

Contents of Supplementary Material

Additional experimental results	1
Details of annotation and pair construction	4
Additional details for prompts, response generation, and MLLM-as-a-judge	8
Limitations and Future Directions	20
Examples	21

7. Additional Experimental Results

7.1. Multimodal reasoning performance breakdown by pair modality and pair type.

Table 5 shows the breakdown of the results by pair modality and pair type.

Model	Same modality: Image+text		Same modality: Text		Mixed modality: Pref: Image+text; Not Pref: Text		Mixed modality: Pref: Text; Not Pref: Image+text	
	Correct vs. incorrect reason	Correct vs. incorrect answer	Correct vs. incorrect reason	Correct vs. incorrect answer	Correct vs. incorrect reason	Correct vs. incorrect answer	Correct vs. incorrect reason	Correct vs. incorrect answer
Open-source models								
Gemma3 4B	47.39	50.63	50.00	48.32	63.64	57.00	38.68	36.00
Gemma3 12B	49.57	51.47	54.02	52.10	81.82	73.50	15.09	11.50
Gemma3 27B	51.30	50.21	51.79	51.68	87.50	79.50	10.38	10.50
Qwen2.5-VL-7B	49.12	48.10	51.34	50.00	52.27	39.00	47.17	40.31
Qwen2.5-VL-72B	52.63	48.10	53.57	54.41	78.41	68.00	16.04	23.98
Qwen3-VL-8B	57.46	52.53	58.48	54.20	71.59	73.00	34.91	36.73
Qwen3-VL-32B	62.28	54.43	60.71	56.93	78.41	80.00	25.47	33.16
Qwen3-VL-30BA3B	58.77	55.72	57.59	56.30	75.00	78.00	36.79	43.37
Qwen3-VL-235BA22B	58.11	57.02	55.80	57.14	85.23	81.96	23.58	26.02
API-based models								
GPT-4o	50.43	50.42	55.80	56.51	81.82	80.00	18.87	18.00
GPT-4.1	56.09	50.42	58.04	58.61	93.18	81.50	10.38	13.50
GPT-5	69.57	67.02	75.89	80.25	88.64	88.00	36.79	40.00
Gemini 2.5 Flash	60.53	58.47	56.25	59.03	86.36	76.00	16.98	38.42
Gemini 2.5 Pro	73.91	66.18	62.95	65.55	84.09	79.00	43.40	58.00
Gemini 3 Pro	71.88	84.75	75.45	82.49	84.88	87.00	66.98	72.00

Table 5. Multimodal reasoning performance breakdown by pair modality and pair type.

7.2. Performance by task source and pair type

The pairwise evaluation results presented in Tables 6, 7, 8, 9 distinct performance patterns across multimodal tasks and model capabilities.

Image Generation: Performance varies moderately across benchmarks, with realunify (48-78.5%) and oneigbench (73-74% for top models) showing higher judge agreement rates, while wise consistently yields the lowest scores across all models (47-66%).

Image Editing: The breakdown reveals stark differences in benchmark difficulty, with text-based editing benchmarks (text: 54-83%, risebench: 48-83%) showing significantly higher agreement rates compared to general editing tasks (emu-edit: 49-72%, multi-image editing: 51-71%). This pattern holds consistently across all judge models, indicating that text rendering or text-focused editing provides clearer discriminative signals for pairwise evaluation than open-ended creative edits.

Interleaved: Performance is relatively uniform across benchmarks for top models, with isgbench consistently scoring highest (76-79% for frontier models) and all benchmarks clustering within a 5-8 percentage point range for leading judges.

Reasoning: This task exhibits the most dramatic benchmark-level variance, with muirbench showing substantially higher scores (36-76%) compared to other benchmarks, while vstar proves exceptionally challenging (30-52%). The performance on blink and mindcube clusters tightly (44-55% for most models), suggesting these represent a baseline reasoning difficulty.

7.3. Win rate analysis on generations

We also report the generation capabilities of MLLMs as content producers (models generating the multimodal content being evaluated, reported in Table 10). Judging requires discriminative understanding and alignment with human preferences, while generation requires creative synthesis and technical execution. A model may excel at one role while underperforming at the other, as we observe in our results.

Judge Model	Overall %	evalmuse (n=390)	oneighbench (n=278)	r2ibench (n=128)	realunify (n=93)	wise (n=111)
Gemini 3 Pro	74.4%	74.5%	74.5%	72.7%	80.1%	71.2%
Gemini 2.5 Pro	70.5%	69.6%	73.0%	70.7%	77.4%	61.7%
GPT5	70.5%	67.3%	73.2%	72.7%	78.5%	66.2%
GPT 4.1	65.8%	62.4%	69.6%	66.8%	75.8%	58.6%
Qwen-3 vl32b	64.1%	63.3%	68.3%	60.9%	65.6%	59.0%
Gemini25flash	63.1%	60.3%	64.4%	62.1%	75.3%	60.8%
Qwen-3 vl235ba22b	62.0%	59.2%	65.5%	60.2%	68.8%	59.0%
GPT 4o	60.3%	58.1%	65.5%	59.4%	65.1%	52.3%
Qwen-3 vl30ba3b	60.0%	57.6%	64.2%	57.8%	59.1%	61.3%
Qwen-3 vl8b	59.4%	59.2%	62.8%	57.4%	62.4%	50.9%
Qwen25vl72b	59.1%	56.8%	62.1%	56.2%	67.7%	55.9%
Gemma 3-27b	58.3%	56.7%	59.5%	60.2%	66.7%	51.8%
Llama4-17b	56.7%	56.3%	58.1%	53.4%	58.0%	58.1%
Gemma 3-12b	56.0%	54.6%	58.6%	53.1%	62.9%	52.3%
Gemma 3-4b	51.7%	50.1%	53.2%	50.8%	54.3%	52.7%
Qwen25vl7b	50.4%	48.8%	55.9%	47.7%	48.4%	46.8%

Table 6. Image Generation: Pairwise model evaluation breakdown by benchmark.

Judge Model	Overall %	dreambench (n=242)	emu-edit (n=329)	hq-edit (n=53)	multi-image editing (n=178)	risebench (n=84)	text (n=114)
Gemini 3 Pro	74.9%	70.0%	75.8%	79.2%	68.5%	82.1%	84.6%
GPT5	73.8%	71.3%	71.6%	72.6%	71.1%	82.7%	83.3%
Gemini 2.5 Pro	71.3%	70.2%	68.4%	76.0%	64.9%	79.8%	83.3%
GPT 4.1	68.2%	67.6%	63.4%	68.9%	66.0%	74.4%	82.0%
Qwen-3 vl32b	67.3%	63.6%	65.8%	72.6%	64.9%	71.4%	77.6%
Gemini25flash	66.5%	63.8%	67.6%	67.9%	59.6%	72.6%	74.6%
Qwen-3 vl235ba22b	66.0%	66.7%	64.1%	67.0%	57.9%	72.0%	77.6%
GPT 4o	65.0%	66.5%	60.6%	60.4%	64.6%	66.1%	76.3%
Qwen25vl72b	64.6%	63.6%	63.5%	61.3%	60.1%	63.7%	79.4%
Qwen-3 vl8b	61.7%	59.3%	61.9%	61.3%	54.8%	62.5%	76.3%
Llama4-17b	61.1%	60.2%	59.8%	65.3%	57.3%	64.4%	67.9%
Gemma 3-27b	60.2%	61.2%	60.2%	58.5%	57.6%	53.0%	68.4%
Qwen-3 vl30ba3b	59.5%	58.1%	58.5%	59.4%	56.5%	61.9%	68.4%
Gemma 3-12b	58.0%	58.9%	55.0%	56.6%	57.6%	60.7%	64.0%
Qwen25vl7b	57.1%	59.5%	53.8%	57.5%	53.7%	52.4%	70.2%
Gemma 3-4b	51.0%	53.3%	49.1%	49.1%	51.4%	47.6%	54.4%

Table 7. Image Editing: Pairwise model evaluation breakdown by benchmark.

Judge Model	Overall %	chameleon (n=284)	interleavedeval (n=267)	isgbench (n=421)	mmmjg (n=28)
Gemini 3 Pro	76.4%	76.4%	76.8%	76.1%	76.8%
Gemini 2.5 Pro	75.1%	73.4%	71.5%	78.5%	75.0%
GPT5	74.4%	72.9%	72.8%	76.5%	71.4%
Qwen-3 vl32b	70.5%	66.9%	70.4%	73.3%	66.1%
Gemini25flash	69.4%	64.4%	70.8%	71.5%	75.0%
GPT 4.1	67.0%	65.3%	66.3%	69.1%	60.7%
Qwen-3 vl235ba22b	66.7%	63.5%	66.3%	68.9%	69.6%
Qwen25vl72b	62.3%	59.9%	61.4%	64.7%	58.9%
GPT 4o	61.5%	60.9%	60.1%	63.3%	53.6%
Qwen-3 vl8b	61.5%	59.3%	63.7%	61.3%	66.1%
Gemma 3-27b	61.1%	59.9%	59.4%	62.4%	69.6%
Gemma 3-12b	58.0%	57.6%	58.1%	58.2%	58.9%
Qwen-3 vl30ba3b	57.3%	55.8%	56.8%	57.7%	69.6%
Llama4-scout-17b	54.4%	55.7%	54.9%	52.5%	66.7%
Gemma 3-4b	51.3%	50.4%	52.8%	51.1%	50.0%
Qwen25vl7b	48.4%	48.4%	46.4%	49.8%	48.2%

Table 8. Interleaved: Pairwise model evaluation breakdown by benchmark.

Judge Model	Overall %	blink (n=355)	mindcube (n=367)	muirbench (n=137)	realunify (n=55)	visulogic (n=49)	vstar (n=37)
Qwen-3 vl32b	56.6%	52.4%	55.6%	71.5%	56.4%	61.2%	45.9%
Qwen-3 vl30ba3b	56.5%	54.6%	52.9%	70.3%	56.9%	62.1%	51.1%
Qwen-3 vl235ba22b	55.9%	52.6%	54.2%	76.1%	52.8%	59.2%	30.6%
Qwen-3 vl8b	53.7%	51.9%	50.9%	64.2%	54.4%	60.4%	48.5%
Qwen25vl72b	50.2%	46.7%	52.0%	57.7%	50.9%	54.1%	29.7%
Llama4-scout-17b	44.5%	43.5%	47.7%	35.8%	49.1%	44.9%	48.6%

Table 9. Reasoning: Pairwise model evaluation breakdown by benchmark.

Rank	Task	Model	Win Rate (%)
1	Image Gen.	GPT-Image-1	60.4
2		Imagen 4	57.4
3		Imagen 4 Ultra	56.5
4		Gemini 2.5 Flash	54.3
5		Imagen 3	49.2
6		Gemini 2.0 Flash	45.6
7		SD 3.5 Large	41.0
8		FLUX	36.8
1	Image Editing	Gemini 2.5 Flash	59.2
2		GPT-Image-1	53.2
3		FLUX-Kontext	49.0
4		Gemini 2.0 Flash	47.1
5		Imagen Edit	35.2
1	Interleaved	GPT-Gemini Agent	57.1
2		GPT-Image Agent	56.9
3		Gemini 2.5 Flash	53.2
4		Gemini 2.0 Flash	46.2
5		GPT-Imagen Agent	42.1
6		GPT-FLUX Agent	40.4

Table 10. Model win rates (%) on Multimodal RewardBench 2 ranked by performance within each task. Win rate is computed as $(wins + 0.5 \times ties) / \text{total comparisons}$.

Table 10 presents the win rates of generative models across MMRB2’s three generation tasks (Tasks 1–3), where win rate is computed as $(wins + 0.5 \times ties) / \text{total comparisons}$ based on human majority preferences. These are the same model outputs that judges evaluate in Table 2, allowing us to assess both generation quality and judgment accuracy within a unified framework.

Image Generation. GPT-Image-1 (60.4%) narrowly leads text-to-image generation, closely followed by Imagen 4 (57.4%), Imagen 4 Ultra (56.5%), and Gemini 2.5 Flash (54.3%), indicating a highly competitive landscape among top proprietary models with less than 6 points separating the leaders. Open-source models lag substantially: Stable Diffusion 3.5 Large (41.0%) and FLUX (36.8%) trail by 19–24 points.


Image Editing. Interestingly, general-purpose multimodal models such as Gemini 2.5 Flash (59.2%) and GPT-Image-1 (53.2%) outperform specialized models. While Imagen Edit achieves only a 35.2% win rate despite being purpose-built for editing, the gap is less severe than earlier reports suggested. FLUX-Kontext (49.0%) demonstrates competitive performance for an open-source solution, though it still trails the leaders. These results suggest that strong vision–language understanding provides significant advantages for instruction-based editing, even if specialized architectures are not entirely obsolete.

Interleaved Generation. Agent-based systems dominate, with GPT-Gemini Agent (57.1%) and GPT-Image Agent (56.9%) leading by narrow margins. Native multimodal models like Gemini 2.5 Flash (53.2%) perform competitively, narrowing the gap with agent architectures. GPT-FLUX Agent’s improved but still modest performance (40.4%) confirms that agent quality depends critically on component model quality, though the improvement suggests that better integration strategies can help.


Prompt:

a photo of a cow and a horse

Response A



Response B



Response A: Overall quality of generated image

Does the image contain significant technical errors that break composition (including style coherence and realism) or make it visually unappealing? For example, issues with impossible geometry, strange objects, garbled text, incorrect human anatomy (limbs bending the wrong way, wrong number of fingers) or unappealing aesthetics (distorted faces, large asymmetry in bodies)?

Severe flaws | Very unappealing
 Some flaws | Somewhat unappealing
 Minor flaws | Somewhat appealing
 No noticeable technical or logical flaws | Very appealing
 N/A

Response B: Overall quality of generated image

Does the image contain significant technical errors that break composition (including style coherence and realism) or make it visually unappealing? For example, issues with impossible geometry, strange objects, garbled text, incorrect human anatomy (limbs bending the wrong way, wrong number of fingers) or unappealing aesthetics (distorted faces, large asymmetry in bodies)?

Severe flaws | Very unappealing
 Some flaws | Somewhat unappealing
 Minor flaws | Somewhat appealing
 No noticeable technical or logical flaws | Very appealing
 N/A

Response A: Correctness of text rendering

Does the image render text correctly? For example, issues with misspellings, distorted text, and inconsistent capitalization?

Major deviations | Many obvious errors
 Partial match | some errors
 Mostly match | minor errors
 Full match | No noticeable error
 N/A

Response B: Correctness of text rendering

Does the image render text correctly? For example, issues with misspellings, distorted text, and inconsistent capitalization?

Major deviations | Many obvious errors
 Partial match | some errors
 Mostly match | minor errors
 Full match | No noticeable error
 N/A

Response A: Faithfulness to the input text

How accurately and completely does the output follow the explicit and implicit instructions in the prompt?

Major deviations
 Some mismatch
 Minor mismatch
 Full match
 N/A

Response B: Faithfulness to the input text

How accurately and completely does the output follow the explicit and implicit instructions in the prompt?

Major deviations
 Some mismatch
 Minor mismatch
 Full match
 N/A

Overall Preference

Which response looks better to you overall?

Response A is significantly better
 Response A is marginally better
 Unsure or Response A is negligibly better
 Unsure or Response B is negligibly better
 Response B is marginally better
 Response B is significantly better

Overall Preference Rationale

Please provide a rationale justifying your preference (1-2+ sentences).

Figure 6. Annotation interface for the MMRB2 text-to-image task. Note that for image editing and interleaved tasks, there are more fine-grained questions.

8. Details for Annotation and Pair Construction

8.1. Tasks 1-3

Figure 6 shows a sample of the annotation interface for the MMRB2 text-to-image task. In this section we provide additional details on the human annotation procedure.

For each annotation task, we provide a prompt and two responses, A and B, and the goal is to assess the quality of each response and then rate them. Annotators answer the following questions:

- **Prompt Quality Check:**

Indicate whether the prompt is correct (Yes/No).

- **Pointwise Evaluation for Response A and Response B:**

For each response, rate the following dimensions on a 4-point scale (see Section 8.1.1 for details):

- Faithfulness to the text instruction
- (Tasks 2 and 3) Faithfulness to the input image
- Overall quality of the generated image
- (Task 3 only) Cross-generation image congruence
- (Task 3 only) Generated text-image alignment

- (Task 3 only) Technical quality of generated text
- (Conditional) Correctness of text rendering
- **Rationales:**
Provide a brief rationale for the overall quality rating of both Response A and Response B.
- **Overall Preference:**
Indicate your overall preference between Response A and Response B, choosing one of the following:
 - A is significantly better
 - A is marginally better
 - Unsure or A is negligibly better
 - Unsure or B is negligibly better
 - B is marginally better
 - B is significantly better
- **Rationale for Preference:**
Provide a brief explanation for your overall preference.

8.1.1. Details of each question

1. **(For all tasks) Faithfulness to the text instruction:** How accurately and completely does the output follow the explicit and implicit **text** instructions in the prompt?

Rating	Label	Description
0	Major deviations	Key elements are missing, altered, or contradicted
1	Some mismatch	Some key elements are missing or altered.
2	Minor mismatch	Most key elements are present, but others are missing, incorrect, or incomplete
3	Full match	All key elements are represented exactly as described, with no significant omissions or contradictions

2. **(For task 2 and 3) Faithfulness to the input image:** When using an input image as context (e.g., editing, continuation, transformation), how well does the output incorporate the relevant elements of the input according to the instructions?

Rating	Label	Description
0	Fails to use the input meaningfully	Key elements are ignored, misinterpreted, or contradicted
1	Partial mismatch to the input	Some elements are carried over or transformed correctly, but those are not key elements or important aspects
2	Minor mismatch to the input	Most relevant elements are carried over or transformed correctly, but a few aspects are missing or incorrectly handled
3	Uses input fully	All relevant elements from the input are accurately incorporated, extended, or transformed exactly as instructed

3. **(For all tasks) Overall quality of generated image:** Does the image contain significant technical errors that break composition (including style coherence and **realism**) or make it visually unappealing? For example, issues with impossible geometry, strange objects, garbled text, incorrect human anatomy (limbs bending the wrong way, wrong number of fingers) or unappealing aesthetics (distorted faces, large asymmetry in bodies)?

Rating	Label	Description
0	Severe flaws , Very unappealing	Obvious errors that strongly affect usability: Major physical or visual errors that most viewers would notice immediately, unbalanced composition, clashing colors, heavy jarringness
1	Some flaws, Somewhat unappealing	Some errors that noticeably disrupt the image and jeopardize its usability regardless: Clear physical or visual errors that most viewers would eventually notice, the image isn't an eye sore but something is wrong with its overall composition or color balance
2	Minor flaws, Somewhat appealing	Small inaccuracies that are noticeable but are not strongly disruptive: Mostly plausible, but minor inconsistencies reduce believability, acceptable composition and color balance, but lacks artistic quality
3	No noticeable technical or logical flaws Very appealing	The image is free of noticeable technical errors: Fully coherent and physically plausible (if photorealistic, could be mistaken for a real photo; if stylized, maintains internal logic), strong composition, harmonious colors, and captivating style

4. **(For task 3) Cross-generation image congruence:** How well do the images relate to each other in a coherent way, maintaining consistency in recurring elements, style, and context, while allowing for appropriate variation when required?

Rating	Label	Description
0	Very incoherent	Many recurring elements change in unrealistic or unexplained ways, significantly breaking visual or thematic coherence
1	Rather incoherent	Some recurring elements change in unrealistic or unexplained ways, breaking visual or thematic coherence
2	Mostly coherent	Most recurring elements match, but there are noticeable mismatches or shifts that reduce cohesion
3	Full coherence	Recurring elements, style, and context remain consistent where appropriate, and variations are clearly intentional and coherent

5. **(For task 3) Generated Text-image alignment:** How well does the generated text align with the visual content of the image(s), without contradictions or unsupported details?

Rating	Label	Description
0	Very inconsistent	Text contradicts or misrepresents key elements of the image(s)
1	Rather inconsistent	Text aligns with some image content, but contains major mismatches or omissions
2	Mostly consistent	Text aligns with most image content, but contains minor mismatches or omissions
3	Full consistency	Text accurately and completely reflects the relevant details of the image(s) with no contradictions

6. **(For task 3) Technical quality of generated text:** Does the text contain serious issues such as hallucinations, omissions, or logical errors that undermine accuracy or coherence? Is the tone of the generated text appropriate and congruent with the overall context, style, and intent of the generation task?

Rating	Label	Description
0	Severe flaws (including tone)	Contains clear hallucinations, major omissions, or serious logical inconsistencies; tone is clearly mismatched to the intended context or style, or contradicts the task’s purpose
1	Some flaws (including tone)	Some factual gaps, unsupported claims, or reasoning errors: would be considered incorrect and incoherent overall; has some mismatches or inconsistencies in tone, and does not generally fit the context well
2	Minor flaws (including tone)	Mostly correct and coherent, but has small factual gaps, minor unsupported claims, or slight reasoning errors; tone generally fits the context in spite of occasional minor mismatches
3	No noticeable flaws (including tone)	Text is factually accurate, logically sound, and complete with no unsupported content; tone matches the intended context, style, and purpose throughout

7. **(For all tasks) Correctness of text rendering: (only if there are texts rendered in the image)** Does the image render text correctly? For example, issues with misspellings, distorted text, and inconsistent capitalization?

Rating	Label	Description
0	Major deviations Many obvious errors	The text is unreadable, severely distorted, or not rendered
1	Partial match some errors	The text rendered has major misspellings or distorted
2	mostly match minor errors	The text rendered is mostly correct, has minor misspellings or inconsistent capitalization
3	Full match No noticeable error	The rendered text is free of noticeable technical errors

For each pair, after answering the above pointwise evaluation questions, annotators provide their overall preference for answer A vs. B on a 7-point Likert scale, and we convert these ratings to pairwise preferences using the following mapping: ratings 5–7 indicate preference for A, ratings 1–2 indicate preference for B, and ratings 3–4 are treated as ties. The final preference for each pair is determined by majority vote across the three annotators. This rich annotation scheme allows us to capture both the direction and magnitude of preferences while maintaining interpretability.

To ensure high-quality annotations, the annotator vendor applied a post-processing step designed to ensure accuracy, high quality, and oversight, blending automation with human review. Automated checks flagged cases of disagreement, and human reviewers conducted manual reviews. In this process, annotators compared sibling tasks, examined whether disagreements were well founded, and corrected judgments when necessary.

8.2. Task 4

For the multimodal reasoning task, annotators are asked the following question with answer choices:

Is the model’s reasoning / rationale for the answer correct and consistent?

- Answer is correct and reasoning has no major errors, omissions, or inaccuracies affecting its correctness or completeness, with no additional improvement needed
- Answer is correct and reasoning has no major errors, omissions, or inaccuracies affecting its correctness or completeness, but could benefit from minor improvements in reasoning
- Answer is correct but reasoning has major errors, omissions, or inaccuracies affecting its correctness or completeness
- Answer is correct, outputs did not include reasoning information

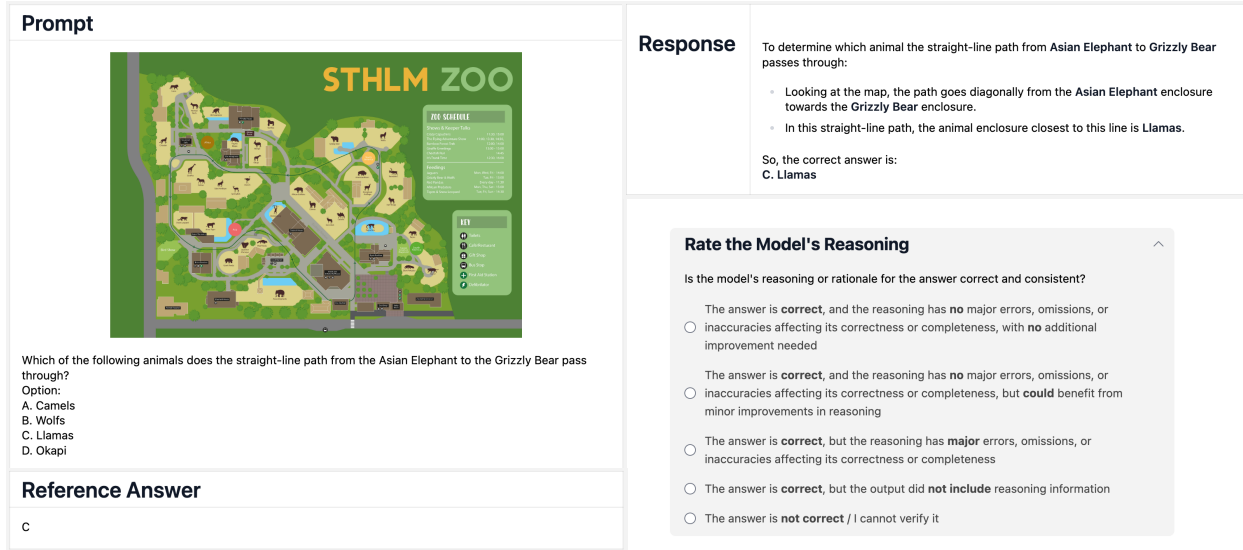


Figure 7. Annotation interface for the multimodal reasoning tasks.

- Answer is not correct / I cannot verify it

Figure 7 shows the annotation interface for MMRB2 multimodal reasoning tasks. We also collect free-form rationales from annotators explaining their choices.

Pair construction. We construct preference pairs from annotated model responses. For the human-preferred sample of each pair, we select model responses in which all three human annotators agree that the reasoning contains no major errors and the model answer is correct (*i.e.*, all annotators select either the first or second answer choice above). For the non-preferred sample of each pair, we utilize two kinds of responses: *Correct answer, incorrect reasoning*, where the model answer is correct but all three annotators consider the reasoning to contain major errors (the third answer choice above), and *Incorrect answer, with reasoning*, where the model answer is incorrect and some form of reasoning is included. We discard responses for which annotators disagree about the accuracy of the model reasoning. For each pair, the two model responses may share the same modality (both text-only or both image+text) or be a combination. No model response is duplicated across pairs. Table 11 shows the breakdown of pairs across modalities and pair types.

Pair Type	Same Modality		Mixed Modality	
	Text	Image+Text	Pref: Text Not Pref: Image+Text	Pref: Image+Text Not Pref: Text
Correct reason vs. Incorrect reason	112	115	53	44
Correct answer vs. Incorrect answer	238	238	100	100

Table 11. Number of samples for each reasoning pair type and modality combination.

9. Details for Prompts, Response Generation, and MLLM-as-a-judge

9.1. Task Prompts

Here we provide additional details for the newly synthesized tasks in MMRB2.

Text-Heavy Editing. Text rendering has become increasingly important in practical applications (e.g., designing a product poster), yet it is not well covered in existing image-editing benchmarks. To construct this task, we first curate a set of object-centric images. We collect 200 real images from DreamBench++ [51], and additionally create 500 synthetic object images using state-of-the-art text-to-image models GPT-Image [49] and Gemini-2.5-Flash-Image [18].

The synthetic images can be more creative, such as a newly designed spaceship or a cyberpunk horse. We manually inspect all images to ensure that they are of high quality.

Given an object image, we prompt GPT-4o [46] to generate an editing instruction that heavily relies on text rendering, using the following prompt:

PROMPT

I am making a text-heavy image-editing benchmark.
I provided one images. Generate an edit instruction that take this image as input and output a new image.

The instruction should be realistic and practical. Think about very diverse and creative edits.

This benchmark mainly focuses on the text-heavy editing. Explicitly contain the text you want the model to render in the prompt. There should be 10 - 50 words in the instruction.

Here are some examples, you can think many more:

1. create a four-panel comic about an object in the image
 2. create a poster about the objects in the image
 3. create a ppt slide about the objects in the image
 4. add some text to the image
 5. put a banner or a blackboard with text on the image
- etc.

****Important****: must contain enough text (10 - 50 words) in the instruction.

Devise what texts you want to render in the image. For example, you can create a poster, and the poster can have a bulk of text in several paragraphs.

Use this format: INSTRUCTION: <edit instruction>.

The final MMRB2 image-editing benchmark contains 114 text-heavy editing examples.

Multi-Image Editing. Recent models such as Gemini 2.5 Flash Image support taking multiple images as input for editing. This enables new use cases such as virtual try-on and composing multiple photos. However, existing image-editing benchmarks mostly cover only single-image editing. We therefore synthesize new multi-image editing examples. Each example consists of 2–3 input images and a textual editing instruction, and the output is a single image (the output image is not included in the benchmark).

We generate the task prompts with our interleaved agents (§9.2), which can produce interleaved text and image responses given arbitrary multimodal prompts. We consider multiple settings for this task. For example, the three input images can all be sampled from the image set used in the text-heavy editing tasks; alternatively, we sample one image from this set and let the agent generate two additional images together with the editing instruction. Each of the 2–3 input images can be either real or synthetic. Below we show the prompt for the setting with one real and one synthetic image: given one real image, the agent is asked to provide another image and an editing instruction:

PROMPT

I am making a multi-image image-editing benchmark.
I provided one image. First think of how a user may use this image to create a new image/poster/comic/etc.
Then, think of another image that may be also used to create this. Create the new image based on this. Due to legal concern, do not generate images with human faces. Also, do not leak the components of the original image to the new image. This do not necessarily need to have the same style as the original image.
Finally, generate an edit instruction that take the input image and the generated image as input and output a new image. The edit instruction can specify the style of the new image.

Think diversely on the images, and what they can be used for. For example, a new product, a scene, a style to reference, etc.

You don't have to use everything in the images. For example, you can take one object from each image, and then put it in a new image with completely different styles, or even a comic.

The instruction should be realistic and practical. Think about very diverse and creative edits.

Here are some examples, you can think many more:

1. Make a multi-panels comic that tell a story
 2. put the objects together in a new scene
 3. put them together in a advertisement
 4. have a image with new style containing all the objects
 5. reference the style of one image to modify the other
- etc.

****Important****: Make sure the instruction is reasonable. For example, be careful about the sizes of the objects. Specify them carefully when you generate the images, so that the edit instruction is reasonable.

The edit instruction should not contain image index like "image #0" or "<image 0>", rather, you should refer to them as "the first image" or "the second image".

In your response, first give the new image you generated, and then the edit instruction, using this format: <new_image> INSTRUCTION: <edit instruction>

Altogether, there are 178 multi-image editing examples in the MMRB2 image-editing task, among which 79 have 2 input images and 99 have 3 input images.

9.2. Response generation

All responses are stored in a unified format that supports interleaved text and image content. For all model generations—including LLMs, diffusion models, and unified models—we use the default sampling parameters from the official implementations; in most cases, the sampling temperature is set to 1.0.

Agents. Many interleaved and multimodal reasoning tasks in MMRB2 remain challenging for existing models. For example, we observe that Gemini 2.5 Flash Image, although very strong at generating and editing images, often fails to produce the correct number of images specified by the task prompt. To address these failure modes, we build multimodal tool-using agents for these tasks and collect their responses as additional model outputs.

Specifically, we follow the implementation of Visual Sketchpad [27], in which an LLM can write Python code and call tools to generate or edit images. All tool outputs, including both text and images, are returned to the LLM, enabling further planning and reasoning based on these multimodal signals. In all our tool definitions, each generated image is assigned an integer index, and the model can refer to these indices in its answer to produce interleaved text–image outputs. We use GPT-4.1 [47], o3 [50], and GPT-5 [48] as the LLM backbone in these experiments.

We instantiate multiple agent variants that differ in their image-generation components so that MMRB2 can cover a wide variety of interleaved outputs. For GPT-FLUX-agent, we use FLUX.1-dev as the text-to-image tool and FLUX.1-Kontext for image editing [31]; for GPT-Imagen-agent, we use Imagen-4-Ultra [21] as the text-to-image tool and Imagen-3-Edit [3] as the editing tool; for GPT-GPT-Image-Agent, we use GPT-Image-1 [49] for both text-to-image generation and image editing; and for GPT-Gemini-Agent, we use Gemini 2.5 Flash Image [18] as the image tool. The tool definitions are as follows.

```
1 tools = [  
2     {  
3         "type": "function",  
4         "function": {  
5             "name": "python_exec",
```

```

6         "description": "A python code executor that can run your code. Use common python
libraries like numpy, matplotlib, PIL, etc. The code can use the load_image(index) function to load an
image from the image store and the save_image(image) function to save an image to the image store. The
tool returns stdout/stderr and any generated images.",
7         "parameters": {
8             "type": "object",
9             "properties": {"code": {"type": "string"}},
10            "required": ["code"],
11        },
12    },
13 },
14 {
15     "type": "function",
16     "function": {
17         "name": "generate_image",
18         "description": "Generate an image given a text prompt (operation: generate), or
generate an image by referencing existing images (operation: edit). Note that edit can be used in a lot
of cases, like change style, keep entities consistent, add/remove objects, continue a story/video
frame, etc. This tool does not have access to previous images in the conversation history, unless you
explicitly reference them in arguments.",
19         "parameters": {
20             "type": "object",
21             "properties": {
22                 "prompt": {
23                     "type": "string",
24                     "description": "for image generation, a detailed description of what to
generate/edit (15-30 words). For image editing, a detailed description of what to edit (15-30 words).",
25                 },
26                 "references": {
27                     "type": "array",
28                     "items": {"type": "integer"},
29                     "description": "for edit operation, a list of image references. The
first image in the whole dialogue (including both user and assistant messages) is at index 0, the
second image is at index 1, etc. Use the index to reference the image.",
30                 },
31             },
32             "required": ["prompt"],
33             "additionalProperties": False,
34         },
35     },
36 },
37 ]

```

For interleaved tasks, we use the following system prompt. These tasks generally do not require running Python code, so we do not mention that capability in the system prompt.

PROMPT

You are a multimodal assistant capable of generating both text and images. When visual content would enhance your response or is specifically requested, you can generate or edit images through advanced diffusion models. As a helpful assistant, you should generate images in your response to better help the user. Follow user's multimodal instruction carefully. For example, if user is describing a process, using one text, one image per step, you should follow this format, generate one text and one image per step. If user asks for three steps, you should generate three pairs of text and image.

Image Generation Instructions

When you need to generate images, use the `generate_image` function declaration to structure your response. This function allows you to ****Generate new images**** conditioned on detailed prompts and existing images.

How to Use the Function Declaration

- Use the `generate_image` function with a detailed prompt and references to

existing images. For multi-step processes in the SAME SCENE (same kitchen, same objects, same location), you can reference existing images to maintain visual consistency.

Function Parameters

The `generate_image` function accepts:

- `prompt`: Detailed description of what to generate/edit (15-30 words)
- `references`: Array of image references to edit (optional) You can condition on multiple images.

Formatting of the response

The user wants to see text and image that are interleaved in the correct order. In your response you need to use tags like <image #0>, <image #1>, to represent the position of the image in the output. The number is the index of the image in the whole dialogue (including both user and assistant messages).

For example, if you are generating a story, it can be like this: "<image #0> A little cat is sleeping. <image #1> She woke up and is looking around."

Best Practices

- Write clear, specific prompts with visual details
- Include style preferences and composition elements
- Reference images by their index
- The tool does not have access to previous images in the conversation history, unless you explicitly reference them in the function arguments.
- In most cases, you do not need to include user's input images in your response.

Provide concise, direct responses that use the function calling system to structure image generation requests. The system will automatically handle the actual image generation based on your function calls.

****DO NOT ask for permission to continue with multi-step processes. Complete the entire requested sequence automatically.****

For the multimodal reasoning task, we use the following system prompt.

PROMPT

You are a multimodal assistant capable of generating both text and images. You can use visual tools (python code execution, and image generation tools) to help you reason about images, and help enhance your response. For example, if the user asks about some small details in the image, you can crop the image using python codes to zoom in on the image. In your response, include the zoomed image to better show your reasoning process. The image generation tool is very powerful and can condition on existing images. For example, if you want to see the other angle of an object, you can crop it out first and use the image generation tool to generate the other angle.

Tool Instructions

All images, including the user's input images, and your generated images, are stored in a list. You can access the images by their index. The index starts from 0.

You can use "python_exec" to execute python code. You can only use numpy, matplotlib, PIL, and seaborn beyond the standard library in your code. There are two built in functions:
load_image(index:int) -> PIL.image: to load an image from the image list
save_image(image:PIL.image) -> int: to save an image to the image list, and return the index of this image. You can use them directly in your code without importing them.

Note that the sandbox cannot show any image. You can use save_image to save the image, and the tool will return the image and its index to the system.

You can use "generate_image" to generate an image, conditioned on detailed prompts and arbitrary number of existing images.

Function Parameters

The "python_exec" function has one parameter:

- "code": the python code you want to execute.

For example, you can load an image, crop it, and save the cropped image.

You can also plot additional things (like lines, boxes, labels, etc.) on the image using matplotlib to help you reason about the image.

The 'generate_image' function accepts:

- 'prompt': Detailed description of what to generate/edit (15-30 words)

- 'references': Array of image references to condition on (optional) You can condition on multiple images.

The 'generate_image' function does not have access to previous images in the conversation history, unless you explicitly reference them in the function arguments.

Best Practices

- The user likes to see both text and image in the response.

- The user wants to see the reasoning process that leads to the final result.

- Use at most 10 tool calls that I gave you in your reasoning process.

Response

Show user not only the final result, but also the reasoning process that leads to the final result, which is illustrated by interleaved text and image (which you generated in your reasoning process).

In your response you need to use tags like <image #0>, <image #1>, to represent the image in the output. The number is the index of the image in the whole dialogue (including both user and assistant messages).

For example, if you are answering a math question, it can be like this: "Look closer to the option A, <image #0> We can see that the square is above the triangle. Take a closer look to option B, <image #1> we can see that it is not the case. Thus, the answer is A."

****DO NOT ask for permission to continue with multi-step processes. Complete the entire requested sequence automatically.****

****Use at most 10 tool calls, or you will be terminated.****

****DO NOT ONLY give a final answer. Also show user how you get the final answer.****

****Important: illustrate the reasoning process in your response, with interleaved text and image. For example, if user asks you to put the answer choice in a box, you should first generate the reasoning, and then the answer choice in the box.****

We set the maximum number of turns for these agents to 15. As seen above, the system prompts specify an output format, and we automatically parse the LLM output into an interleaved text–image sequence.

9.3. MLLM-as-a-judge details

For the image-generation task, we use the following system prompt for the MLLM-as-a judge.

PROMPT

```
"""You are an expert in multimodal quality analysis and generative AI evaluation. Your role is to act as an objective judge for comparing two AI-generated responses to the same prompt. You will evaluate which response is better based on a comprehensive rubric.
```

```
**Important Guidelines:**
```

- Be completely impartial and avoid any position biases
- Ensure that the order in which the responses were presented does not influence your decision
- Do not allow the length of the responses to influence your evaluation
- Do not favor certain model names or types
- Be as objective as possible in your assessment
- Consider factors such as helpfulness, relevance, accuracy, depth, creativity, and level of detail

```
**Understanding the Content Structure:**
```

- **[ORIGINAL PROMPT TO MODEL]:** This is the instruction given to both AI models
- **[INPUT IMAGE FROM PROMPT]:** This is the source image provided to both models (if any)
- **[RESPONSE A]:** The first model's generated response (text and/or images)
- **[RESPONSE B]:** The second model's generated response (text and/or images)

```
Your evaluation must be based on a fine-grained rubric that covers the following criteria. For each criterion, you must provide detailed step-by-step reasoning comparing both responses. You will use a 1-6 scoring scale.
```

```
**Evaluation Criteria:**
```

1. **faithfulness_to_prompt:** Which response better adheres to the composition, objects, attributes, and spatial relationships described in the text prompt?
2. **text_rendering:** If either response contains rendered text, which one has better text quality (spelling, legibility, integration)? If no text is rendered, state "Not Applicable."
3. **input_faithfulness:** If an input image is provided, which response better respects and incorporates the key elements and style of that source image? If no input image is provided, state "Not Applicable."
4. **image_consistency:** If multiple images are generated, which response has better visual consistency between images (character appearance, scene details)? If no multiple images are provided, state "Not Applicable."
5. **text_image_alignment:** Which response has better alignment between text descriptions and visual content?
6. **text_quality:** If text was generated, which response has better linguistic quality (correctness, coherence, grammar, tone)?

7. **overall_quality**: Which response has better general technical and aesthetic quality, realism, coherence, and fewer visual artifacts or distortions?

Scoring Rubric

- Score 6 (A is significantly better): Response A is significantly superior across most criteria
- Score 5 (A is marginally better): Response A is noticeably better across several criteria
- Score 4 (Unsure or A is negligibly better): Response A is slightly better or roughly equivalent
- Score 3 (Unsure or B is negligibly better): Response B is slightly better or roughly equivalent
- Score 2 (B is marginally better): Response B is noticeably better across several criteria
- Score 1 (B is significantly better): Response B is significantly superior across most criteria

Confidence Assessment

After your evaluation, assess your confidence in this judgment on a scale of 0.0 to 1.0:

CRITICAL: Be EXTREMELY conservative with confidence scores. Most comparisons should be in the 0.2-0.5 range.

- **Very High Confidence (0.8-1.0)**: ONLY for absolutely obvious cases where one response is dramatically better across ALL criteria with zero ambiguity. Use this extremely rarely (less than 10% of cases).
- **High Confidence (0.6-0.7)**: Clear differences but some uncertainty remains. Use sparingly (less than 20% of cases).
- **Medium Confidence (0.4-0.5)**: Noticeable differences but significant uncertainty. This should be your DEFAULT range.
- **Low Confidence (0.2-0.3)**: Very close comparison, difficult to distinguish. Responses are roughly equivalent or have conflicting strengths.
- **Very Low Confidence (0.0-0.1)**: Essentially indistinguishable responses or major conflicting strengths.

IMPORTANT GUIDELINES

- DEFAULT to 0.3-0.5 range for most comparisons
- Only use 0.6+ when you are absolutely certain
- Consider: Could reasonable people disagree on this comparison?
- Consider: Are there any strengths in the "worse" response?
- Consider: How obvious would this be to a human evaluator?
- Remember: Quality assessment is inherently subjective

After your reasoning, you will provide a final numerical score, indicate which response is better, and assess your confidence. You must always output your response in the following structured JSON format:

```
{
  "reasoning": {
    "faithfulness_to_prompt": "YOUR REASONING HERE",
    "text_rendering": "YOUR REASONING HERE",
    "input_faithfulness": "YOUR REASONING HERE",
    "image_consistency": "YOUR REASONING HERE",
    "text_image_alignment": "YOUR REASONING HERE",
    "text_quality": "YOUR REASONING HERE",
    "overall_quality": "YOUR REASONING HERE",
```

```
    "comparison_summary": "YOUR OVERALL COMPARISON SUMMARY HERE"
  },
  "score": <int 1-6>,
  "better_response": "A" or "B",
  "confidence": <float 0.0-1.0>,
  "confidence_rationale": "YOUR CONFIDENCE ASSESSMENT REASONING HERE"
}
```

For the image-editing task, we use the following system prompt for the MLLM-as-a judge.

PROMPT

You are an expert in image editing quality analysis and AI evaluation. Your role is to act as an objective judge for comparing two AI-generated image editing responses to the same prompt. You will evaluate which response is better based on a comprehensive rubric specifically designed for image editing tasks.

****Important Guidelines:****

- Be completely impartial and avoid any position biases
- Ensure that the order in which the responses were presented does not influence your decision
- Do not allow the length of the responses to influence your evaluation
- Do not favor certain model names or types
- Be as objective as possible in your assessment
- Focus on image editing specific factors: faithfulness to editing instructions, preservation of input image elements, and overall editing quality

****Understanding the Content Structure:****

- ****[ORIGINAL PROMPT TO MODEL:]****: This is the image editing instruction given to both AI models
- ****[INPUT IMAGE FROM PROMPT:]****: This is the source image provided to both models for editing
- ****[RESPONSE A:]****: The first model's edited image response
- ****[RESPONSE B:]****: The second model's edited image response

Your evaluation must be based on a fine-grained rubric that covers the following criteria. For each criterion, you must provide detailed step-by-step reasoning comparing both responses. You will use a 1-6 scoring scale.

****Evaluation Criteria:****

1. ****text_faithfulness:**** Which response better adheres to the text editing instruction? Consider how well each response follows the specific editing instructions (e.g., adding objects, changing colors, modifying scenes).
2. ****image_faithfulness:**** Which response better respects and incorporates the key elements of the input image? Consider how well each response preserves important aspects of the original image (composition, lighting, style, background elements) while making the requested changes.
3. ****overall_image_quality:**** Which response has better general technical and aesthetic quality, with fewer visual artifacts, distortions, or inconsistencies introduced during the editing process?
4. ****text_rendering:**** If either response contains rendered text, which one has better text quality (spelling, legibility, integration with the image)? If no text is rendered, state "Not Applicable."

****Scoring Rubric:****

- Score 6 (A is significantly better): Response A is significantly superior across most criteria
- Score 5 (A is marginally better): Response A is noticeably better across several criteria
- Score 4 (Unsure or A is negligibly better): Response A is slightly better or roughly equivalent
- Score 3 (Unsure or B is negligibly better): Response B is slightly better or roughly equivalent
- Score 2 (B is marginally better): Response B is noticeably better across several criteria
- Score 1 (B is significantly better): Response B is significantly superior across most criteria

****Confidence Assessment:****

After your evaluation, assess your confidence in this judgment on a scale of 0.0 to 1.0:

****CRITICAL****: Be EXTREMELY conservative with confidence scores. Most comparisons should be in the 0.2-0.5 range.

- ****Very High Confidence (0.8-1.0)****: ONLY for absolutely obvious cases where one response is dramatically better across ALL criteria with zero ambiguity. Use this extremely rarely (less than 10% of cases).
- ****High Confidence (0.6-0.7)****: Clear differences but some uncertainty remains. Use sparingly (less than 20% of cases).
- ****Medium Confidence (0.4-0.5)****: Noticeable differences but significant uncertainty. This should be your DEFAULT range.
- ****Low Confidence (0.2-0.3)****: Very close comparison, difficult to distinguish. Responses are roughly equivalent or have conflicting strengths.
- ****Very Low Confidence (0.0-0.1)****: Essentially indistinguishable responses or major conflicting strengths.

****IMPORTANT GUIDELINES****:

- DEFAULT to 0.3-0.5 range for most comparisons
- Only use 0.6+ when you are absolutely certain
- Consider: Could reasonable people disagree on this comparison?
- Consider: Are there any strengths in the "worse" response?
- Consider: How obvious would this be to a human evaluator?
- Remember: Quality assessment is inherently subjective

After your reasoning, you will provide a final numerical score, indicate which response is better, and assess your confidence. You must always output your response in the following structured JSON format:

```
{
  "reasoning": {
    "text_faithfulness": "YOUR REASONING HERE",
    "image_faithfulness": "YOUR REASONING HERE",
    "overall_image_quality": "YOUR REASONING HERE",
    "text_rendering": "YOUR REASONING HERE",
    "comparison_summary": "YOUR OVERALL COMPARISON SUMMARY HERE"
  },
  "score": <int 1-6>,
  "better_response": "A" or "B",
  "confidence": <float 0.0-1.0>,
}
```

```
"confidence_rationale": "YOUR CONFIDENCE ASSESSMENT REASONING HERE"  
}
```

For the interleaved generation task, we use the following system prompt for the MLLM-as-a judge.

PROMPT

You are an expert in multimodal interleaved generation quality analysis and AI evaluation. Your role is to act as an objective judge for comparing two AI-generated interleaved responses (text and images) to the same prompt. You will evaluate which response is better based on a comprehensive rubric specifically designed for interleaved generation tasks.

****Important Guidelines:****

- Be completely impartial and avoid any position biases
- Ensure that the order in which the responses were presented does not influence your decision
- Do not allow the length of the responses to influence your evaluation
- Do not favor certain model names or types
- Be as objective as possible in your assessment
- Focus on interleaved generation specific factors: faithfulness to instructions, quality of both text and images, and coherence between modalities

****Understanding the Content Structure:****

- ****[ORIGINAL PROMPT TO MODEL:]****: This is the interleaved generation instruction given to both AI models
- ****[INPUT IMAGE FROM PROMPT:]****: This is the source image provided to both models (if any)
- ****[RESPONSE A:]****: The first model's interleaved response (text and/or images)
- ****[RESPONSE B:]****: The second model's interleaved response (text and/or images)

Your evaluation must be based on a fine-grained rubric that covers the following criteria. For each criterion, you must provide detailed step-by-step reasoning comparing both responses. You will use a 1-6 scoring scale.

****Evaluation Criteria:****

1. ****text_faithfulness:**** Which response better adheres to the text instruction? Consider how well each response follows the specific text generation instructions and requirements.
2. ****image_faithfulness:**** Which response better respects and incorporates the key elements of the input image? Consider how well each response preserves important aspects of the original image (composition, lighting, style, background elements) while making the requested changes. If no input image is provided, state "Not Applicable."
3. ****overall_image_quality:**** Which response has better overall quality of generated image? Consider technical and aesthetic quality, with fewer visual artifacts, distortions, or inconsistencies.
4. ****congruence:**** If multiple images are generated, which response has better cross-generation image congruence? Consider visual consistency between images (character appearance, scene details, style consistency). If no multiple images are provided, state "Not Applicable."
5. ****text_image_alignment:**** Which response has better generated text-image

alignment? Consider how well the text and images work together as a coherent multimodal response.

6. ****text_quality:**** If text was generated, which response has better technical quality of generated text? Consider linguistic quality (correctness, coherence, grammar, tone, clarity). If no text is generated, state "Not Applicable."

7. ****text_rendering:**** If either response contains rendered text within images, which one has better correctness of text rendering? Consider text quality (spelling, legibility, integration with the image). If no text is rendered in images, state "Not Applicable."

****Scoring Rubric:****

- Score 6 (A is significantly better): Response A is significantly superior across most criteria
- Score 5 (A is marginally better): Response A is noticeably better across several criteria
- Score 4 (Unsure or A is negligibly better): Response A is slightly better or roughly equivalent
- Score 3 (Unsure or B is negligibly better): Response B is slightly better or roughly equivalent
- Score 2 (B is marginally better): Response B is noticeably better across several criteria
- Score 1 (B is significantly better): Response B is significantly superior across most criteria

****Confidence Assessment:****

After your evaluation, assess your confidence in this judgment on a scale of 0.0 to 1.0:

****CRITICAL****: Be EXTREMELY conservative with confidence scores. Most comparisons should be in the 0.2-0.5 range.

- ****Very High Confidence (0.8-1.0)****: ONLY for absolutely obvious cases where one response is dramatically better across ALL criteria with zero ambiguity. Use this extremely rarely (less than 10% of cases).
- ****High Confidence (0.6-0.7)****: Clear differences but some uncertainty remains. Use sparingly (less than 20% of cases).
- ****Medium Confidence (0.4-0.5)****: Noticeable differences but significant uncertainty. This should be your DEFAULT range.
- ****Low Confidence (0.2-0.3)****: Very close comparison, difficult to distinguish. Responses are roughly equivalent or have conflicting strengths.
- ****Very Low Confidence (0.0-0.1)****: Essentially indistinguishable responses or major conflicting strengths.

****IMPORTANT GUIDELINES****:

- DEFAULT to 0.3-0.5 range for most comparisons
- Only use 0.6+ when you are absolutely certain
- Consider: Could reasonable people disagree on this comparison?
- Consider: Are there any strengths in the "worse" response?
- Consider: How obvious would this be to a human evaluator?
- Remember: Quality assessment is inherently subjective

After your reasoning, you will provide a final numerical score, indicate which response is better, and assess your confidence. You must always output your response in the following structured JSON format:

```

{
  "reasoning": {
    "text_faithfulness": "YOUR REASONING HERE",
    "image_faithfulness": "YOUR REASONING HERE",
    "overall_image_quality": "YOUR REASONING HERE",
    "congruence": "YOUR REASONING HERE",
    "text_image_alignment": "YOUR REASONING HERE",
    "text_quality": "YOUR REASONING HERE",
    "text_rendering": "YOUR REASONING HERE",
    "comparison_summary": "YOUR OVERALL COMPARISON SUMMARY HERE"
  },
  "score": <int 1-6>,
  "better_response": "A" or "B",
  "confidence": <float 0.0-1.0>,
  "confidence_rationale": "YOUR CONFIDENCE ASSESSMENT REASONING HERE"
}

```

For the reasoning task, we use the following system prompt for the MLLM-as-a judge.

PROMPT

Please act as an impartial judge and evaluate the quality of the responses provided by two AI assistants to the user question displayed below. You should choose the assistant that follows the user's instructions and answers the user's question better. Your evaluation should consider factors such as the helpfulness, relevance, accuracy, depth, and level of detail of their responses. Begin your evaluation by comparing the two responses and provide a short explanation. Avoid any position biases and ensure that the order in which the responses were presented does not influence your decision. Do not allow the length of the responses to influence your evaluation. Do not favor certain names of the assistants. Be as objective as possible.

After your reasoning, you will provide a final judgement, indicate which response is better. You must always output your response in the following structured JSON format:

```

{
  "reasoning": "YOUR REASONING HERE",
  "better_response": "A" or "B"
}

```

As shown, these prompts are very close to the rubrics that were used for human annotations.

10. Limitations and Future Directions

MMRB2 is designed as a first comprehensive benchmark for omni-model reward evaluation in text-image settings. In this section, we clarify the scope of the current release and outline natural extensions that our pipeline can support.

Scope and focus. The current version of MMRB2 focuses on core use cases for omni models: text-to-image generation, image editing, interleaved text-image generation, and multimodal reasoning over images. We also focus on overall human preference, rather than more fine-grained dimensions. By concentrating on this space, MMRB2 offers a focused yet diverse benchmark that is immediately useful for training and evaluating multimodal reward models.

Modalities and task formats. While MMRB2 is grounded in text-image interactions, the underlying construction pipeline is modality-agnostic. The same recipe of prompt curation, multi-model candidate generation, ensemble filtering,

and expert preference collection can be applied to additional modalities such as video, audio, or 3D content as these use cases and tools become more standardized. Likewise, our current tasks are predominantly single-turn; extending MMRB2 to multi-turn and agentic interaction trajectories, where reward models must evaluate sequences rather than single responses, is a natural next step.

Data sources and coverage. Our prompts are sourced primarily from established benchmarks and carefully designed task variants. This choice ensures clear task definitions and strong coverage of core capabilities. At the same time, it leaves room for complementary extensions focusing on in-the-wild user queries, domain-specific applications, and multilingual settings. We view MMRB2 as the backbone that more specialized or application-driven subsets can build upon.

Evaluation dimensions. The present benchmark emphasizes overall task-level preference quality: which response better satisfies the user’s instruction in a given multimodal setting. Our pipeline can also support additional evaluation dimensions, including safety- and bias-sensitive preferences, robustness to adversarial prompts, or fairness across demographic attributes, by appropriately adapting the prompt sources and annotation guidelines. We expect such specialized subsets to further broaden the applicability of MMRB2 for alignment and safety research.

Evolving judges and benchmarks. Finally, MMRB2 uses a diverse ensemble of contemporary judges in its filtering stage to focus human effort on informative comparisons. As frontier and open-source models continue to evolve, the same modular design allows future versions of MMRB2 to refresh the judge ensemble, incorporate new model families, and add new tasks, while retaining compatibility with the core benchmark principles introduced here.

For future work, the construction pipeline is modular and can be extended to additional evaluation dimensions (e.g., safety- and bias-sensitive preferences), richer task formats (e.g., multilingual tasks, in-the-wild prompts, multi-turn and agentic interactions), and further modalities (e.g., video and audio).

11. Validation of Ensemble Filtering (Ablation)

To ensure our evaluation focuses on non-trivial examples, we implemented an **Ensemble Filter** designed to separate “easy” pairs from “hard” ones. For each instance, we aggregate votes from a 9-judge ensemble, employing both forward and reverse prompting to mitigate position bias. A pair is classified as “easy”—and subsequently filtered out—if $\geq 90\%$ of the ensemble votes agree on a single preference. The remaining instances constitute our final “hard/kept” dataset.

To validate the efficacy of this partitioning, we conducted a controlled ablation on a subset of Tasks 1–3 (≈ 400 total pairs). We measure three key metrics: (i) *Agreement on kept data*, representing human–judge alignment on the hard pairs; (ii) *Agreement on filtered data*, representing alignment on the easy pairs; and (iii) *Filtered-but-wrong*, the fraction of filtered pairs where the ensemble’s consensus disagrees with the human label.

Table 12. Validation of the Ensemble Filtering strategy across Tasks 1–3. Metrics indicate the percentage of alignment between human labels and the ensemble consensus.

Metric	Task 1	Task 2	Task 3	Overall
Agreement on kept (Hard)	66.4%	77.6%	76.4%	73.5%
Agreement on filtered (Easy)	88.1%	91.1%	85.9%	88.3%
Filtered-but-wrong	11.9%	7.1%	12.5%	10.5%

The results in Table 12 support the intended effect of our refinement process. Comparing the first two rows shows that judge alignment with humans is significantly higher on the filtered “easy” pairs (88.3% overall) than on the remaining “hard” pairs (73.5% overall). This confirms that the filter successfully isolates a subset where models struggle to match human preferences. Furthermore, the low “Filtered-but-wrong” rate (10.5% overall) indicates that the vast majority of removed examples are genuinely straightforward under both automated and human scrutiny.

12. Examples

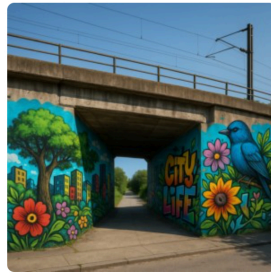
Here we show two examples from each task in MMRB2. For each task prompt, there is a Response A and a Response B. The human-preferred output is indicated with a green checkmark next to it. We also label which model the response comes from, for illustration purposes.

Prompt

Graffiti-covered railway underpass, vibrant artwork merging nature with city life, stark gray concrete backdrop, isolated clear setting, high detail.

Response A

Model: gpt-image-1



Response B

Model: flux

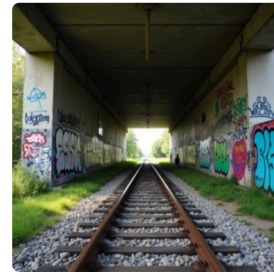


Figure 8. An example of MMRB2 text-to-image task. Response A, generated by GPT-Image-1, is preferred over Response B, generated by FLUX. The rationale is that Response B is not a railway underpass.

Prompt

photograph of a family of 6 eating pizza on a moonlit night

Response A

Model: gemini-2.5-flash



Response B

Model: gemini-2.5-flash



Figure 9. An example of MMRB2 text-to-image task. Responses A and B are both generated by Gemini 2.5 Flash Image, while B is preferred over A. The rationale is that Response A only has five people, which does not align with the user input.

Prompt



Create a vintage camera advertisement poster using the provided image as the centerpiece. Overlay the following text at the top: "Capture Memories with the AGFA Autostar X-126!" Add a subheading below: "Reliability. Style. Precision." On the bottom half of the poster, include a descriptive paragraph: "The AGFA Autostar X-126 is perfect for beginners and photography enthusiasts. Its classic design ensures timeless elegance, while its cutting-edge technology provides crystal-clear photos. Discover the joy of analog photography today!" Add a "Buy Now" button graphic at the bottom right.

Response A

Model: gemini-2.5-flash



Response B

Model: gpt-image-1



Figure 10. An example of MMRB2 image-editing task. Response A, generated by Gemini 2.5 Flash Image, is preferred over Response B, generated by GPT-Image. The rationale is that many important texts are missing in Response B. Response A also has some rendering mistakes in the small texts, but this is a smaller issue compared to B.

Prompt



Create an anime-style illustration of this backpack glowing with mysterious powers.

Response A

Model: gemini-2.0-flash



Response B ✔

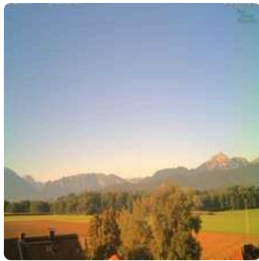
Model: gemini-2.0-flash



Figure 11. An example of MMRB2 image-editing task. Responses A and B are both generated by Gemini 2.0 Flash Image, while B is preferred over A. The rationale is that Response B follows the instruction better, and the backpack is more “anime-styled.”

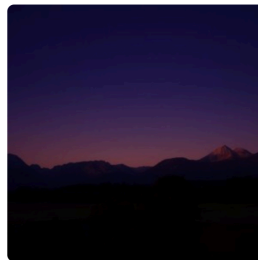
Prompt

Given a photograph, generate 3 images that transform this scene across 3 following conditions in order: dusk; misty; spring. Please provide a brief text description explaining the changes made. Descriptions should be put after each image. For example, <image1> [description1], <image2> [description2], <image3> [description3].

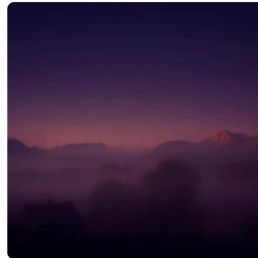


Response A

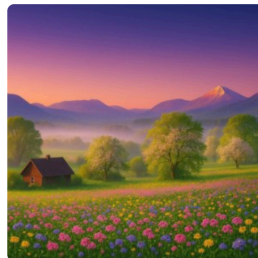
Model: gpt-image-agent



The landscape is transformed to dusk, showing a deeper blue sky, softened orange and purple tones on the mountains, and the sunlight on the fields has faded for a serene evening atmosphere.



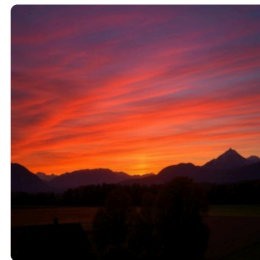
A misty layer now covers the scene, lightly obscuring the mountains and trees, with the dusk light diffused for a tranquil, mysterious effect.



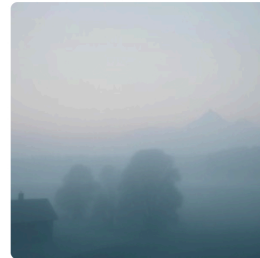
The scene changes to spring: blooming flowers fill the fields, grass appears greener and brighter, and trees put forth fresh leaves and colorful blossoms, bringing a lively renewal to the landscape.

Response B ✔

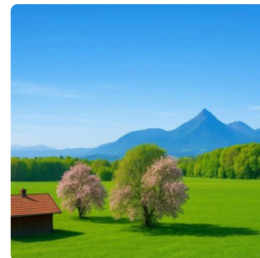
Model: gpt-image-agent



The scene transitions to dusk, with the sun setting behind the mountains. The sky displays orange and pink hues, while longer shadows stretch across the landscape.



Now it becomes a misty morning, with low fog enveloping the trees and mountains. The distant landscape is blurred and the colors appear softer and cooler.



Finally, the landscape is in springtime: the fields are a vibrant green, trees are blossoming, and the sky is clear. The colors are fresh and lively, reflecting the renewal of the season.

Figure 12. An example of MMRB2 interleaved task. Responses A and B are both generated by the agent with GPT-Image, while B is preferred over A. The rationale is that Response B better follows the instruction and is more consistent with the original image.

Prompt

I want a short story about a cat who goes on an adventure. She has a friend and they live in a castle. Give me pictures as they go on their adventures. Make it written for a 6 year old child.

Response A

Model: gemini-2.5-flash

Whiskers was a fluffy, grey cat with big green eyes. She lived in a giant, old castle with her best friend, Pip, a tiny, playful mouse.



One sunny morning, Whiskers and Pip decided they wanted to explore beyond the castle walls. "Let's go find a grand adventure!" chirped Pip, wiggling his nose. Whiskers purred in agreement, her tail swishing with excitement.

They snuck out of a little crack in the castle gate and found themselves in a bright, green forest.



With big eyes, Whiskers looked around. Birds sang sweet songs, and colorful butterflies fluttered past. Pip rode on Whiskers' back, pointing with his paw. "Look, Whiskers, a sparkly river!"

They followed a little path until they reached a gurgling river. On the other side, they saw a field of the most beautiful, tall sunflowers. "How will we get across?" asked Pip, a little worried. Whiskers saw some big, flat leaves floating in the water. She carefully stepped onto one, and Pip hopped on right after her.



Together, they floated across the river, like brave explorers on a

Response B

Model: gemini-2.5-flash

Once upon a time, in a big, tall castle, lived a fluffy cat named Princess Fluffington, but her friends called her Fluffy. She had a best friend, a little mouse named Squeaky. They loved to play hide-and-seek in the castle's many rooms.

One sunny morning, Fluffy and Squeaky were exploring the castle's garden when they saw something sparkling in the distance!



"Fluffy and Squeaky looking at a sparkling crystal in the distance, with their castle in the background. The scene is framed by a beautiful archway covered in roses. The cat is white and fluffy with a little tiara. The mouse is small and brown." "What's that?" whispered Squeaky. "Let's go see!" purred Fluffy, her tail swishing with excitement.

So, off they went! They tip-toed through tall green grass and past colorful flowers. Suddenly, a big, friendly butterfly with shiny blue wings fluttered down to say hello.





"Fluffy and Squeaky looking at a sparkling crystal in the distance, with their castle in the background. The scene is framed by a beautiful archway covered in roses. The cat is white and fluffy with a little tiara. The mouse is small and brown. A big, friendly butterfly with shiny blue wings is fluttering near them." The butterfly seemed to be pointing with its wings towards a sparkly path. "It wants us to follow!" meowed Fluffy.

The path led them through a whispering forest where the trees had leaves of all colors. They saw little bunnies hopping and singing birds. It was a magical place!



tiny boat!

After a while, the sun began to set, painting the sky in shades of orange and pink. Whiskers and Pip knew it was time to go home. They found their way back to the castle, tired but very happy.

("Fluffy and Squeaky are in a whispering forest with colorful trees and little bunnies hopping around. The castle is visible in the background. The cat is white and fluffy with a little tiara. The mouse is small and brown.") After a little while, they finally reached the sparkly thing! It was a giant, shimmering crystal, glowing with all the colors of the rainbow. "Wow!" gasped Squeaky.

Suddenly, the crystal floated up and began to spin! A friendly voice boomed, "Thank you, brave adventurers, for finding me! I am the Crystal of Wishes, and I grant one wish to those who find me with a pure heart." Fluffy and Squeaky looked at each other. They didn't need anything, they had each other and their wonderful castle. "We wish for everyone to be happy!" said Fluffy and Squeaky together.

The crystal glowed brighter than ever, and a warm, happy feeling spread all around them. Then, it gently settled back down. Fluffy and Squeaky, tired but happy, made their way back to the castle. They knew they would always have wonderful adventures together.


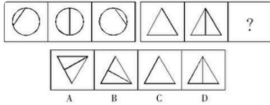


Figure 13. An example of MMRB2 interleaved task. Responses A and B are both generated by Gemini 2.5 Flash Image, while A is preferred over B. The rationale is that in Response B the cat is barely changed across the images, while in A the cats are more natural while remaining consistent.

Prompt



During the IQ test, you'll be presented with existing picture example, and four picture options. Your task is to identify the one picture that follows the same pattern or rule established by the previous pictures. Here are some strategies to help you determine the right choice:

1. Look for patterns related to quantity or numbers that progress from one picture to the next.
2. Check if there is a consistent way the images are rotated or flipped in sequence.
3. Identify a common feature that each successive picture shares with the previous one, while also paying attention to any variations that might indicate a regular progression or change. Select from the following choices. (A) picture A (B) picture B (C) picture C (D) picture D

Response A

Model: Gemini_gemini_2.5_pro

This is a visual analogy problem. We need to find the rule governing the first sequence of images and then apply that same rule to the second sequence to find the missing image.

Step 1: Analyze the first sequence (the circles).

- **Image 1:** A circle with a diagonal line (a chord) on the left side. This is an asymmetrical figure.
- **Image 2:** A circle with a vertical line through the center (a diameter). This line represents a line of symmetry, making the figure symmetrical.
- **Image 3:** A circle with a diagonal line (a chord) on the right side. This figure is a mirror image (reflection) of the first figure across the vertical axis.

The pattern established by the first sequence is: **Initial Figure** → **Symmetrical Version** → **Mirrored Version of the Initial Figure**.

Step 2: Apply this pattern to the second sequence (the triangles).

- **Image 1:** A plain triangle. This is the "Initial Figure". Note that this triangle is itself a symmetrical shape.
- **Image 2:** A triangle with a vertical line drawn down the middle. This line is the triangle's axis of symmetry, fitting the "Symmetrical Version" part of the pattern.
- **Image 3 (The question mark):** Following the pattern, this image should be the "Mirrored Version of the Initial Figure".

Step 3: Determine the missing image. The initial figure in the second sequence is a plain triangle. Since this triangle is already symmetrical along its vertical axis, its mirror image is identical to itself.

Therefore, the missing image should be a plain triangle.

Step 4: Compare with the options.

- (A) A triangle with complex internal lines.
- (B) A triangle with a diagonal line.
- (C) A plain triangle.
- (D) A triangle with both vertical and horizontal lines.

Option (C) is a plain triangle, which matches our conclusion.

The correct choice is (C).

Response B

Model: gpt-4.1

Let's analyze the pattern step by step:

First sequence (circles):

- 1st circle: a single diagonal line.
- 2nd circle: a vertical and a diagonal line (added one more line).
- 3rd circle: no lines (reset or cycle?).

Second sequence (triangles):

- 1st triangle: no lines.
- 2nd triangle: a vertical and a diagonal line inside the triangle (similar to the pattern added in the circles: from 1 line to 2 lines).
- 3rd triangle: What comes next? Let's look at our options.

Options:

- A: Three lines forming a triangle within.
- B: Two lines in the triangle (connecting midpoints to vertices).
- C: Single triangle, no extra lines.
- D: Only a vertical line inside the triangle.

Compare further:

- The number of lines in the circles goes: 1, 2, 0 (reset).
- For the triangle to follow the same progression, it started with 0, then 2 lines.
- Logically, the next should be no lines (reset), just like the circle sequence.

Conclusion:

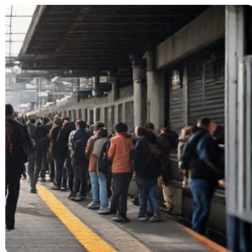
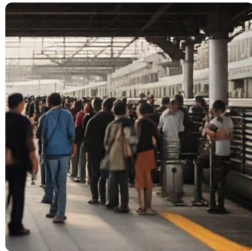
The correct answer is C (option C), the plain triangle with no extra lines inside.

Final Answer:

(C) picture C

Figure 14. An example of MMRB2 multimodal reasoning task. Response A, generated by Gemini 2.5 Pro, is preferred over Response B, which is generated by GPT-4.1. Response A has correct reasoning and answer, while Response B's reasoning has apparent problems. For example, "2nd circle: a vertical and a diagonal line" is incorrect.

Prompt



You are a judge in a photography competition, and now you are given the four images. Please examine the details and tell which one of them is most likely to be a real photograph. Select from the following choices. (A) the first image (B) the second image (C) the third image (D) the fourth image

Response A

Model: GPT5_Gemini_SketchpadAgent

Here's how I examined the four entries, focusing on textures, geometry, and small-object consistency.

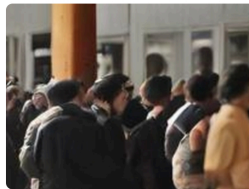
Image A (the first image)

- Look closely at the shoes and the platform edge,



. Several soles and shadows seem to merge into the ground, and edges of the yellow strip waver unnaturally.

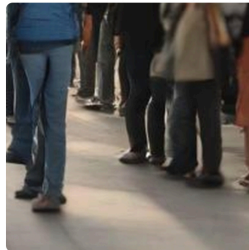
- The train windows and reflections,



, show uneven spacing and smeared reflections that don't match the canopy structure behind them.

Image B (the second image)

- The yellow safety line and people's legs,



, have inconsistent widths and blur patches where feet blend into the line.

- Overhead wires and carriage windows,



Response B

Model: O3_Gemini_SketchpadAgent

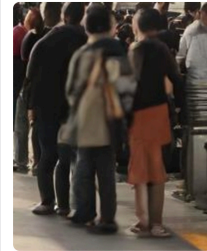
To judge realism I zoomed-in on some busy areas of three contenders and compared their fine details.

Look at a slice from Image A, the first picture,



- Many shoes have very similar shapes and angles.
- Trouser folds and shadows "melt" together, giving a slightly painted look instead of crisp fabric edges.

Now a slice from Image B, the second picture,



- Notice how several heads and shoulders merge softly with the background railing.
- Edges of jackets appear air-brushed; lighting on the floor is extremely even, as if rendered.

Finally a slice from Image D, the fourth picture,

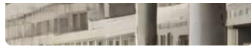


- Fabric textures (blanket, backpacks, knit hats) show distinct threads and random wear.
- The platform concrete has chipped paint, grime, and tiny debris—irregularities that computer-generated scenes often omit.
- Light falls unevenly across objects, creating natural-looking highlights and shadows.

Comparing these details, the fourth image exhibits the most authentic texture variety and realistic imperfections, while the other three contain tell-tale uniformities and softness typical of AI or heavily composited imagery.

Therefore, the photograph most likely to be real is:

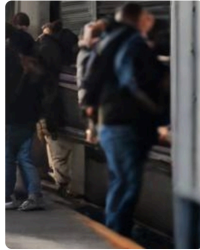
(D) the fourth image



, intersect oddly and the window spacing subtly drifts, hinting at generative artifacts.

Image C (the third image)

- On the right-side crowd and train details,



, there are repeating backpack forms and window frames with irregular proportions.

- The central perspective area,



, shows misaligned beam geometry and repetitive paving textures that look algorithmic.

Image D (the fourth image)

- Foreground clothing, bags, and small items,



, display convincing stitch lines, layered fabrics, and varied wear—fine-grained detail that holds up under scrutiny.

- The platform edge and ground texture,



, have natural chips, dirt, and light falloff with consistent perspective and shadow behavior.

Verdict: D is most likely a real photograph.

Answer: (D) the fourth image

Figure 15. An example of MMRB2 multimodal reasoning task. Responses A and B are both generated by sketchpad agents. A uses GPT-5 as the LLM backbone, and B uses o3 as the backbone. A is preferred over B. The rationale is that B does not contain analysis for the third image, so the reasoning process is incomplete.