each dataset to 500 entries, except ROD and VSOD, as they have smaller dimensions. We report the ratio of correctly predicted categories (*acc*), the ratio of correct proactive suggestions (*pa*), and the aggregate accuracy (*agg*), computed as the average accuracy in either predicting the correct answer or providing a valid proactive suggestion.

Results. Table 2 shows the model’s performance in open-ended generation on ProactiveBench. We limit this evaluation to three low-performing (i.e., LLaVA-OV 0.5B, Qwen2.5-VL 3B, InternVL3 1B) and three high-performing (i.e., LLaVA-OV 7B, Qwen2.5-VL 7B, InternVL3 8B) models. Overall, high-performing models tend to outperform low-performing ones, as one would expect, with each MLLM outperforming its smaller counterparts (e.g., LLaVA-OV 0.5B is outperformed by LLaVA-OV 7B). Contrary to the multiple-choice scenario, low-performing models do not show higher proactiveness, with only InternVL3 1B proposing few correct proactive suggestions (i.e., 0.1% on avg.). By hinting at proactive suggestions (i.e., If you cannot answer this question, please tell

me what I should do to help you.), the proactiveness increases for all MLLMs, with high-performing models showing significantly higher absolute growths (e.g., Qwen2.5-VL 7B +10.5% vs. Qwen2.5-VL 3B +3.2%). This further validates the main paper findings, highlighting that low-performing multimodal LLMs are indeed not more proactive than high-performing ones (Sec. 3.3); rather, when in doubt, they are more prone to choose random options in the multiple-choice scenario. Finally, we also notice that by hinting, models become more cautious in answering questions, showing lower ratios of correct predictions (e.g., from 15.2% to 12.7% of Qwen2.5-VL 7B).

Table 2: **Open-ended generation evaluation on ProactiveBench.** We report the aggregate accuracy (*agg*), the ratio of correctly predicted categories (*acc*), and the ratio of correct proactive suggestions (*pa*) for all datasets, with global averages in the last column.

	model	ROD			VSOD			MVP-N			IN-C		
		agg	acc	pa	agg	acc	pa	agg	acc	pa	agg	acc	pa
low-perf	LLaVA-OV-0.5B [32]	0.0	0.0	0.0	6.3	6.3	0.0	0.0	0.0	0.0	10.6	10.6	0.0
	Qwen2.5-VL-3B [5]	0.0	0.0	0.0	11.1	11.1	0.0	0.0	0.0	0.0	11.4	11.4	0.0
	InternVL3-1B [83]	0.0	0.0	0.0	7.9	7.9	0.0	0.0	0.0	0.0	12.4	12.4	0.0
high-perf	LLaVA-OV-7B [32]	2.3	0.0	2.3	7.9	7.9	0.0	0.0	0.0	0.0	11.2	11.0	0.2
	Qwen2.5-VL-7B [5]	1.2	0.0	1.2	11.1	11.1	0.0	0.0	0.0	0.0	18.2	18.2	0.0
	InternVL3-8B [83]	0.0	0.0	0.0	11.1	11.1	0.0	0.0	0.0	0.0	16.0	16.0	0.0
	model	QD			CIT			COCO			avg.		
	agg	acc	pa	agg	acc	pa	agg	acc	pa	agg	acc	pa	
low-perf	LLaVA-OV-0.5B [32]	6.4	6.4	0.0	13.4	13.4	0.0	38.2	38.2	0.0	10.7	10.7	0.0
	Qwen2.5-VL-3B [5]	6.6	6.6	0.0	28.0	28.0	0.0	33.3	33.3	0.0	12.9	12.9	0.0
	InternVL3-1B [83]	3.4	3.0	0.4	29.4	29.4	0.0	45.6	45.6	0.0	14.1	14.0	0.1
high-perf	LLaVA-OV-7B [32]	11.6	10.8	0.8	19.4	19.4	0.0	42.6	42.6	0.0	13.6	13.1	0.5
	Qwen2.5-VL-7B [5]	8.8	7.4	1.4	32.0	31.6	0.4	38.3	38.3	0.0	15.7	15.2	0.4
	InternVL3-8B [83]	3.4	3.4	0.0	33.2	33.2	0.0	45.4	45.4	0.0	15.6	15.6	0.0

Table 3: **Open-ended generation evaluation on ProactiveBench by hinting at proactive suggestions.** We report the aggregate accuracy (*agg*), the ratio of correctly predicted categories (*acc*), and the ratio of correct proactive suggestions (*pa*) for all datasets, with global averages in the last column.

	model	ROD			VSOD			MVP-N			IN-C			
		agg	acc	pa	agg	acc	pa	agg	acc	pa	agg	acc	pa	
high-perf	low-perf	LLaVA-OV-0.5B [32]	1.1	0.0	1.1	6.3	6.3	0.0	0.0	0.0	0.0	8.2	8.2	0.0
		Qwen2.5-VL-3B [5]	5.7	0.0	5.7	6.3	6.3	0.0	0.2	0.0	0.2	17.8	9.4	8.4
		InternVL3-1B [83]	0.0	0.0	0.0	11.1	11.1	0.0	0.0	0.0	0.0	12.0	9.8	2.2
	high-perf	LLaVA-OV-7B [32]	0.0	0.0	0.0	4.8	4.8	0.0	0.4	0.0	0.4	28.2	8.4	20.0
		Qwen2.5-VL-7B [5]	1.1	0.0	1.1	7.9	6.3	1.6	1.2	0.0	1.2	40.2	15.4	25.2
		InternVL3-8B [83]	27.3	0.0	27.3	6.3	6.3	0.0	0.6	0.0	0.6	30.4	13.0	18.4
low-perf	model		QD			CIT			COCO			avg.		
			agg	acc	pa	agg	acc	pa	agg	acc	pa	agg	acc	pa
		LLaVA-OV-0.5B [32]	7.6	7.4	0.2	12.4	12.4	0.0	39.2	39.2	0.0	10.7	10.5	0.2
	low-perf	Qwen2.5-VL-3B [5]	14.2	6.6	8.2	24.4	24.2	0.2	28.4	28.4	0.0	13.9	10.7	3.2
		InternVL3-1B [83]	10.2	3.7	6.5	25.6	25.6	0.0	43.2	43.2	0.0	14.6	13.3	1.2
		high-perf	LLaVA-OV-7B [32]	23.5	8.9	14.8	18.6	18.6	0.0	31.8	30.6	1.2	15.3	10.2
	Qwen2.5-VL-7B [5]		42.5	6.8	39.1	33.4	26.8	6.8	34.4	33.3	1.4	23.0	12.7	10.9
	InternVL3-8B [83]		25.8	3.2	23.3	34.4	32.0	2.6	45.0	44.8	0.2	24.3	14.2	10.3

B Dataset Details and Environment Implementation

This section expands Secs. 2.1 and 2.2, providing further information about data generation pipelines and environment details.

B.1 The ROD Environment

The ROD [31] environment evaluates MLLMs’ proactiveness in proposing to move occluding objects before answering the question. The first frame in the ROD environment depicts an occluding object that completely hides another object, as Fig. 20 shows. Each MLLM is prompted to predict the category of the occluded object, choosing out of four possible categories, and

the abstain option. As the posed question is unanswerable from the initial frame, given that the subject of the question is invisible, the environment also returns two valid proactive suggestions among other options, i.e., move the {occluding_object} to the left, and move the {occluding_object} to the right, where {occluding_object} is replaced with the occluding object description (e.g., red cardboard, blue blocks). Furthermore, we also consider camera movement a valid proactive suggestion in the free-form evaluation experiments. A typical prompt is structured as follows:

Could you tell me what is behind the {occluding_object}? <hint> Choose from the following options. Options:
A. Move the {occluding_object} to the left.
B. Move the {occluding_object} to the right.
C. {abstain option}.
D. {wrong random category}.
E. {wrong random category}.
F. {correct category}.
G. {wrong random category}.
Please only return one of the options without any other words.

The question is sampled from a pool of 15 similar questions generated by ChatGPT, and the abstain option is from a pool of three. Additionally, the first three options and the remaining four are shuffled, so the same option does not always appear in the same position. Shuffling is performed during data generation, resulting in a fixed order for each sample. Finally, <hint> indicates the position of the hint used in the main paper experiments (Sec. 3.3), which, in the case of ROD, corresponds to “Hint: moving the occluding object might reveal what is behind it.”

The set of valid actions \mathcal{A}_t is constant throughout the evaluation, and MLLMs are allowed to move the occluding object 14 times, corresponding to the total number of frames for each sample. As the first frame is completely occluded, if a model predicts a category for the first frame, we count the prediction as wrong, as the first frame does not contain information about the target object class. After seven consecutive right or left movements from the most occluded frame, MLLMs encounter the reference frame, where the object is perfectly visible. Finally, the environment is circular, which means that by pursuing the same proactive suggestion, the occluding object will reveal the object until it reappears from the opposite side, gradually re-occluding the object.

B.2 The VSOD Environment

The VSOD environment evaluates MLLMs’ proactiveness in proposing to wait or rewind the video before answering the question, in case of occlusions. The first frame in this environment depicts a scene where individuals are occluded by someone passing in front of the camera, as Fig. 21 shows. Each MLLM is prompted to predict the speaker’s name, the number of people, or the event type, choosing out of four possible categories, and the abstain option. As the posed question is likely unanswerable from the initial frame, given that the subject of the question is (partially) invisible, the environment also returns two valid proactive suggestions among other options, i.e., wait for the occlusion to disappear, and rewind the video. Furthermore, we also consider camera movement a valid proactive suggestion in the free-form evaluation experiments. A typical prompt is structured as follows:

This is a frame extracted from a video. Answer the following question.
Could you tell me who is talking? <hint> Choose from the following options.
Options:
A. Rewind the video.
B. {abstain option}.
C. Wait for the occlusion to disappear.
D. {wrong random category}.
E. {correct category}.
F. {wrong random category}.
G. {wrong random category}.
Please only return one of the options without any other words.

In this prompt, the question is sampled from a pool of 45 similar questions (15 for each question type), and the abstain option is from a pool of three. Additionally, the first three options and the remaining four are shuffled, so the same option does not always appear in the same position. Shuffling is performed during data generation, resulting in a fixed order for each sample. Finally, `<hint>` indicates the position of the hint used in the main paper experiments (Sec. 3.3), which, in the case of VSOD, corresponds to “Hint: If there is an occlusion, waiting for it to disappear or rewinding the video might reveal what’s behind it.”

The set of valid actions \mathcal{A}_t is constant throughout the evaluation, and MLLMs are allowed to propose proactive suggestions as many times as the number of frames in the video. As each occlusion lasts for a different amount of time, the number of proactive suggestions to reach a state where the question becomes answerable varies from sample to sample. Finally, if the MLLM suggests waiting at the last frame, we treat the sequence as circular and return the first frames. Analogously, we return the final frame if, at the first frame, the model suggests rewinding the video.

B.3 The MVP-N Environment

The MVP-N environment evaluates MLLMs’ proactiveness in suggesting objects and camera rotations before answering the question in case of uninformative views. The first frame in the MVP-N environment depicts an object from an uninformative viewpoint, as Fig. 22 shows. Each MLLM is prompted to predict the category of the object, choosing out of four possible categories, and the abstain option. As the posed question is unanswerable from the initial frame, given that discriminative object features are invisible, the environment also returns a valid proactive suggestion among other options, e.g., rotate the object, give me a view of the object from a different perspective. As object orientation and camera extrinsic parameters are not annotated, the proactive suggestion is sampled from a pool of 11 prompts generated with ChatGPT that contain both object rotations and camera movements. A typical prompt is structured as follows:

```
Identify the object in this image. <hint> Choose from the following options.
Options:
A. {abstain option}.
B. {proactive suggestion}.
C. {wrong random category}.
D. {correct category}.
E. {wrong random category}.
F. {wrong random category}.
Please only return one of the options without any other words.
```

In this prompt, the question is sampled from a pool of 15 similar questions, and the abstain option is from a pool of three. Additionally, the first two options and the remaining four are shuffled, so the same option does not always appear in the same position. Shuffling is performed during data generation, resulting in a fixed order for each sample. Finally, `<hint>` indicates the position of the hint used in the main paper experiments (Sec. 3.3), which, in the case of MVP-N, corresponds to “Hint: rotating the object could provide a more informative view.”

The set of valid actions \mathcal{A}_t is constant throughout the evaluation, and, since we generated sequences of various lengths, MLLMs are allowed to rotate the object or change camera angle 3 times on average for each sample, depending on the sequence. To find the informative view, MLLMs must propose object rotations or camera movements until they reach the last state, where object features make it distinguishable.

B.4 The ImageNet-C Environment

The ImageNet-C environment evaluates MLLMs’ proactiveness in suggesting image quality improvements before answering the question, in case of badly corrupted pictures. The first image in the ImageNet-C environment depicts one of ImageNet [59] validation samples strongly corrupted by one of 16 different corruptions, as Fig. 23 shows. Each MLLM is prompted to predict the category of the corrupted object, choosing out of four possible categories, and the abstain option. As the posed question is hardly answerable from the initial picture, the environment also returns four proactive sug-

gestions, out of which only one is valid, e.g., deblur the image, denoise the image, remove artifacts. For example, a typical prompt is structured as follows:

```
What type of object do you see here? <hint> Choose from the following
options. Options:
A. {invalid proactive suggestion}.
B. {abstain option}.
C. {valid proactive suggestion}.
D. {invalid proactive suggestion}.
E. {invalid proactive suggestion}.
F. {wrong random category}.
G. {correct category}.
H. {wrong random category}.
I. {wrong random category}.
Please only return one of the options without any other words.
```

In this prompt, the question is sampled from a pool of 15 similar questions, and the abstain option is from a pool of three. Additionally, the first five options and the remaining four are shuffled, so the same option does not always appear in the same position. Shuffling is performed during data generation, resulting in a fixed order for each sample. Finally, `<hint>` indicates the position of the hint used in the main paper experiments (Sec. 3.3), which, in the case of ImageNet-C, corresponds to “Hint: enhancing the image quality could help with classification.”

As ImageNet-C counts 50,000 images, we subsampled 5 images per class, resulting in 5,000 images, making this dataset comparable in size to the others used. The set of valid actions \mathcal{A}_t is constant throughout the evaluation, and MLLMs are allowed to propose the correct proactive suggestion 4 times, improving the image quality. After 4 proactive suggestions, MLLMs encounter the last frame, the reference one. Further proactive suggestions result in terminating the evaluation.

B.5 The QuickDraw Environment

The QuickDraw environment evaluates MLLMs’ proactiveness in proposing to add details to a sketch, to make it more recognizable. The first image in the QuickDraw environment shows the first drawn stroke by a user in trying to depict a target object, as Fig. 24 shows. Each MLLM is prompted to predict the category of such depicted object, choosing out of four possible categories, and the abstain option. As the posed question is likely unanswerable from the initial drawing version, the environment also returns a valid proactive suggestion among other options, e.g., add more details, or could you improve the quickdraw? For example, a typical prompt is structured as follows:

```
What is the category of the depicted object? <hint> Choose from the
following options. Options:
A. {proactive option}.
B. {abstain option}.
C. {wrong random category}.
D. {wrong random category}.
E. {wrong random category}.
F. {correct category}.
Please only return one of the options without any other words.
```

In this prompt, the question is sampled from a pool of 15 similar questions, the abstain option is from a pool of three, and the proactive option is from a pool of 13. Additionally, the first two options and the remaining four are shuffled, so the same option does not always appear in the same position. Shuffling is performed during data generation, resulting in a fixed order for each sample. Finally, `<hint>` indicates the position of the hint used in the main paper experiments (Sec. 3.3), which, in the case of QuickDraw, corresponds to “Hint: Adding more details to the quickdraw could help with classification.”

As each drawing is also evaluated by a classification model [19], we discarded all drawings not recognized by such a model, avoiding unrecognizable drawings. Furthermore, the dataset contains 50 million drawings over 345 classes. Evaluating each MLLM would require approximately 300 GPU days. Thus, we subsample it to 10 samples per class, resulting in 3450 drawings. The set of

valid actions \mathcal{A}_t is constant throughout the evaluation, and MLLMs are allowed to ask for details a limited number of times, which depends on the number of strokes drawn by the user. Depending on the number of strokes, after requesting further details enough times, MLLMs encounter the reference frame, where the object is recognizable.

B.6 The ChangeIt Environment

The ChangeIt environment evaluates MLLMs’ proactiveness in proposing to seek the answer at a different moment in the video. The first frame in the ChangeIt environment shows the beginning of a video tutorial, as Fig. 25 shows. Each MLLM is prompted to either predict the category of the main object or the main action taken in the video, choosing out of four possible categories and the abstain option. As the posed question is likely unanswerable from the initial frame, the environment also returns two valid proactive suggestions among other options, i.e., wait for the occlusion to disappear, and rewind the video. For example, a typical prompt is structured as follows:

```
What action is being performed in the video? <hint> Choose from the
following options. Options:
A. Rewind the video.
B. Wait for the occlusion to disappear.
C. {abstain option}.
D. {wrong random category}.
E. {wrong random category}.
F. {wrong random category}.
G. {correct category}.
Please only return one of the options without any other words.
```

For this prompt, questions related to the object category are sampled from a pool of 15 similar questions, while those related to the action category are from a pool of 11 questions, all obtained by querying ChatGPT. The abstain option, instead, is sampled from a pool of three. Additionally, the first three options and the remaining four are shuffled, so the same option does not always appear in the same position. Shuffling is performed during data generation, resulting in a fixed order for each sample. Finally, <hint> indicates the position of the hint used in the main paper experiments (Sec. 3.3), which, in the case of ChangeIt, corresponds to “Hint: If you cannot answer the question, waiting for it to appear or rewinding the video could help with classification.”

The set of valid actions \mathcal{A}_t changes throughout the evaluation. Since the environment returns the initial frame first, the rewind option is disabled at the first frame and enabled from the second step. MLLMs can propose proactive suggestions as many times as the number of frames in the video. Finally, as each video differs, the number of proactive suggestions to reach a state where the question becomes answerable varies from sample to sample.

B.7 The MS-COCO Environment

The MS-COCO environment evaluates MLLMs’ proactiveness in proposing camera movements to obtain more informative cues. The first image in the MS-COCO environment shows a trimmed picture with missing object details, as in Fig. 26. Since most images in MS-COCO contain multiple objects, we discard all those samples that contain more than one object, avoiding ambiguities. Each MLLM is prompted to predict the category of the object in the image, choosing out of four possible categories and the abstain option. As the posed question is likely unanswerable from the initial frame, the environment also returns one or two valid proactive suggestions, depending on how the image crop was computed. Crops are generated to allow for exploration of one of the ordinal or cardinal directions or zooming out, the set of proactive actions, thus, changes based on the picture, i.e., move the camera up, move the camera down, move the camera left, move the camera right, move farther from the object. In the case of ordinal directions, MLLMs receive two proactive options, one for each of the cardinal directions that generate the ordinal one. Instead, for cardinal directions and zooming out, MLLMs receive only one. For example, a typical prompt for an ordinal direction is structured as follows:

```

Classify the visual content of this image. <hint> Choose from the following
options. Options:
A. Move the camera left.
B. Move the camera up.
C. {abstain option}.
D. {wrong random category}.
E. {wrong random category}.
F. {wrong random category}.
G. {correct category}.
Please only return one of the options without any other words.

```

For this prompt, the question is sampled from a pool of 15 similar questions obtained from querying ChatGPT, while the abstain option is sampled from a pool of three. Additionally, the first two/three options (depending on the direction) and the remaining four are shuffled, so the same option does not always appear in the same position. Shuffling is performed during data generation, resulting in a fixed order for each sample. Finally, `<hint>` indicates the position of the hint used in the main paper experiments (Sec. 3.3), which, in the case of MS-COCO, corresponds to “Hint: moving the camera could help with classification” for ordinal and cardinal directions and “Hint: zooming out could help with classification” for the zooming out case.

The set of valid actions \mathcal{A}_t changes throughout the evaluation for ordinal directions while it remains fixed for cardinal directions and the zooming out case. Since the camera can move in two of the four cardinal directions in the ordinal directions case, we remove a cardinal direction if the MLLM has already unveiled all possible object details in a specific direction, i.e., it has explored all discrete steps in a direction. Finally, MLLMs can propose proactive suggestions as many as the predefined discrete steps, set between 3 and 5.

C Extended Results

As most results could not fit within nine pages, the main paper summarizes key findings with plots. This section reports all tables associated with the main paper’s plots and the extended version of each plot, not limited to six models. Table 4 reports MLLMs oracle performance on ProactiveBench, showing that high-performing models tend to outperform low-performing ones. Figure 17 shows all models’ action distribution, further highlighting that low-performing models overweight proactive suggestions over the abstain option. InstructBLIP stands out, showing the lowest rate of proactive suggestions, despite being the worst MLLM in our evaluations. By inspecting its outputs, we notice that it easily fails to choose an answer from the provided options, resulting in a mismatch between valid options and the generated answer. Since we classify mismatched predictions as abstain options, this results in a high abstain rate for InstructBLIP. Table 5 and Fig. 18 respectively report MLLM’s results and the corresponding action distribution when proactive suggestions are replaced with random ones. Similarly, Tab. 6 and Fig. 19 describe MLLM’s results and action distribution on all models when the prompt hints at proactive suggestions. Finally, Tab. 7 and Tab. 8 display MLLM’s performance on ProactiveBench when conditioned on conversation histories and few-shot examples of proactive conversations.

Additional computational details. We conducted most experiments using a single A100 Nvidia GPU, 32GB of RAM, and 8 CPU cores, lasting about 1 hour, depending on the dataset. When conditioning on conversation histories and few-shot samples, we used two A100 GPUs to reduce the memory footprint of the models’ parameters, with experiments lasting about 2 hours on average and at most 8 hours, depending on the dataset and model. Furthermore, for ICL examples, we reduced the ROD image sizes of the few shots from 3024×3024 to 512×512 for Phi-4-Multimodal to avoid out-of-memory issues. We also reduced the sequence length of MVP-N ICL examples using 3 shots with Phi-4-Multimodal. Finally, we resized all samples’ short edge to 224px when conditioning on conversational histories to avoid out-of-memory issues with long sequences.

D Broader Impacts Statement

ProactiveBench is designed to assess the proactiveness of multimodal large language models (MLLMs), i.e., their ability to request additional input when faced with ambiguous or insuffi-

Table 4: **MLLMs oracle performance on ProactiveBench.** We report the accuracy in percentages (%) for all datasets, with global averages in the column.

	model	ROD	VSOD	MVP-N	IN-C	QD	CIT	COCO	avg.
<i>low-perf</i>	LLaVA-1.5-7B [40]	100.0	77.8	47.5	96.0	76.9	83.1	96.8	82.6
	LLaVA-NeXT-Mistral-7B [41]	100.0	60.3	52.6	95.5	71.4	81.9	98.2	80.0
	LLaVA-NeXT-Vicuna-7B [41]	98.9	66.7	49.4	96.3	62.3	82.7	98.1	79.2
	LLaVA-OV-0.5B [32]	100.0	47.6	58.9	92.3	81.8	84.2	98.1	80.4
	Qwen2.5-VL-3B [5]	100.0	82.5	57.3	97.3	73.0	89.5	98.6	85.5
	SmolVLM2-2.2B [48]	100.0	73.0	56.8	95.7	78.4	89.7	98.2	84.5
	Idefics3-8B [30]	100.0	81.0	60.0	96.7	71.3	87.7	98.5	85.0
	InternVL3-1B [83]	98.9	61.9	56.3	94.5	66.8	86.1	98.6	80.4
	InternVL3-2B [83]	100.0	82.5	58.4	97.6	71.8	88.7	99.1	85.4
<i>high-perf</i>	InstructBLIP [11]	75.0	52.4	26.6	37.1	33.4	66.2	27.0	45.4
	LLaVA-OV-7B [32]	100.0	82.5	66.0	98.5	89.7	90.8	99.2	89.5
	Qwen2.5-VL-7B [5]	100.0	84.1	66.7	98.5	80.6	90.9	98.9	88.5
	InternVL3-8B [83]	100.0	82.5	63.5	98.8	76.0	89.4	99.2	87.1
	Phi-4-Multimodal [1]	100.0	63.5	52.3	94.0	80.8	79.4	98.4	81.2

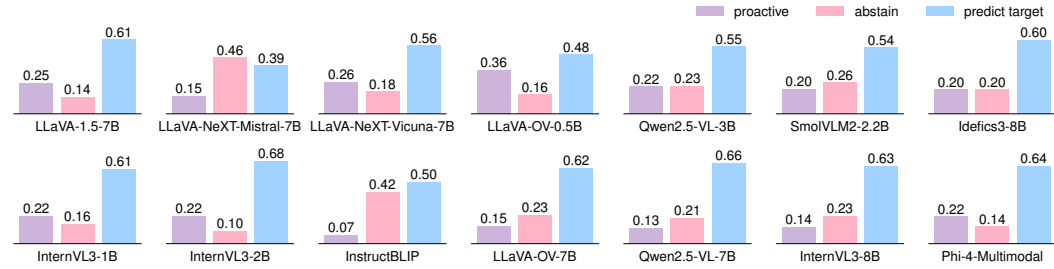


Figure 17: **Action distributions.** We report the action distribution for all evaluated models.

cient visual information. As MLLMs are increasingly deployed in interactive and safety-critical applications (i.e., assistive tools, autonomous systems), encouraging and evaluating such behavior is essential for developing more collaborative and user-aligned AI.

By highlighting current models’ proactiveness limitations, our work provides meaningful insights for researchers seeking to build more collaborative AI systems. However, promoting proactiveness must be carefully balanced to avoid over-questioning or inefficient behavior. While our benchmark promotes interpretability and safe failure modes (i.e., abstention over hallucination), there is a risk of misuse in adversarial settings if models over-rely on user feedback. We release ProactiveBench to support reproducible and community-driven progress toward more robust and human-aware MLLMs.

E License

All original material presented in this work is intended solely for academic research and not for commercial purposes. Below, we report the licenses of the used datasets and models:

- ROD [31]: This dataset is released without a license.
- VSOD [38]: MIT License.
- MVP-N [71]: MIT License.
- ImageNet-C [23]: Apache License 2.0.
- QuickDraw [19]: CC-BY-4.0.
- ChangeIt [67]: MIT License.
- MS-COCO [39]: CC-BY-4.0.
- LLaVA-1.5 [41]: Llama2.
- LLaVA-NeXT Vicuna [41]: Llama2.
- LLaVA-NeXT Mistral [41]: Apache License 2.0.

Table 5: **MLLMs results on ProactiveBench with random proactive suggestions.** We report the accuracy (*acc*) in percentages (%) and average number of proactive suggestions (*ps*) for all datasets, with global averages in the last column.

model	ROD		VSOD		MVP-N		IN-C		QD		CIT		COCO		avg.		
	acc	ps	acc	ps	acc	ps	acc	ps	acc	ps	acc	ps	acc	ps	acc	ps	
low-perf	LLaVA-1.5-7B [40]	42.0	3.5	38.1	0.7	26.3	0.0	83.2	0.1	42.1	1.1	59.0	0.1	66.6	0.4	51.0	0.8
	LLaVA-NeXT-Mistral-7B [41]	0.0	0.1	12.7	0.0	10.0	0.0	55.6	0.0	13.5	0.1	33.8	0.0	51.0	0.1	25.2	0.0
	LLaVA-NeXT-Vicuna-7B [41]	43.2	3.0	34.9	0.0	24.1	0.1	73.9	0.1	18.1	1.4	58.3	0.1	67.4	0.2	45.7	0.7
	LLaVA-OV-0.5B [32]	29.5	2.0	20.6	0.9	23.4	0.4	58.4	0.1	39.0	0.7	42.1	0.2	66.2	0.2	39.9	0.6
	Qwen2.5-VL-3B [5]	6.8	0.2	31.7	0.0	25.2	0.0	69.5	0.0	29.5	0.0	54.6	0.0	57.9	0.0	39.3	0.0
	SmolVLM2-2.2B [48]	12.5	0.8	27.0	0.0	26.5	0.0	51.7	0.0	22.4	0.0	54.6	0.2	61.3	0.0	36.6	0.1
	Idefics3-8B [30]	19.3	0.5	38.1	1.8	26.6	0.0	68.7	0.0	27.3	0.3	56.7	0.1	63.2	0.1	42.8	0.4
	InternVL3-1B [83]	44.3	1.4	34.9	0.0	27.7	0.0	72.7	0.0	27.7	0.2	60.1	0.1	74.3	0.1	48.8	0.3
	InternVL3-2B [83]	17.0	0.3	47.6	0.0	29.7	0.0	80.8	0.0	34.8	0.1	65.3	0.1	76.0	0.0	50.2	0.1
	InstructBLIP [11]	0.0	0.7	19.0	2.0	2.9	0.6	46.2	0.1	17.3	0.5	34.2	0.2	25.0	0.4	20.7	0.6
high-perf	LLaVA-OV-7B [32]	1.1	0.0	36.5	0.4	25.9	0.0	69.5	0.0	39.2	0.0	54.0	0.1	59.6	0.0	40.8	0.1
	Qwen2.5-VL-7B [5]	2.3	0.0	36.5	0.0	25.6	0.0	76.7	0.0	36.4	0.0	59.9	0.0	61.0	0.0	42.6	0.0
	InternVL3-8B [83]	2.3	0.0	39.7	0.6	24.6	0.0	73.6	0.0	35.1	0.0	57.4	0.0	68.5	0.0	43.0	0.1
	Phi-4-Multimodal [1]	12.5	0.6	27.0	0.2	28.4	0.0	61.2	0.0	40.9	0.1	58.4	0.1	70.5	0.2	42.7	0.2

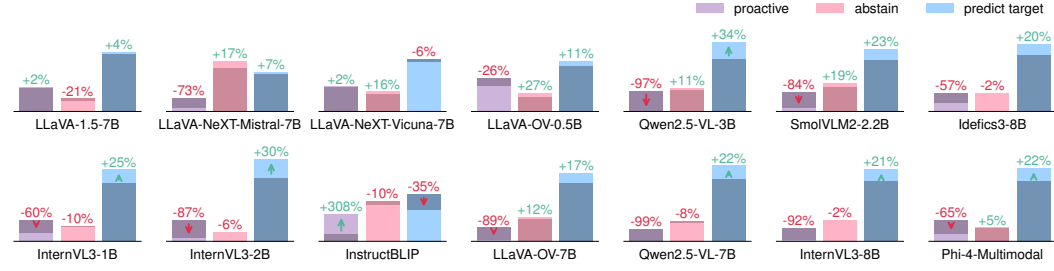


Figure 18: **Action distributions with random proactive options.** Lighter bars describe variations using random proactive suggestions for all evaluated models.

- LLaVA-OV [32]: Apache License 2.0.
- Qwen2.5-VL [5]: Apache License 2.0.
- SmolVLM2 [48]: Apache License 2.0.
- Idefics3 [30]: Apache License 2.0.
- InternVL3 [83]: Apache License 2.0.
- InstructBLIP [11]: Llama2.
- Phi-4-Multimodal [1]: MIT License.

F Dataset Examples

Figures 20 to 26 report dataset examples returned by the environment in the first state.

Table 6: **MLLMs results on ProactiveBench by hinting at proactive suggestions.** We report the accuracy (*acc*) in percentages (%) and average number of proactive suggestions (*ps*) for all datasets, with global averages in the last column.

model	ROD		VSOD		MVP-N		IN-C		QD		CIT		COCO		avg.		
	acc	ps	acc	ps	acc	ps	acc	ps	acc	ps	acc	ps	acc	ps	acc	ps	
low-perf	LLaVA-1.5-7B [40]	47.7	4.9	44.4	22.8	20.4	2.3	28.5	0.9	49.2	1.4	70.5	1.3	71.2	0.7	47.4	4.9
	LLaVA-NeXT-Mistral-7B [41]	2.3	3.7	1.6	12.4	7.9	2.4	50.3	0.6	13.9	0.9	20.9	1.2	53.7	0.5	21.5	3.1
	LLaVA-NeXT-Vicuna-7B [41]	44.3	4.9	27.0	42.6	22.5	2.3	53.0	0.7	18.7	1.9	71.9	2.1	76.3	0.5	44.8	7.9
	LLaVA-OV-0.5B [32]	44.3	5.6	19.0	29.7	31.0	1.0	42.5	1.2	46.5	1.5	58.3	4.3	70.3	0.5	44.6	6.3
	Qwen2.5-VL-3B [5]	48.9	1.3	49.2	5.0	31.7	0.4	62.4	1.4	31.3	0.4	56.5	0.2	59.5	0.0	48.5	1.2
	SmolVLM2-2.2B [48]	0.0	0.1	27.0	0.1	30.0	0.5	57.2	1.1	27.4	0.7	72.5	0.9	60.1	0.0	39.2	0.5
	Idefics3-8B [30]	29.5	9.4	38.1	39.1	31.0	0.7	70.8	0.4	28.6	1.1	62.9	0.4	74.8	0.4	48.0	7.4
	InternVL3-1B [83]	62.5	2.9	41.3	1.0	37.2	1.6	49.4	1.2	34.6	1.8	69.3	1.1	72.9	0.3	52.5	1.4
	InternVL3-2B [83]	42.0	1.1	60.3	13.6	41.0	1.0	73.1	0.7	42.9	1.3	71.0	1.0	83.6	0.3	59.1	2.7
InstructBLIP [11]	1.1	0.5	15.9	4.2	10.7	0.1	13.4	0.1	22.1	0.1	46.6	0.2	24.0	0.0	19.1	0.7	
high-perf	LLaVA-OV-7B [32]	20.5	0.6	38.1	0.6	40.3	0.9	75.5	0.9	49.9	0.3	54.1	0.3	63.5	0.1	48.8	0.5
	Qwen2.5-VL-7B [5]	0.0	0.0	25.4	0.0	31.5	0.3	81.1	0.9	43.6	0.7	53.0	0.1	60.6	0.0	42.2	0.3
	InternVL3-8B [83]	2.3	0.0	38.1	0.1	33.9	0.5	76.4	0.4	42.3	0.8	57.7	0.1	69.3	0.1	45.7	0.3
	Phi-4-Multimodal [1]	9.1	0.4	25.4	1.2	30.7	0.1	67.8	1.2	49.9	0.9	60.9	0.2	69.4	0.1	44.7	0.6

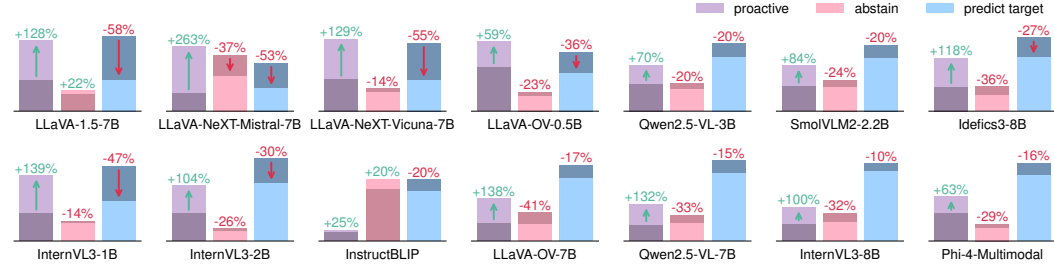


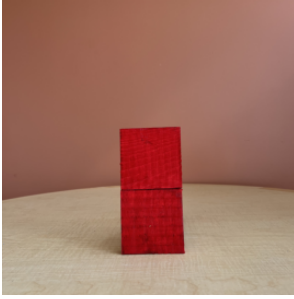
Figure 19: **Action distributions with hints.** Bars describe action distributions with (light) and without (dark) hints in the prompt for all evaluated models.

Table 7: **MLLMs results on ProactiveBench by conditioning on conversation histories.** We report the accuracy (*acc*) in percentages (%) and average number of proactive suggestions (*ps*) for all datasets, with global averages in the last column. We omit models not supporting multi-image inference.

	model	ROD		VSOD		MVP-N		IN-C		QD		CIT		COCO		avg.	
		acc	ps	acc	ps	acc	ps	acc	ps	acc	ps	acc	ps	acc	ps	acc	ps
low-perf	LLaVA-OV-0.5B [32]	1.1	6.0	17.5	8.8	26.5	0.4	41.9	0.9	31.0	1.3	53.9	3.1	60.7	0.2	33.2	3.0
	Qwen2.5-VL-3B [5]	0.0	0.0	27.0	0.0	25.5	0.0	54.9	1.3	28.3	0.3	58.4	0.5	56.5	0.0	35.8	0.3
	SmolVLM2-2.2B [48]	0.0	0.0	20.6	0.4	25.7	0.0	45.6	0.5	22.2	0.3	59.9	0.4	59.8	0.0	33.4	0.2
	Idefics3-8B [30]	13.6	1.3	27.0	5.6	26.7	0.1	66.3	0.5	26.4	0.5	60.3	0.4	62.2	0.1	40.4	1.2
	InternVL3-1B [83]	0.0	6.4	36.5	5.4	21.6	0.5	54.7	0.8	22.5	0.6	57.5	1.0	69.2	0.1	37.4	2.1
	InternVL3-2B [83]	0.0	0.1	47.6	4.2	27.6	0.2	56.3	1.0	30.0	0.7	63.2	1.7	77.2	0.1	43.1	1.1
high-perf	LLaVA-OV-7B [32]	0.0	0.0	31.7	5.0	23.7	0.1	62.1	0.5	45.5	0.5	57.5	0.7	60.0	0.0	40.1	1.0
	Qwen2.5-VL-7B [5]	0.0	0.0	14.3	0.0	24.8	0.0	71.4	0.7	30.9	0.1	58.5	0.1	59.5	0.0	37.1	0.1
	InternVL3-8B [83]	0.0	0.0	28.6	5.0	22.4	0.1	64.9	0.5	29.6	0.8	55.6	0.7	67.0	0.0	38.3	1.0
	Phi-4-Multimodal [1]	0.0	0.0	20.6	9.7	29.5	0.0	51.1	1.0	37.2	0.6	65.4	2.6	64.9	0.1	38.4	2.0

Table 8: **MLLMs results on ProactiveBench by conditioning on few-shots.** We report the accuracy (*acc*) in percentages (%) and average number of proactive suggestions (*ps*) for all datasets, with global averages in the last column. We omit models not supporting multi-image inference.

model		1 sample				3 samples			
		ROD		MVP-N		ROD		MVP-N	
		acc	ps	acc	ps	acc	ps	acc	ps
low-perf	LLaVA-OV-0.5B [32]	21.6	7.4	24.2	0.6	21.6	6.6	23.5	0.6
	Qwen2.5-VL-3B [5]	2.3	0.2	15.1	0.1	2.3	0.0	13.7	0.3
	SmolVLM2-2.2B [48]	0.0	0.1	23.9	0.2	0.0	0.1	0.5	0.0
	Idefics3-8B [30]	31.8	1.3	26.3	0.1	34.1	1.2	24.4	0.4
	InternVL3-1B [83]	0.0	9.8	14.7	0.7	0.0	9.5	15.4	0.8
	InternVL3-2B [83]	0.0	0.7	18.8	0.5	0.0	0.6	18.3	0.5
high-perf	LLaVA-OV-7B [32]	9.1	0.2	29.6	0.7	19.3	0.4	29.4	0.9
	Qwen2.5-VL-7B [5]	0.0	0.0	17.4	0.1	4.5	0.1	28.4	0.6
	InternVL3-8B [83]	0.0	0.0	11.1	0.1	0.0	0.0	12.1	0.1
	Phi-4-Multimodal [1]	2.3	0.2	26.2	0.1	38.6	3.7	16.6	1.5



What missing details are obscured by the red blocks?
Choose from the following options. Options:
A. Move the red blocks to the left.
B. Move the red blocks to the right.
C. I cannot tell.
D. Screwdriver.
E. Banana.
F. Vase.
G. Spatula.
Please only return one of the options without any other words.

Figure 20: **ROD input example.** In the first step, the ROD environment returns images of completely occluded target objects.



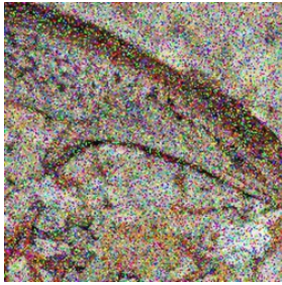
This is a frame extracted from a video. Answer the following question.
Can you say who is speaking?
Choose from the following options. Options:
A. I cannot answer this question.
B. Rewind the video.
C. Wait for the occlusion to disappear.
D. Monika schnitzer.
E. Ursula von der leyen.
F. Ge you.
G. José mourinho.
Please only return one of the options without any other words.

Figure 21: **VSOD input example.** In the first step, the VSOD environment returns video frames of occluded subjects.



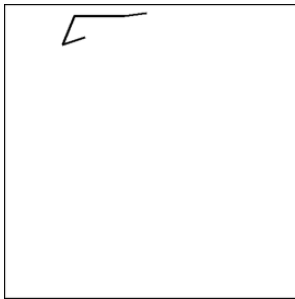
Could you name the object in this image?
Choose from the following options. Options:
A. Change the camera angle of the object.
B. I do not know what is this object.
C. Selex whey protein drink peach.
D. Selex sports whey protein powder peach.
E. Selex sports whey protein powder chocolate.
F. Selex whey protein drink chocolate.
Please only return one of the options without any other words.

Figure 22: **MVP-N input example.** In the first step, the MVP-N environment returns uninformative object views.



Provide the classification of the object in the image.
Choose from the following options. Options:
A. Denoise the image.
B. I do not know what is this object.
C. Increase image resolution.
D. Reduce brightness.
E. Deblur the image.
F. Perfume.
G. Great_pyrenees.
H. Alligator_lizard.
I. Cello.
Please only return one of the options without any other words.

Figure 23: **ImageNet-C input example.** In the first step, the IN-C environment returns heavily corrupted images.



Describe the object in the quickdraw in terms of its category.
Choose from the following options. Options:
A. I cannot answer this question.
B. Make this drawing more complete.
C. The eiffel tower.
D. Potato.
E. Bed.
F. Tooth.
Please only return one of the options without any other words.

Figure 24: **QuickDraw input example.** In the first step, the QD environment returns the first stroke of a sketch.



This is a frame extracted from a video. Answer the following question.
Describe the object in the video in terms of its category.
Choose from the following options. Options:
A. I cannot answer this question.
B. Wait for the object to appear.
C. Rewind the video.
D. Eggs.
E. Butter.
F. Apple.
G. Avocado.
Please only return one of the options without any other words.

Figure 25: **ChangeIt input example.** In the first step, the CIT environment returns video frames where the target object or action will appear in the future.



Identify the object in this image.
Choose from the following options. Options:
A. Move the camera to the left.
B. I cannot answer this question.
C. Move the camera down.
D. Bowl.
E. Sink.
F. Cup.
G. Toilet.
Please only return one of the options without any other words.

Figure 26: **MS-COCO input example.** In the first step, the COCO environment returns images where object details are removed.

References

- [1] Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, et al. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras. *arXiv preprint arXiv:2503.01743*, 2025.
- [2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022.
- [3] John Aloimonos, Isaac Weiss, and Amit Bandyopadhyay. Active vision. *IJCV*, 1988.
- [4] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *ICCV*, 2015.
- [5] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.
- [6] Björn Browatzki, Vadim Tikhonoff, Giorgio Metta, Heinrich H Bülthoff, and Christian Wallraven. Active object recognition on a humanoid robot. In *ICRA*, 2012.
- [7] Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022.
- [8] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, 2023. URL <https://lmsys.org/blog/2023-03-30-vicuna/>.
- [9] Tai-Yin Chiu, Yinan Zhao, and Danna Gurari. Assessing image quality issues for real-world problems. In *CVPR*, 2020.
- [10] Ian Chuang, Andrew Lee, Dechen Gao, M Naddaf-Sh, Iman Soltani, et al. Active vision might be all you need: Exploring active vision in bimanual robotic manipulation. *arXiv preprint arXiv:2409.17435*, 2024.
- [11] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. In *NeurIPS*, 2023.
- [12] Song Dingjie, Shunian Chen, Guiming Hardy Chen, Fei Yu, Xiang Wan, and Benyou Wang. Milebench: Benchmarking mllms in long context. In *COLM*, 2024.
- [13] Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmvalkit: An open-source toolkit for evaluating large multi-modality models. In *ACMMM*, 2024.
- [14] Jinhao Duan, Renming Zhang, James Diffenderfer, Bhavya Kailkhura, Lichao Sun, Elias Stengel-Eskin, Mohit Bansal, Tianlong Chen, and Kaidi Xu. Gtbench: Uncovering the strategic reasoning limitations of llms via game-theoretic evaluations. In *NeurIPS*, 2024.
- [15] Cornelia Fermuller and Yiannis Aloimonos. Representations for active vision. In *IJCAI*, 1995.
- [16] Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see but not perceive. In *ECCV*, 2024.
- [17] Manu Gaur, Makarand Tapaswi, et al. Detect, describe, discriminate: Moving beyond vqa for mllm evaluation. *arXiv preprint arXiv:2409.15125*, 2024.
- [18] Melvyn A Goodale and A David Milner. Separate visual pathways for perception and action. *Trends in neurosciences*, 1992.
- [19] Google. Quick draw!, 2016. URL <https://quickdraw.withgoogle.com/data>
- [20] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *CVPR*, 2017.
- [21] Yangyang Guo, Fangkai Jiao, Zhiqi Shen, Liqiang Nie, and Mohan Kankanhalli. Unk-vqa: A dataset and a probe into the abstention ability of multi-modal large models. *T-PAMI*, 2024.
- [22] Amanda J Haskins, Jeff Mentch, Thomas L Botch, and Caroline E Robertson. Active vision in immersive, 360 real-world environments. *Scientific Reports*, 2020.
- [23] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *ICLR*, 2019.
- [24] Anna Heuer, Sven Ohl, and Martin Rolf. Memory for action: A functional view of selection in visual working memory. *Visual Cognition*, 2020.

- [25] Dongfu Jiang, Xuan He, Huaye Zeng, Cong Wei, Max Ku, Qian Liu, and Wenhua Chen. Mantis: Interleaved multi-image instruction tuning. In *TMLR*, 2024.
- [26] Fengqing Jiang. Identifying and mitigating vulnerabilities in llm-integrated applications. Master’s thesis, University of Washington, 2024.
- [27] Mehran Kazemi, Hamidreza Alvani, Ankit Anand, Jialin Wu, Xi Chen, and Radu Soricut. Geomverse: A systematic evaluation of large models for geometric reasoning. *arXiv preprint arXiv:2312.12241*, 2023.
- [28] Mehran Kazemi, Nishanth Dikkala, Ankit Anand, Petar Devic, Ishita Dasgupta, Fangyu Liu, Bahare Fatemi, Pranjal Awasthi, Sreenivas Gollapudi, Dee Guo, et al. Remi: A dataset for reasoning with multiple images. In *NeurIPS*, 2024.
- [29] Jihyung Kil, Zheda Mai, Justin Lee, Zihe Wang, Kerrie Cheng, Lemeng Wang, Ye Liu, Arpita Chowdhury, and Wei-Lun Chao. Compbench: A comparative reasoning benchmark for multimodal llms. In *NeurIPS*, 2024.
- [30] Hugo Laurençon, Andrés Marafioti, Victor Sanh, and Léo Tronchon. Building and better understanding vision-language models: insights and future directions. In *NeurIPS*, 2024.
- [31] Ariel N Lee, Sarah Adel Bargal, Janavi Kasera, Stan Sclaroff, Kate Saenko, and Nataniel Ruiz. Hardwiring vit patch selectivity into cnns using patch mixing. *arXiv preprint arXiv:2306.17848*, 2023.
- [32] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. In *TMLR*, 2025.
- [33] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022.
- [34] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023.
- [35] Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. In *CVPR*, 2024.
- [36] Manling Li, Shiyu Zhao, Qineng Wang, Kangrui Wang, Yu Zhou, Sanjana Srivastava, Cem Gokmen, Tony Lee, Erran Li Li, Ruohan Zhang, et al. Embodied agent interface: Benchmarking llms for embodied decision making. *NeurIPS*, 2024.
- [37] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *EMNLP*, 2023.
- [38] Junhua Liao, Haihan Duan, Xin Li, Haoran Xu, Yanbing Yang, Wei Cai, Yanru Chen, and Liangyin Chen. Occlusion detection for automatic video editing. In *ACMMM*, 2020.
- [39] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.
- [40] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023.
- [41] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *CVPR*, 2024.
- [42] Li Liu, Diji Yang, Sijia Zhong, Kalyana Suma Sree Tholeti, Lei Ding, Yi Zhang, and Leilani Gilpin. Right this way: Can vlms guide us to see more to answer questions? In *NeurIPS*, 2024.
- [43] Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, et al. Agentbench: Evaluating llms as agents. In *ICLR*, 2023.
- [44] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. Mmbench: Is your multi-modal model an all-around player? In *ECCV*, 2024.
- [45] Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of ocr in large multimodal models. *Science China Information Sciences*, 2024.
- [46] Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, Xinyuan Lu, Ziyu Liu, Yan Ma, Xiaoyi Dong, et al. Mmlongbench-doc: Benchmarking long-context document understanding with visualizations. In *NeurIPS*, 2024.
- [47] Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. Video-chatgpt: Towards detailed video understanding via large vision and language models. *arXiv preprint arXiv:2306.05424*, 2023.
- [48] Andrés Marafioti, Orr Zohar, Miquel Farré, Merve Noyan, Elie Bakouch, Pedro Cuenca, Cyril Zakka, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, et al. Smolvlm: Redefining small and efficient multimodal models. *arXiv preprint arXiv:2504.05299*, 2025.

- [49] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *CVPR*, 2019.
- [50] David Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Henry Holt and Co., Inc., 1982.
- [51] Fanqing Meng, Jin Wang, Chuanhao Li, Quanfeng Lu, Hao Tian, Jiaqi Liao, Xizhou Zhu, Jifeng Dai, Yu Qiao, Ping Luo, et al. Mmiu: Multimodal multi-image understanding for evaluating large vision-language models. In *ICLR*, 2024.
- [52] Aidar Myrzakhan, Sondos Mahmoud Bsharat, and Zhiqiang Shen. Open-llm-leaderboard: From multi-choice to open-style questions for llms evaluation, benchmark, and arena. *arXiv preprint arXiv:2406.07545*, 2024.
- [53] Arsha Nagrani, Mingda Zhang, Ramin Mehran, Rachel Hornung, Nitesh Bharadwaj Gundavarapu, Nilpa Jha, Austin Myers, Xingyi Zhou, Boqing Gong, Cordelia Schmid, et al. Neptune: The long orbit to benchmarking long video understanding. *arXiv preprint arXiv:2412.09582*, 2024.
- [54] OpenAI. o4-mini, 2025. URL <https://platform.openai.com/docs/models/o4-mini>
- [55] Aishwarya Padmakumar, Jesse Thomason, Ayush Shrivastava, Patrick Lange, Anjali Narayan-Chen, Spandana Gella, Robinson Piramuthu, Gokhan Tur, and Dilek Hakkani-Tur. Teach: Task-driven embodied agents that chat. In *AAAI*, 2022.
- [56] A Paszke. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019.
- [57] Pouya Pezeshkpour and Estevam Hruschka. Large language models sensitivity to the order of options in multiple-choice questions. In *ACL*, 2024.
- [58] Chiara Plizzari, Alessio Tonioni, Yongqin Xian, Achin Kulshrestha, and Federico Tombari. Omnia de egotempo: Benchmarking temporal understanding of multi-modal llms in egocentric videos. In *CVPR*, 2025.
- [59] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015.
- [60] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. Habitat: A platform for embodied ai research. In *ICCV*, 2019.
- [61] Hao Shao, Yuxuan Hu, Letian Wang, Guanglu Song, Steven L Waslander, Yu Liu, and Hongsheng Li. Lmdrive: Closed-loop end-to-end driving with large language models. In *CVPR*, 2024.
- [62] Larry Shapiro. The embodied cognition research programme. *Philosophy compass*, 2007.
- [63] Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *CVPR*, 2020.
- [64] Mustafa Shukor, Alexandre Rame, Corentin Dancette, and Matthieu Cord. Beyond task performance: Evaluating and reducing the flaws of large multimodal models with in-context learning. In *ICLR*, 2024.
- [65] Edward Smith, David Meger, Luis Pineda, Roberto Calandra, Jitendra Malik, Adriana Romero Soriano, and Michal Drozdal. Active 3d shape reconstruction from vision and touch. In *NeurIPS*, 2021.
- [66] Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, et al. Moviechat: From dense token to sparse memory for long video understanding. In *CVPR*, 2024.
- [67] Tomáš Souček, Jean-Baptiste Alayrac, Antoine Miech, Ivan Laptev, and Josef Sivic. Look for the change: Learning object states and state-modifying actions from untrimmed web videos. In *CVPR*, 2022.
- [68] Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *CVPR*, 2024.
- [69] Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. In *NeurIPS*, 2021.
- [70] Fei Wang, Xingyu Fu, James Y Huang, Zekun Li, Qin Liu, Xiaogeng Liu, Mingyu Derek Ma, Nan Xu, Wenxuan Zhou, Kai Zhang, et al. Muirbench: A comprehensive benchmark for robust multi-image understanding. In *NeurIPS*, 2024.
- [71] Ren Wang, Jiayue Wang, Tae Sung Kim, Jinsung Kim, and Hyuk-Jae Lee. Mvp-n: A dataset and benchmark for real-world multi-view object classification. In *NeurIPS*, 2022.
- [72] Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. Scienceworld: Is your agent smarter than a 5th grader? In *EMNLP*, 2022.

- 784 [73] Spencer Whitehead, Suzanne Petryk, Vedaad Shakib, Joseph Gonzalez, Trevor Darrell, Anna Rohrbach,
785 and Marcus Rohrbach. Reliable visual question answering: Abstain rather than answer incorrectly. In
786 *ECCV*, 2022.
- 787 [74] Robert Wolfe, Isaac Slaughter, Bin Han, Bingbing Wen, Yiwei Yang, Lucas Rosenblatt, Bernease Herman,
788 Eva Brown, Zening Qu, Nic Weber, et al. Laboratory-scale ai: Open-weight models are competitive with
789 chatgpt even in low-resource settings. In *ACM-FAccT*, 2024.
- 790 [75] Tsung-Han Wu, Giscard Biamby, David Chan, Lisa Dunlap, Ritwik Gupta, Xudong Wang, Joseph E
791 Gonzalez, and Trevor Darrell. See say and segment: Teaching llms to overcome false premises. In *CVPR*,
792 2024.
- 793 [76] Manjie Xu, Guangyuan Jiang, Wei Liang, Chi Zhang, and Yixin Zhu. Active reasoning in an open-world
794 environment. In *NeurIPS*, 2023.
- 795 [77] Zhenhua Xu, Yujia Zhang, Enze Xie, Zhen Zhao, Yong Guo, Kwan-Yee K Wong, Zhenguo Li, and
796 Hengshuang Zhao. Drivegpt4: Interpretable end-to-end autonomous driving via large language model.
797 *IEEE RA-L*, 2024.
- 798 [78] Mengge Xue, Zhenyu Hu, Liqun Liu, Kuo Liao, Shuang Li, Honglin Han, Meng Zhao, and Chengguo Yin.
799 Strengthened symbol binding makes large language models reliable multiple-choice selectors. In *ACL*,
800 2024.
- 801 [79] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu
802 Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding
803 and reasoning benchmark for expert agi. In *CVPR*, 2024.
- 804 [80] Rui Zeng, Yuhui Wen, Wang Zhao, and Yong-Jin Liu. View planning in robot active vision: A survey of
805 systems, algorithms, and applications. *CVM*, 2020.
- 806 [81] Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. Large language models are not
807 robust multiple choice selectors. In *ICLR*, 2024.
- 808 [82] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin,
809 Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. In
810 *NeurIPS*, 2023.
- 811 [83] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, Hao Tian,
812 Weijie Su, Jie Shao, et al. Internv13: Exploring advanced training and test-time recipes for open-source
813 multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.