MultiBanana: A Challenging Benchmark for Multi-Reference Text-to-Image Generation

Supplementary Material

A. Results of Qwen3-VL Judge

In Section 4, we report the average scores from Gemini 2.5 and GPT-5 as judges. As demonstrated in Section 4.4, Qwen3-VL-8B-Instruct [2] achieves strong correlation with human judgments comparable to GPT-5 [57] and Gemini 2.5 [19], confirming its reliability as a judge model (see Table 4). Although we used stable API versions for the closed-source judges, to ensure full reproducibility, we additionally employ Qwen3-VL-8B-Instruct, an open-weight, fixed-version model, as the judge. This also broadens access to our benchmark for researchers who do not have access to closed-source VLMs via APIs.

Table 5 and Table 6 present per-task total scores evaluated using Qwen3-VL as the sole judge, calculated in the same way as described in Section 3.4. The results are broadly consistent with those obtained using closed-model judges. The overall ranking of image generation models is preserved. We also observe the same key trends: performance degrades as the number of reference images increases, and background replacement tasks tend to yield lower scores across both open-source and closed-source models. These observations reinforce the validity of Qwen3-VL as a reliable, reproducible alternative. We expect this fixed, open-weight judge to serve as a dependable baseline for future evaluations on our benchmark.

Table 5. Per-task total scores for each single- and two-reference task, evaluated by Qwen3-VL. "Back." denotes the Background task. Both open-source and closed-source models exhibit lower performance on background replacement tasks. Note that for Nano Banana and GPT-Image-1, we use the versions available as of January 2026.

Model	**Single**	**Add**	**Back.**	**Color**	**Hair**	**Makeup**	**Material**	**Pose**	**Replace**	**Style**	**Text**	**Tone**
DreamOmni2	8.47	6.51	4.59	8.60	7.40	7.89	6.76	6.25	6.86	5.10	5.48	5.14
OmniGen2	9.04	5.34	6.04	5.07	6.70	6.68	6.12	5.91	6.34	3.24	4.29	4.00
Qwen-Image-Edit-2509	9.16	7.01	4.34	6.46	8.00	6.68	5.86	5.87	7.57	3.41	4.95	6.08
Nano Banana	9.45	8.13	7.97	9.67	9.16	9.25	8.95	8.76	8.30	9.03	8.81	8.52
GPT-image-1	9.51	8.09	8.08	9.70	9.04	9.24	9.17	8.80	8.30	9.10	8.52	8.78

Table 6. Per-task total scores for each multi-reference task, evaluated by Qwen3-VL. The local, global, back, and object columns under X correspond to X–1 Objects + Local, X–1 Objects + Global, X–1 Objects + Background, and X Object, respectively. Both open-source and closed-source models exhibit a general trend of decreasing scores across all tasks as the number of references increases. They also tend to perform worse on background replacement tasks, especially as the number of reference images increases. Note that for Nano Banana and GPT-Image-1, we use the versions available as of January 2026.

Model	**3-references**				**4-references**				**5-references**			
	local	global	back	object	local	global	back	object	local	global	back	object
DreamOmni2	4.39	4.27	4.33	3.68	3.80	2.95	2.77	2.90	3.23	2.12	2.86	2.26
OmniGen2	4.71	4.86	4.39	4.13	4.37	3.86	3.43	2.95	3.99	2.75	2.14	2.26
Qwen-Image-Edit-2509	5.22	5.81	5.81	5.63	3.24	4.51	4.42	3.94	1.05	1.48	1.28	1.10
Nano Banana	7.63	7.21	6.96	6.91	7.96	6.77	6.47	6.78	7.96	6.54	6.03	5.87
GPT-image-1	7.83	7.26	6.99	6.72	7.73	6.74	5.94	6.83	7.92	6.54	6.16	5.83

Model	**6-references**				**7-references**				**8-references**			
	local	global	back	object	local	global	back	object	local	global	back	object
DreamOmni2	3.34	2.26	1.72	1.93	2.61	2.45	2.10	1.81	2.44	1.65	1.83	1.59
OmniGen2	-	-	-	-	-	-	-	-	-	-	-	-
Qwen-Image-Edit-2509	1.00	1.49	1.00	1.00	1.00	1.46	1.00	1.13	1.00	1.00	1.00	1.00
Nano Banana	6.84	6.08	5.34	5.64	6.37	5.78	5.13	5.44	6.46	6.24	5.31	5.28
GPT-image-1	7.11	5.86	5.51	5.92	6.81	5.78	5.37	5.74	7.07	5.86	5.14	5.55

B. Details on Benchmark Construction

B.1. Image Collection

We constructed the reference image collection for our benchmark using both real images sourced from LAION-5B [68] and synthetic images generated by Nano Banana [22] and GPT-Image-1 [56]. By combining real and synthetic data, we mitigated the categorical biases inherent in the real-image distribution while expanding the coverage and diversity of the reference set. To generate synthetic images, we designed distinct axes of variation for four major categories: humans, animals, objects, and text-containing images.

Human Category. We systematically combined attributes such as makeup, hairstyle, hair color, pose, strong facial expressions, lighting, and overall tonal style to create a wide range of appearance conditions. Examples include ethnic or fashion-model-style makeup, vivid or unconventional hair colors, model-like or manga-like poses, strong emotional expressions (e.g., anger, joy, surprise), dramatic lighting (e.g., window light through blinds or neon illumination), and global color palettes (e.g., sepia or bluish). These conditions were intentionally selected to supplement the variations underrepresented in real-world data.

Animal Category. We first specified major species groups (dogs, cats, birds, reptiles, wild animals, and fantasy animals). We then added attributes to enhance the diversity of animal imagery. These included human-like behaviors (e.g., a bear reading a book, a fox playing in a rock band), rare colors (e.g., a blue horse or a pink panda), distinctive lighting conditions, and varied global tones such as sepia. This incorporated rare or extreme examples that seldom appear in real datasets.

Object Category. We expanded diversity along two axes: intrinsic object attributes and photographic conditions. The former included distinctive textures (e.g., denim, leather, knit), characteristic prints (e.g., floral, striped, checkered), and variations in material or design (e.g., matte, glossy, transparent). The latter included variations in lighting and global tonal adjustments (e.g., sepia, cool color tones), thereby ensuring a broad range of appearance conditions for objects.

Images Containing Text. Images containing text, which were largely absent from real data, were generated exclusively using GPT-Image-1, as Nano Banana exhibited limited ability to render non-English text. We defined three levels of text length (a single word, a short phrase of up to five words, and a longer sentence). We combined each with two scene types: text rendered directly within a landscape (e.g., ocean, forest) and text appearing naturally in the environment (e.g., on signs in real scenes). These six prompt patterns were generated in English and subsequently translated into Japanese and Chinese to enrich the multilingual text in images.

Scene Diversity. Finally, all synthetic prompts were designed to incorporate scene diversity, including European cityscapes, university campuses, tropical beaches, event halls, and other varied environments. The generated subjects were well-framed, resulting in images suitable for use as high-quality reference material.

B.2. Image Filtering

First, to remove images unsuitable for editing, we filtered them based on the size of the foreground objects, as in [91]. YOLOv12 [73] was applied to all images to detect objects and then performed semantic segmentation using SAM [64], conditioned on the detected bounding boxes. Images were discarded if they met any of the following criteria: the bounding box area covered less than 2% of the entire image, the segmented region within the bounding box covered less than 30%, or the CLIP similarity [47] between the YOLO-predicted class name and the bounding box region was below 20. These cases were judged to contain objects that were not clearly visible. However, images where no objects were detected were retained, assuming they are useful for background or style-level editing tasks. After this filtering, 48% of the images were retained.

We also filtered out images that were inappropriate or unsuitable as references, such as unsafe content, charts, and screenshots of system messages, using both automated screening based on Gemini and human review. For inappropriate content, we specifically targeted categories including hate, harassment, violence, self-harm, sexual content, nudity, shocking content, illegal activity, and other distressing material, ultimately excluding approximately 3% of the collected images. We also manually removed nearly identical synthetic data. An example is shown in Figure 8.

B.3. Category Classification

To construct multi-reference image editing tasks, each reference image had to be accurately categorized. For example, in portrait transformation tasks that modify a person's hairstyle or makeup, both reference images must contain a person. Similarly, in replacement tasks involving humans or objects, the corresponding entities must be present in the reference images, and in background replacement tasks, the background scenes must be identified in the reference images. To satisfy these task-dependent requirements, we performed hierarchical image classification on a large combined set of real and synthetic images.

For synthetic data, attributes can be specified directly in the prompt during generation, eliminating the need for additional classification. In contrast, real images exhibit substantial variability, requiring precise categorization to determine whether they can serve as valid reference images. To address this, we designed a structured annotation protocol

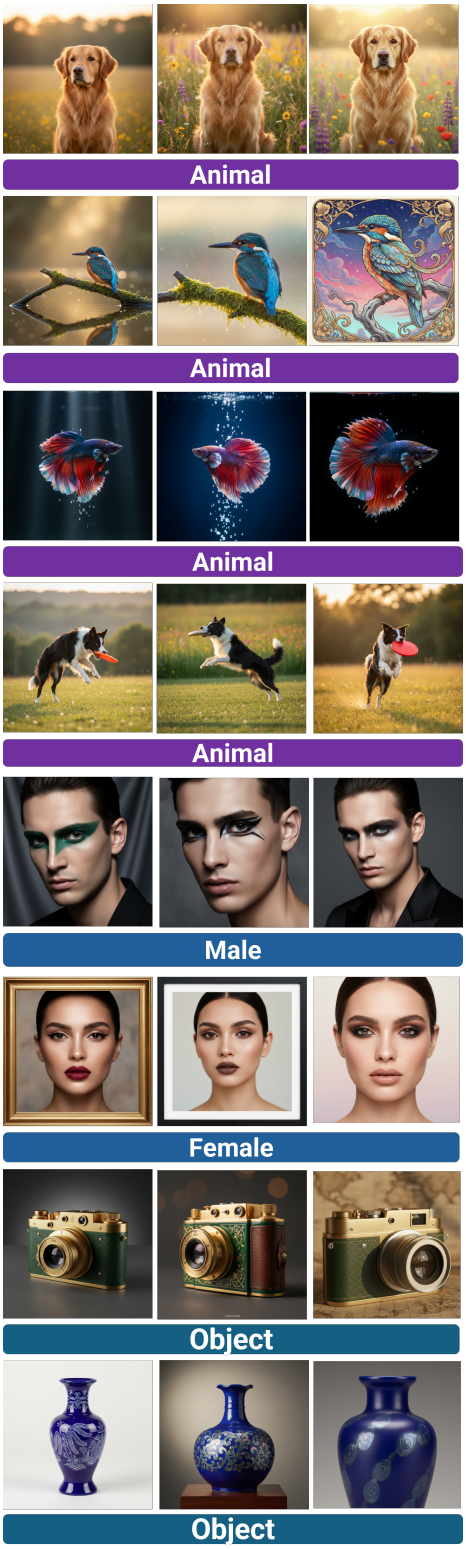

Figure 8. Examples of a duplicated synthetic image.

using Gemini that consists of three classification processes for the human, object, and background components of an image.

Human Image Classification Human classification begins with identifying how many people appear in the image. If there is precisely one person, the model additionally classifies attributes such as gender, makeup, hairstyle distinctiveness, pose clarity, and facial expression characteristics. The output follows a fixed template, enabling consistent extraction of appearance features relevant to multi-reference generation.

Object Image Classification Object image classification follows a similar structure. The model first determines whether the image contains no object, a single object, or multiple objects. When exactly one object is detected, it is further categorized according to its visual and material properties, distinguishing among animal-like forms, patterned or decorative textiles, engineered or manufactured materials, and objects lacking distinctive design features.

Background Image Classification Background classification is conducted from three viewpoints: realism, tonal uniformity, and lighting strength. Realism distinguishes photographic scenes from stylized or synthetic ones; tone indicates whether the overall color distribution is uniform or varied; and lighting strength identifies whether a clear directional key light is present.

B.4. Task Construction

The editing instructions were generated using Gemini. First, we manually selected suitable reference categories for each task, for instance, the *person* category for the hairstyle modification task. Next, we randomly sampled images from each category and presented them to Gemini to produce corresponding editing instructions. For example, in editing tasks that require object transformations, we first prompted the model to determine which reference image's object should be modified and which reference image's attributes (e.g., color) should be used as the target. Afterward, several instruction examples were provided to Gemini, and we asked Gemini to generate new instructions following the same syntax. Afterward, editing instructions that could cause visual breakdowns were removed after being scored with Gemini. Additionally, Gemini assessed the difficulty of editing instructions, and those judged easy, such as cases without domain differences, were removed. Finally, the editing instructions were manually verified not to cause image breakdowns or result in overly simple edits, and assigned the remaining samples to the appropriate difficulty categories (cross-domain, scale and viewpoint differences, rare concept, and multilingual text rendering).

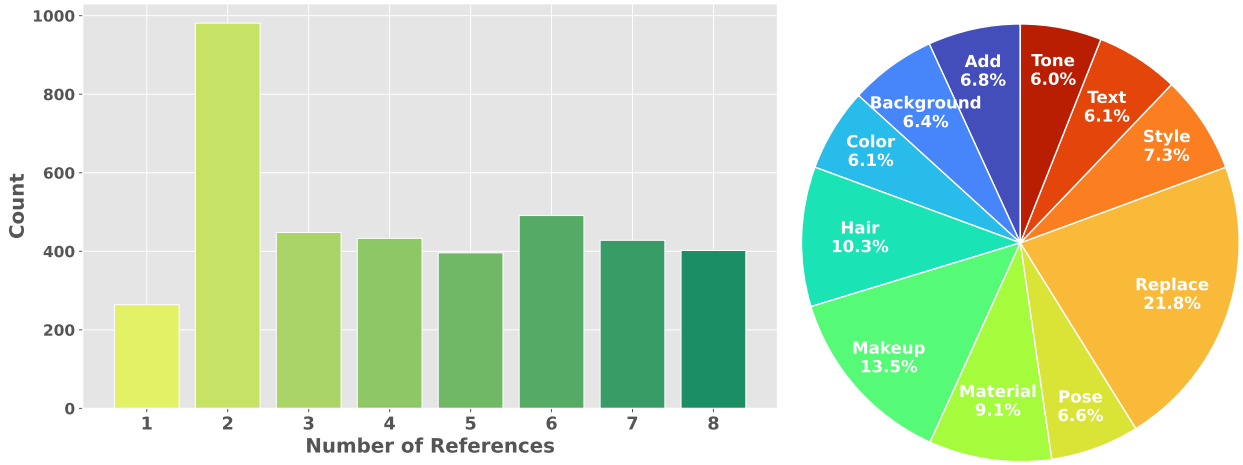

Figure 9. (**Left**) Number of tasks by reference count. The two-reference tasks contain more samples than the other reference tasks because they include multiple task types. (**Right**) Breakdown of the two-reference tasks. It contains eleven tasks.

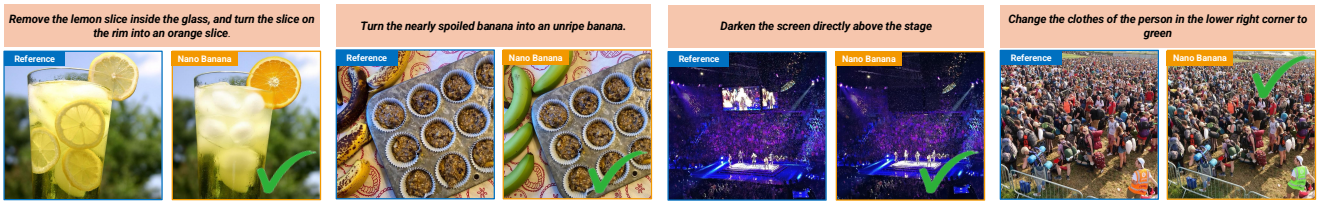

Figure 10. Example of "Hard" category of ImgEdit. These tasks are almost fully solvable by advanced models such as Nano Banana.

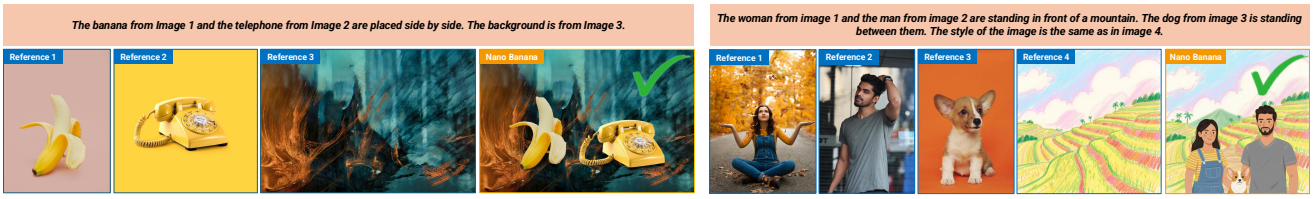

Figure 11. Example of DreamOmni2 benchmark. These tasks are almost fully solvable by advanced models such as Nano Banana.

C. Further Statistics for MultiBanana

The left of Figure 9 shows the number of tasks by each reference count. The two-reference tasks contain more samples than the other tasks because they include multiple task types. The right of Figure 9 shows the breakdown of the two-reference tasks. It contains eleven tasks considered in the prior work [83]: subject addition, subject replacement, background change, color modification, material modification, pose modification, hairstyle modification, makeup modification, tone transformation, style transfer, and text correction. Each task is guaranteed to include at least 6% of the editing samples, corresponding to roughly 60 samples, which exceeds the number of samples in Xia et al. [83].

D. Comparison with Prior Benchmarks

Table 1 shows that existing benchmarks do not provide systematic evaluation across diverse multi-reference conditions, support only a limited number of references, and fail to adequately account for heterogeneity among reference images. To illustrate that existing benchmarks are almost solved by state-of-the-art models such as Nano Banana [22] and GPT-Image-1 [56], we evaluate the image generation capabilities of these models using ImgEdit [91], the latest benchmark for image editing, and DreamOmni2 [83], the state-of-the-art benchmark for multi-reference image generation. The quantitative results show that recent closed-source, state-of-the-art models achieve consistently high scores, indicating that these benchmarks are nearing saturation and may soon be unable to meaningfully distin-

Table 7. Comparison results of different models on ImgEdit-Bench [91]. "Overall" is computed by averaging the scores across all task types. GPT-4.1 [54] is used for evaluation. The results show that recent closed-source, state-of-the-art models achieve substantially higher scores, suggesting that the benchmark may be approaching its ceiling in distinguishing high-end models. Evaluation results excluding Nano Banana are taken from the official GitHub repository of Ye et al. [91].

Model	Add	Adjust	Extract	Replace	Remove	Background	Style	Hybrid	Overall
MagicBrush [93]	2.84	1.58	1.51	1.97	1.58	1.75	2.38	1.62	1.83
AnyEdit [92]	3.18	2.95	1.88	2.47	2.23	2.24	2.85	1.56	2.45
OmniGen2 [80]	3.57	3.06	1.77	3.74	3.20	3.57	4.81	2.52	3.44
Kontext-dev [42]	3.83	3.65	2.27	4.45	3.17	3.98	4.55	3.35	3.71
Nano Banana [22]	3.63	**4.66**	**3.73**	**4.69**	**4.71**	**4.60**	4.47	**4.10**	**4.37**
GPT-Image-1 [56]	**4.61**	4.33	2.90	4.35	3.66	4.57	**4.93**	3.96	4.20

Table 8. Quantitative comparison of multimodal instruction-based editing and generation in DreamOmni2 Benchmark [83]. Both tasks are evaluated using Gemini [19] and Doubao [5]. This result shows that recent closed-source, state-of-the-art models achieve substantially higher scores, suggesting that the benchmark may be approaching its ceiling in distinguishing among high-end models. We refer to evaluation results from Xia et al. [83].

Method	Editing Task				Generation Task			
	Concrete ↑		Abstract ↓		Concrete ↑		Abstract ↓	
	Gemini	Doubao	Gemini	Doubao	Gemini	Doubao	Gemini	Doubao
Omnigen2 [80]	0.2195	0.2927	0.0427	0.0793	0.2083	0.2500	0.1000	0.0778
Kontext [42]	0.0488	0.1220	0.0183	0.0122	0.2500	0.3750	0.0556	0.1222
Qwen-Image-Edit-2509 [79]	0.2683	0.2927	0.0488	0.1159	0.1250	0.2917	0.1111	0.1556
DreamOmni2 [83]	0.5854	0.6585	0.5854	0.6280	0.5833	**0.6667**	0.5778	**0.6333**
Nano Banana [22]	**0.6829**	0.7073	0.6463	0.5488	0.5000	0.5417	0.5556	0.5488
GPT-Image-1 [56]	**0.6829**	**0.7805**	**0.7195**	**0.7439**	**0.6250**	0.6250	**0.6889**	**0.6333**

guish among high-performing models (Table 7 and Table 8). The evaluation results are taken from the official GitHub repository of ImgEdit[1] and the results reported in the DreamOmni2 paper. Qualitative results suggest that, even in the additional "Hard" category of ImgEdit, most tasks are already solvable by advanced models such as Nano Banana (Figure 10). A similar tendency is observed for DreamOmni2 benchmark as well (Figure 11).

Taken together, these findings indicate that current benchmarks for image editing and multi-reference image generation are approaching their limits in evaluating cutting-edge models. There is a clear need for more challenging benchmarks that better capture the capabilities and failure modes of the latest generation of image models.

E. Reliability and Cost of VLM Judges

As described in Section 3.4, we employ VLM judges as a time- and cost-efficient proxy for human evaluation. In Section 4.4, we demonstrate that the VLM judge correlates strongly with human ratings, supporting our usage of VLM. Here, we provide a more detailed analysis of their reliability and the associated cost considerations.

E.1. Cross-VLM Correlation

To demonstrate the consistency of our evaluation metrics, we examined correlations among different VLMs. We use the evaluated results for Nano Banana and GPT-Image-1 and examine the correlation between GPT's and Gemini's evaluation scores. To assess the reliability of the correlation coefficient, we partitioned the evaluation results into 10 subsets and computed the mean correlation across them. Table 9 shows that the evaluated scores exhibit positive correlations, which demonstrates that our evaluation criteria are well-defined and consistent.

E.2. Self-Consistency Analysis

We measure the standard error of the VLM judge scores across three different random seeds in Table 10. On the 1–10 rating scale, the observed standard errors are below 0.1, and we do not observe an increasing trend in standard error as the number of references increases, indicating

[1] https://github.com/PKU-YuanGroup/ImgEdit?tab=readme-ov-file

Table 9. Correlation between GPT's and Gemini's evaluated scores. ± shows 95% confidence interval.

Evaluation criteria	Correlation coefficients
Instruction Alignment	0.645 ± 0.021
Reference Consistency	0.549 ± 0.026
Background Subject Match	0.601 ± 0.020
Physical Realism	0.620 ± 0.022
Visual Quality	0.577 ± 0.018
Overall	0.650 ± 0.014

Table 10. Standard error of total scores evaluated by the average of GPT and Gemini, for GPT-Image-1 and Nano Banana.

Model	3-ref	4-ref	5-ref	6-ref	7-ref	8-ref
GPT-Image-1	.0234	.0138	.0139	.0153	.0343	.0244
Nano Banana	.0260	.0423	.0033	.0089	.0062	.0254

that the VLM judge provides self-consistent evaluations for multi-reference image generation.

E.3. Sensitivity Analysis

To further ensure the reliability of our evaluation metrics, we examined whether they can detect performance gains from fine-tuning on external image-editing datasets, thereby demonstrating that they align well with those used in other image-editing benchmarks. We compare Qwen-Image-Edit with Qwen-Image-Edit-2509, a fine-tuned variant designed to improve reference consistency and support multi-reference inputs. As shown in Table 11, our evaluation metrics successfully capture the performance gains of Qwen-Image-Edit-2509 over Qwen-Image-Edit. We conducted this comparison using single-reference tasks because Qwen-Image-Edit supports only a single input image.

E.4. Cost of VLM Judge

Regarding evaluation costs, a single model evaluation costs approximately $44 with GPT-5 and $29 with Gemini 2.5 in January 2026. As an API-free alternative, we confirmed that Qwen3-VL [2] can be reliable enough (see Table 4).

F. Detailed Results

F.1. Per-Task Evaluation

Figure 15, Figure 16, and Figure 17 show the scores of each model across five evaluation metrics for each task, evaluated by GPT, Gemini, and their average, respectively. Taking the weighted average across five metrics, Figure 18 shows the total score of each model for single and two-reference tasks, evaluated by GPT, Gemini, and their average, respectively. For multi-reference tasks, Table 12, Table 13, and Table 14 show the total score, evaluated by GPT, Gemini, and their average, respectively.

Table 11. Comparison of Qwen-Image-Edit and Qwen-Image-Edit-2509 on single-reference task.

	Qwen-Image-Edit	Qwen-Image-Edit-2509
single	6.332	7.499

Overall, we observed a substantial gap in Instruction Alignment and Reference Consistency between the closed-source models and the other open-source models. In particular, GPT-Image-1 achieves significantly superior performance in Instruction Alignment, especially in style and text modification tasks in the two-reference task. Meanwhile, in Reference Consistency, Nano Banana achieves the highest scores in the single-reference task. Nano Banana also performs on par with GPT-Image-1 in Reference Consistency across background modification, subject replacement, subject addition, and makeup tasks, indicating its strong ability to leverage reference images, particularly when the number of references is small. On the other hand, Nano Banana performs worse than the other models in background modification tasks in Background Subject Match, Physical Realism, and Visual Quality. This result suggests that Nano Banana struggles with background modification.

F.2. The Effect of the Number of References

According to the per-metric scores in Figure 17 and the total scores in Table 14, all models exhibit decreasing performance as the number of reference images increases. For Instruction Alignment and Reference Consistency, GPT-Image-1 and Nano Banana achieve relatively high scores, whereas the open-source model, DreamOmni2, tends to approach the minimum score of 1 as the number of references increases. In contrast, for quality-related metrics, Background Subject Match, Physical Realism, and Visual Quality, the DreamOmni2 maintains high scores even when more reference images are provided. As discussed in Section 4, this is because closed-source models prioritize adherence to the references and editing instructions over visual quality, while open-source models show a tendency to preserve visual quality but often ignore the references.

In the finer-grained task-level evaluation, we found that, even as the number of reference images increased, the X–1 Object + Global task did not show severe decreases in Background Subject Match, Physical Realism, or Visual Quality scores. The task requires modifying the image to a specified style; therefore, even if the reference images come from different domains, which may degrade the visual quality, the final output often converges to a unified style.

In the results of Qwen-Image-Edit-2509, the model may be trained to handle up to three reference images; therefore,

Table 12. Detailed per-task total scores for the multi-reference tasks, evaluated by GPT. The local, global, back, and object columns under X correspond to X–1 Objects + Local, X–1 Objects + Global, X–1 Objects + Background, and X Object, respectively.

Model	**3-references**				**4-references**				**5-references**			
	local	global	back	object	local	global	back	object	local	global	back	object
DreamOmni2	3.94	4.19	3.92	3.91	3.34	3.24	2.87	3.20	3.30	3.06	2.73	3.22
OmniGen2	4.42	4.59	4.47	4.47	4.41	3.37	3.10	3.27	3.78	3.28	2.87	3.31
Qwen-Image-Edit-2509	4.42	4.65	4.75	4.77	3.08	3.35	2.78	3.54	1.93	2.40	1.98	2.10
Nano Banana	5.44	5.81	5.10	5.64	4.57	5.36	4.27	5.25	5.12	5.93	3.96	4.93
GPT-image-1	6.58	7.40	6.98	6.40	6.18	6.98	6.73	6.56	6.18	6.66	6.22	6.28
Model	**6-references**				**7-references**				**8-references**			
	local	global	back	object	local	global	back	object	local	global	back	object
DreamOmni2	3.37	3.03	2.82	2.96	3.24	3.02	2.71	2.86	3.26	2.79	2.57	2.82
OmniGen2	-	-	-	-	-	-	-	-	-	-	-	-
Qwen-Image-Edit-2509	1.85	2.18	1.68	1.79	1.51	1.89	1.57	1.88	1.68	2.29	1.62	1.83
Nano Banana	4.43	4.88	3.89	4.90	4.40	4.78	3.57	4.81	4.31	4.82	3.21	4.45
GPT-image-1	6.16	6.08	5.68	5.90	5.38	6.09	5.59	5.94	4.96	5.82	5.03	5.15

Table 13. Detailed per-task total scores for the multi-reference tasks, evaluated by Gemini. The local, global, back, and object columns under X correspond to X-1 Objects + Local, X–1 Objects + Global, X–1 Objects + Background, and X Object, respectively.

Model	**3-references**				**4-references**				**5-references**			
	local	global	back	object	local	global	back	object	local	global	back	object
DreamOmni2	3.00	2.49	2.45	2.74	2.84	2.73	2.38	2.44	2.63	2.50	2.36	2.66
OmniGen2	3.03	2.89	3.00	3.25	3.10	3.06	2.27	2.73	2.84	3.02	2.41	2.50
Qwen-Image-Edit-2509	3.27	3.16	3.29	3.97	2.29	3.00	2.15	2.79	1.14	1.69	1.21	1.20
Nano Banana	4.34	4.69	4.61	4.89	3.42	4.34	3.79	4.14	4.25	4.50	3.16	3.85
GPT-image-1	5.48	6.28	5.37	5.16	4.83	5.26	4.75	5.04	4.89	5.20	4.25	4.10
Model	**6-references**				**7-references**				**8-references**			
	local	global	back	object	local	global	back	object	local	global	back	object
DreamOmni2	2.47	2.54	2.14	2.36	2.65	2.50	2.15	2.30	2.42	2.33	2.04	2.19
OmniGen2	-	-	-	-	-	-	-	-	-	-	-	-
Qwen-Image-Edit-2509	1.00	1.26	1.00	1.03	1.01	1.06	1.33	1.17	1.18	1.31	1.01	1.00
Nano Banana	3.25	3.91	2.55	3.79	3.05	3.73	2.26	3.52	2.85	3.64	2.53	3.25
GPT-image-1	3.66	4.66	3.46	4.02	4.18	4.58	3.24	3.52	3.28	4.07	2.91	2.96

when provided with four or more references, the input becomes out-of-distribution. This causes the model to produce perceptually invalid outputs, and then the scores approach the minimum value of 1 (see Figure 29).

F.3. Difficult Reference Combination

Figure 19 shows the scores of each model in difficult reference combination tasks across evaluation metrics.

Cross-domain diversity In cross-domain cases, samples with reference images from different domains (darker color) received lower Reference Consistency scores because all models tend to distort individual reference attributes to achieve a consistent style.

Scale and viewpoint differences In the different scale and viewpoint cases, the Reference Consistency scores declined consistently across all models. This suggests that when models attempt to reproduce objects at different scales or from different viewpoints, they may fail to preserve fine details or adjust the pose to achieve a more natural appearance.

Rare concept references In the rare-concept case, the model may struggle to handle uncommon subjects relative to more familiar ones. Because the reference images often depict these subjects at a large scale, the model tends to reproduce this scale without appropriate adjustment. This suggests that, unlike common concepts, the model may find it more difficult to flexibly control attributes such as size

Table 14. Detailed per-task total scores for each multi-reference task, evaluated by the average of GPT and Gemini. The local, global, back, and object columns under X correspond to X–1 Objects + Local, X–1 Objects + Global, X–1 Objects + Background, and X Object, respectively.

Model	3-references				4-references				5-references			
	local	global	back	object	local	global	back	object	local	global	back	object
DreamOmni2	3.47	3.34	3.18	3.33	3.09	2.98	2.62	2.82	2.97	2.78	2.54	2.94
OmniGen2	3.73	3.74	3.74	3.86	3.76	3.22	2.69	3.00	3.31	3.15	2.64	2.91
Qwen-Image-Edit-2509	3.85	3.90	4.02	4.37	2.69	3.17	2.47	3.17	1.54	2.04	1.60	1.65
Nano Banana	4.89	5.25	4.85	5.27	4.00	4.85	4.03	4.70	4.69	5.22	3.56	4.39
GPT-image-1	6.03	6.84	6.18	5.78	5.50	6.12	5.74	5.80	5.54	5.93	5.24	5.19
Model	**6-references**				**7-references**				**8-references**			
	local	global	back	object	local	global	back	object	local	global	back	object
DreamOmni2	2.92	2.78	2.48	2.66	2.94	2.76	2.43	2.58	2.84	2.56	2.31	2.50
OmniGen2	-	-	-	-	-	-	-	-	-	-	-	-
Qwen-Image-Edit-2509	1.42	1.72	1.34	1.41	1.26	1.47	1.45	1.52	1.43	1.80	1.32	1.42
Nano Banana	3.84	4.39	3.22	4.35	3.72	4.25	2.92	4.16	3.58	4.23	2.87	3.85
GPT-image-1	4.91	5.37	4.57	4.96	4.78	5.33	4.42	4.73	4.12	4.94	3.97	4.05

when dealing with rare concepts. As a result, rare concept samples tend to receive lower Physical Realism scores.

Multilingual references In the multilingual text rendering case, we observed a consistent decline across all metrics except Reference Consistency. This may occur because models have limited text-rendering capability in languages other than English, leading to failures in cross-language conversion. As a result, instruction alignment and visual quality decrease. However, GPT-Image-1 exhibits a smaller decline than other models, owing to its comparatively stronger text-rendering ability in non-English languages.

F.4. Risks of Model Bias in Synthetic References

Synthetic references generated by closed models (i.e., Nano Banana, GPT-Image-1) may introduce in-distribution advantages when those same models are also evaluated on the benchmark. To verify this, we divided the benchmark into three subsets based on the source of the reference images: those from GPT-Image-1, those from Nano Banana, and those from both. The mean scores across all subsets remain within their respective 99% confidence intervals, indicating no statistically significant bias (Figure 12).

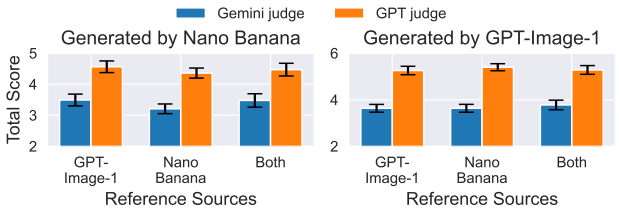

Figure 12. Analysis of potential in-distribution bias. Error bars indicate the 99% confidence intervals. For image generation models, we use the versions available as of January 2026.

Table 15. Comparison of the mean scores when prioritizing either reference consistency or background–subject match for the subset generated by Nano Banana, latest version as of January 2026.

Prompt	Reference Consistency	Background–Subject Match
Original	3.58	3.67
Consistency priority	4.17	3.25
Matching priority	3.40	4.03

F.5. Potential Conflict in Cross-Domain Task

One might expect that preserving subject details (reference consistency) and maintaining a coherent scene (background-subject match) are inherently at odds in cross-domain tasks—e.g., placing a photorealistic person into an anime background. However, our evaluation shows this is not always the case. As shown in Figure 13 (left), a generated image can satisfy both criteria, achieving a GPT-5 score of 7 for reference consistency and 9 for background-subject match. To further verify this, we created cross-domain editing tasks with explicit instructions that prioritize one criterion over the other for 20 randomly selected prompts. Table 15 shows that Nano Banana can improve the prioritized criterion while maintaining the other criterion.

F.6. Alignment with Real-World Use Cases

As shown in Figure 13 (right), MultiBanana is directly relevant to real-world use cases, such as advertising. Our

Figure 13. **Left**: A sample generated by GPT-Image-1 on a cross-domain task. **Right**: Task example of our benchmark directly relevant to real-world use cases, such as advertising.

benchmark development using synthetic data also aligns with the community standard. For example, ImgEdit [91] similarly employs GPT-4o for instruction generation, which also aims to develop "practical and powerful tools for real-world applications."

F.7. Agentic Inference

We introduced three agentic frameworks: Iterative Prompt Refinement (IPR), Context-Aware Feedback Generation (CAFG), and Selective Reference Adaptation (SRA). Here, we refer to Gen as the *Generator*, which produces an image, and Plan as the *Planner*, which updates the instruction prompt (and selects reference images) for the next step. Let P^t denote the instruction prompt at step t, $\{R_i\}_{i\in[I]}$ the reference images, and G^t the generated image with $G^0 = \varnothing$. We conducted experiments with both GPT and Gemini. For GPT, we use GPT-Image-1 as the *Generator* and GPT-5 as the *Planner*. For Gemini, we use Nano Banana as the *Generator* and Gemini 2.5 Flash as the *Planner*.

F.7.1. Iterative Prompt Refinement (IPR)

The IPR framework is formulated as follows. The *Planner* refines the prompt based on the generation result from the previous step.

$$G^{t+1} = \mathrm{Gen}(P^t, \{R_i\}_{i\in[I]}, \varnothing),$$
$$P^{t+1} = \mathrm{Plan}(P^t, \{R_i\}_{i\in[I]}, G^{t+1}).$$

F.7.2. Context-Aware Feedback Generation (CAFG)

The CAFG framework is formulated as follows. The *Generator* generates the image based on the generation result (context) from the previous step.

$$G^{t+1} = \mathrm{Gen}(P^t, \{R_i\}_{i\in[I]}, \boldsymbol{G^t}),$$
$$P^{t+1} = \mathrm{Plan}(P^t, \{R_i\}_{i\in[I]}, G^{t+1}).$$

F.7.3. Selective Reference Adaptation (SRA)

The SRA framework is formulated as follows. The *Planner* selects only the reference images that should be improved based on the generation result from the previous step, and the *Generator* receives these as context. SRA is expected to reduce agents' task complexity by adaptively decreasing the number of references.

$$G^{t+1} = \mathrm{Gen}(P^t, \{R_i\}_{\boldsymbol{i\in U^t}}, G^t),$$
$$P^{t+1}, \boldsymbol{U^{t+1}} = \mathrm{Plan}(P^t, \{R_i\}_{i\in[I]}, G^{t+1}).$$

where U^t is the index set of reference images that are insufficiently reflected in the generated image G^t.

F.7.4. Experiments

We set the maximum number of steps t to 3 and evaluate three agentic frameworks—IPR, CAFG, and SRA—using Gemini, and the IPR framework using GPT. Figure 14 illustrates an example of step-wise improvement in the IPR framework. Figure 20, Figure 21, Figure 22, and Figure 23 present the evaluation results of multi-reference generations at each step, averaged across judgments from Gemini 2.5 Flash and GPT-5, broken down by category. Figure 24 presents the results for single and two-reference generations. Nano Banana (Gemini) shows modest improvements in Physical Realism and Visual Quality across refinement steps, while Instruction Alignment and Reference Consistency either remain unchanged or deteriorate. In contrast, GPT demonstrates consistent improvements across all categories. This suggests that Gemini's planner progressively loses information from the original prompt as refinement steps proceed.

G. Implementations

Code: `https://github.com/matsuolab/multibanana`.

Dataset: `https://huggingface.co/datasets/kohsei/MultiBanana-Benchmark`.

API endpoints: For VLM evaluation, we used stable API versions: `gpt-5-2025-08-07` and `gemini-2.5-flash`. For image generation models, we use the latest available API versions as of November 2025, unless otherwise noted.

Figure 14. Example of changes in the Iterative Prompt Refinement (IPR) framework. Comparison and evaluation results between Nano Banana and GPT on the task of generating images from 8 object references according to the instruction prompt.

H. Extended Related Works

H.1. Controllable Text-to-Image Generation

Stable Diffusion [15, 60, 65], FLUX [41, 42], DALL-E [63], and Imagen [32, 67], have demonstrated strong text-to-image generation capabilities, establishing a scalable foundation for the task. To enhance controllability, models such as ControlNet [94] and T2I-Adapter [52] introduced external conditioning modules, enabling image-conditioned generation. Additionally, training-free approaches (e.g., pix2pix-zero [59] and prompt-to-prompt [25]) have also been widely proposed [6, 31, 39, 49, 74, 75]. These advancements collectively demonstrate that diffusion models are becoming increasingly flexible and adaptable across diverse conditional generation settings.

H.2. Benchmarks for Reference-Based Generation

Reference-based image generation is closely related to industrial applications such as content production [37, 66, 78, 88, 90, 96], advertising [28, 33, 51], and fashion design [11, 12, 16, 16, 36, 86, 98]. Among reference-driven image generation tasks, the most widely recognized one is image editing. RealEdit [71] focuses on single-image editing and provides an empirical analysis of real-world image-editing use cases. SpotEdit [20] proposes a benchmark for visually guided image editing. These works focus on benchmarking single-reference image edition/generation, while we focus on multi-reference image generation. In recent years, increasing attention has been directed toward multi-reference image generation, in which multiple reference images are jointly used to generate a single image. MultiRef-Bench [9] studies controllable image generation with multiple visual references, but our benchmark outperforms it by providing 3,769 samples (compared to 1,990), supporting up to 8 references (compared to 6), and using raw images (compared to bounding boxes or masks), making it more practical.

H.3. Instruction-Following in LLMs

The task of generating an image that faithfully reflects multiple references is analogous to instruction following in LLMs [14, 46, 76, 95, 97], in which models must simultaneously satisfy multiple constraints. In the text domain, recent studies have revealed that LLMs struggle to follow all given instructions as the number of constraints grows, often ignoring or inadequately addressing a subset of them [23, 24, 35, 40]. Multi-reference image generation poses a parallel challenge: as the number of reference images increases, models must reconcile potentially conflicting visual cues while remaining faithful to every reference.

I. Limitations and Discussions

Our benchmark focuses on static image generation and does not address video generation or editing [21, 27, 55], where temporal consistency across frames introduces additional challenges for reference-based conditioning [1, 10, 29]. Moreover, our prompts describe object placement in natural language (e.g., ”on the left,” ”in the foreground”), which is inherently ambiguous. Combining reference images with structured layout inputs [45, 61, 89, 94] could enable finer-grained control over subject composition.

J. Prompts

J.1. Prompts of Agentic Framework

Prompt of the *Generator* in the IPR framework

{prompt}{reference_image_files}

Prompt of the *Planner* in the IPR framework

You are a prompt-refiner agent.

Previous prompt: {previous_prompt}

You are given multiple reference images and one generated image.

The generated image is the last one, and the reference images come before it.

Based on the reference images and the generated image (the last image), propose a refined prompt that improves how reference images are used and adjusts the composition/style.

Return only the new prompt text.

{reference_image_files}{previously_generated_image_file}

Prompt of the *Generator* in the CAFG framework

{prompt}

The last image provided is the previously generated image from the last step and all images before it are reference images.

Please refine the previous generation using the reference images and enhance composition/style.

{reference_image_files}{previously_generated_image_file}

Prompt of the *Planner* in the CAFG framework

You are a prompt-refiner agent.

Previous prompt: {previous_prompt}

You are given multiple reference images and one generated image.

The generated image is the last one, and the reference images come before it.

Based on the reference images and the generated image (the last image), propose a refined prompt that improves how reference images are used and adjusts the composition/style.

Return only the new prompt text.

{reference_image_files}{previously_generated_image_file}

Prompt of the *Generator* in the SRA framework

{prompt}

The last image provided is the previously generated image from the last step and all images before it are reference images.

Please refine the previous generation using the reference images and enhance composition/style.

{reference_image_files}{previously_generated_image_file}

Prompt of the *Planner* in the SRA framework

You are a prompt-refiner and reference-selector agent.

Previous prompt: {previous_prompt}

You are given multiple reference images (in order) and one generated image (the last one).

Your task:

1. Identify which reference images are insufficiently reflected in the generated image.

2. Propose a refined prompt that improves the generation.

3. Return your response in JSON format ONLY. Do not include any other text.

JSON format:

{

”indices”: [0, 2, 3],

”prompt”: ”Your refined prompt text here”

}

Where ’indices’ is an array of 0-based indices (from 0 to {len(reference_image_files)-1}) of reference images

that are insufficiently reflected in the generated image.

’prompt’ is the refined instruction prompt for the next generation step.

Example response:

{

”indices”: [0, 2, 3],

”prompt”: ”Focus more on the lighting from the first image and the composition from images 3 and 4. Emphasize the color palette and texture details.”

}

{reference_image_files}{previously_generated_image_file}

J.2. Prompts of VLM as Judge

Multi-Reference Image Genearation Evaluation

You are a strict data rater specializing in grading multi-reference driven image generation. You will be given reference images, a task instruction, and the generation results.

Reference Images: {reference_image_files}

Editing Instruction: {instruction}

Final Output: {generated_image_files}

Your task is to evaluate the effectiveness of replacement editing from five independent perspectives, each on a 10-point scale. Note that the average score should be considered 4 points.

1. Text-Instruction Alignment

Evaluate whether the generated image accurately follows the given text instruction. Check whether the specified objects appear in the correct positions, whether the instructed subjects are depicted properly, and whether no unintended elements are introduced. For example, if the instruction says "change the language," but the actual written content itself is altered incorrectly, or if unnecessary objects are added, the score should be reduced. If the instruction requires including a reference subject but the generated image fails to include that referenced content, the score must be 1. Even if the instruction is followed correctly, the score must not exceed 6 points if the generated image still exhibits any composited or unnatural appearance.

2. Reference Consistency

Evaluate how consistent the generated image is with the provided reference images. Compare the output to each reference and assess how faithfully the structure and attributes are reproduced. Fine details, such as hair ornaments, patterns on clothing, and other small features, must match the references; otherwise the score must not exceed 4 points. If even a single object fails to follow the details of the reference images, the score must not exceed 6 points.

3. Background-Subject Match

Evaluate whether the subject blends naturally with the background. Check whether the subject appears to be floating, unnaturally pasted on, or visually inconsistent with its surroundings. Images that look like multiple pictures simply pasted together should receive a score of 1. If there is even the slightest inconsistency in style, tone, lighting, or overall visual impression compared to the reference images, the score must also not exceed 4 points.

4. Physical Realism

Evaluate whether the generated image maintains physical plausibility. Penalize cases where the image violates basic physical laws—for example, a person floating in mid-air, standing on water, or having the lower body missing despite no obstruction. If there is even a slight impression that the image looks composited or artificially pasted together, the score must not exceed 4 points. Likewise, if it is unclear whether the subject is actually making proper contact with the ground, the score must also not exceed 6 points.

5. Visual Quality

Evaluate the overall perceptual quality of the image. Assess whether the image is visually appealing and aesthetically coherent. If the composition appears unnatural or the image does not look aesthetically pleasing to a human observer, the score must not exceed 4 points.

Each of the five scores must be evaluated independently. Do not force any score to be tied to or capped by another score.

First, explain the reasoning, then present the final assessment.

Start the reasoning with Reasoning: .

After explaining the reasoning, present the final assessment in the format:

Instruction Alignment: ⟨A number from 1 to 10⟩.

Reference Consistency: ⟨A number from 1 to 10⟩.

Background-Subject Match: ⟨A number from 1 to 10⟩.

Physical Realism: ⟨A number from 1 to 10⟩.

Visual Quality: ⟨A number from 1 to 10⟩.

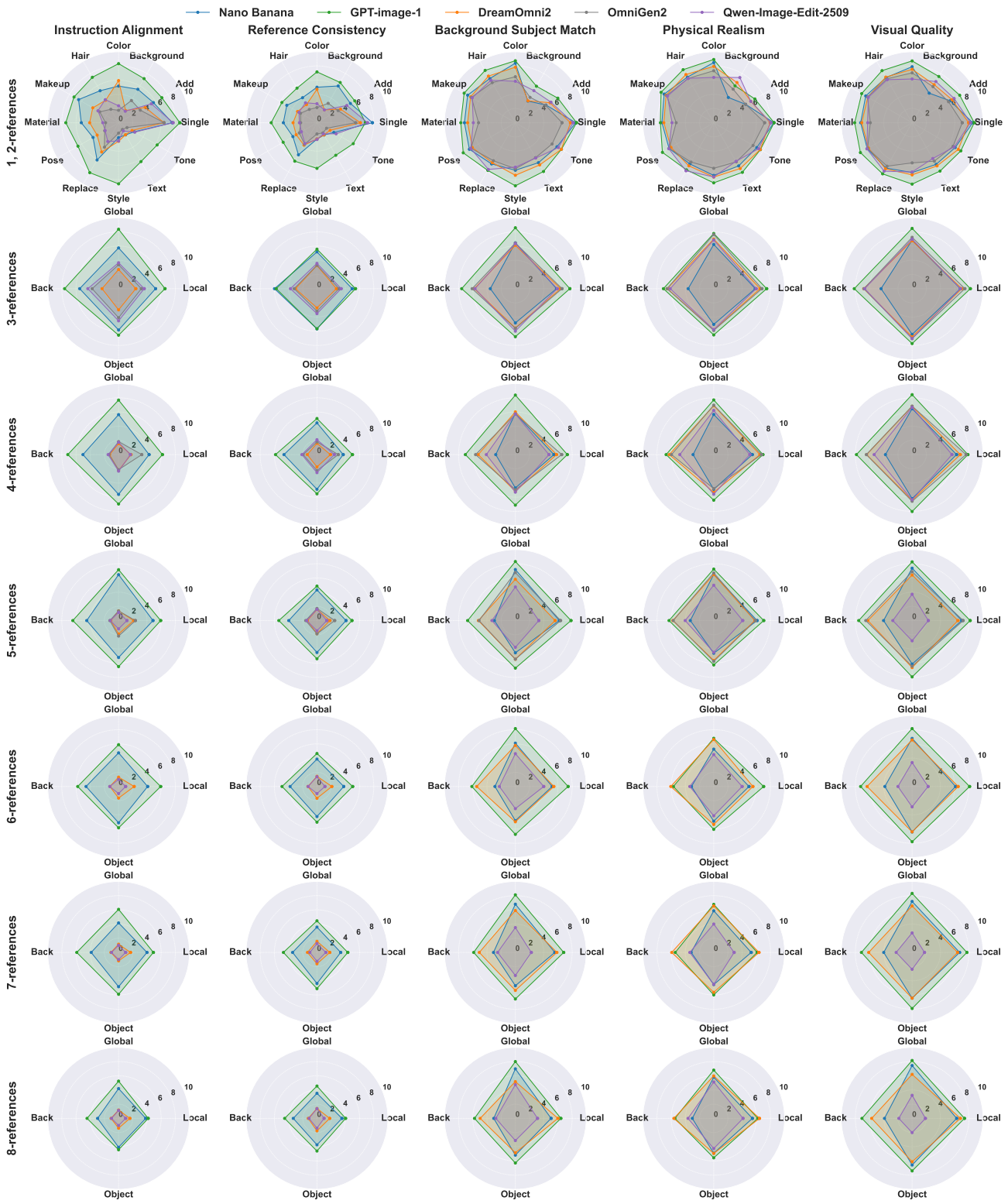

Figure 15. Scores of each model across evaluation metrics for each task by GPT. The horizontal axis denotes the scores for the five evaluation criteria, and the vertical axis denotes the number of reference images.

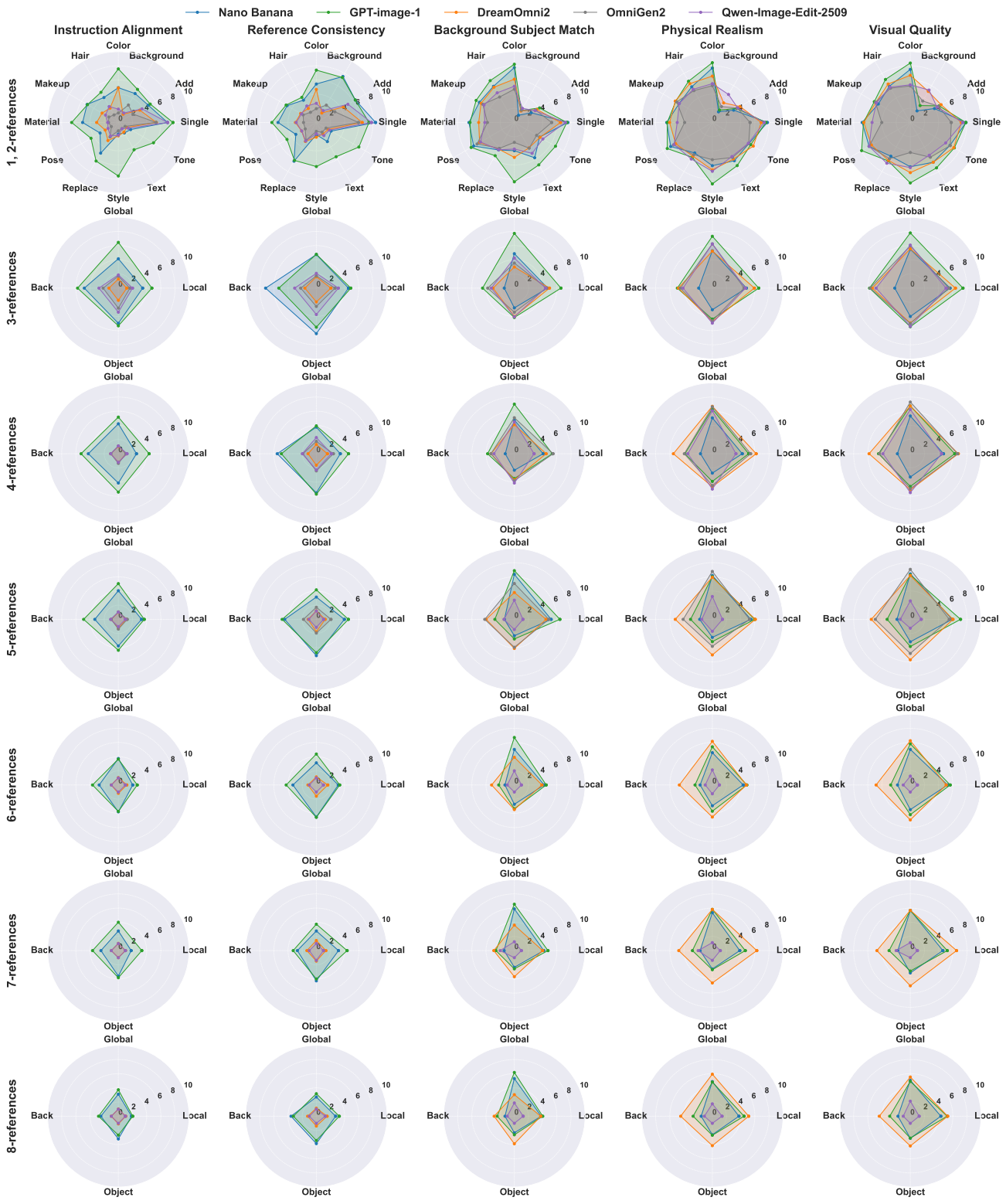

Figure 16. Scores of each model across evaluation metrics for each task by Gemini. The horizontal axis denotes the scores for the five evaluation criteria, and the vertical axis denotes the number of reference images.

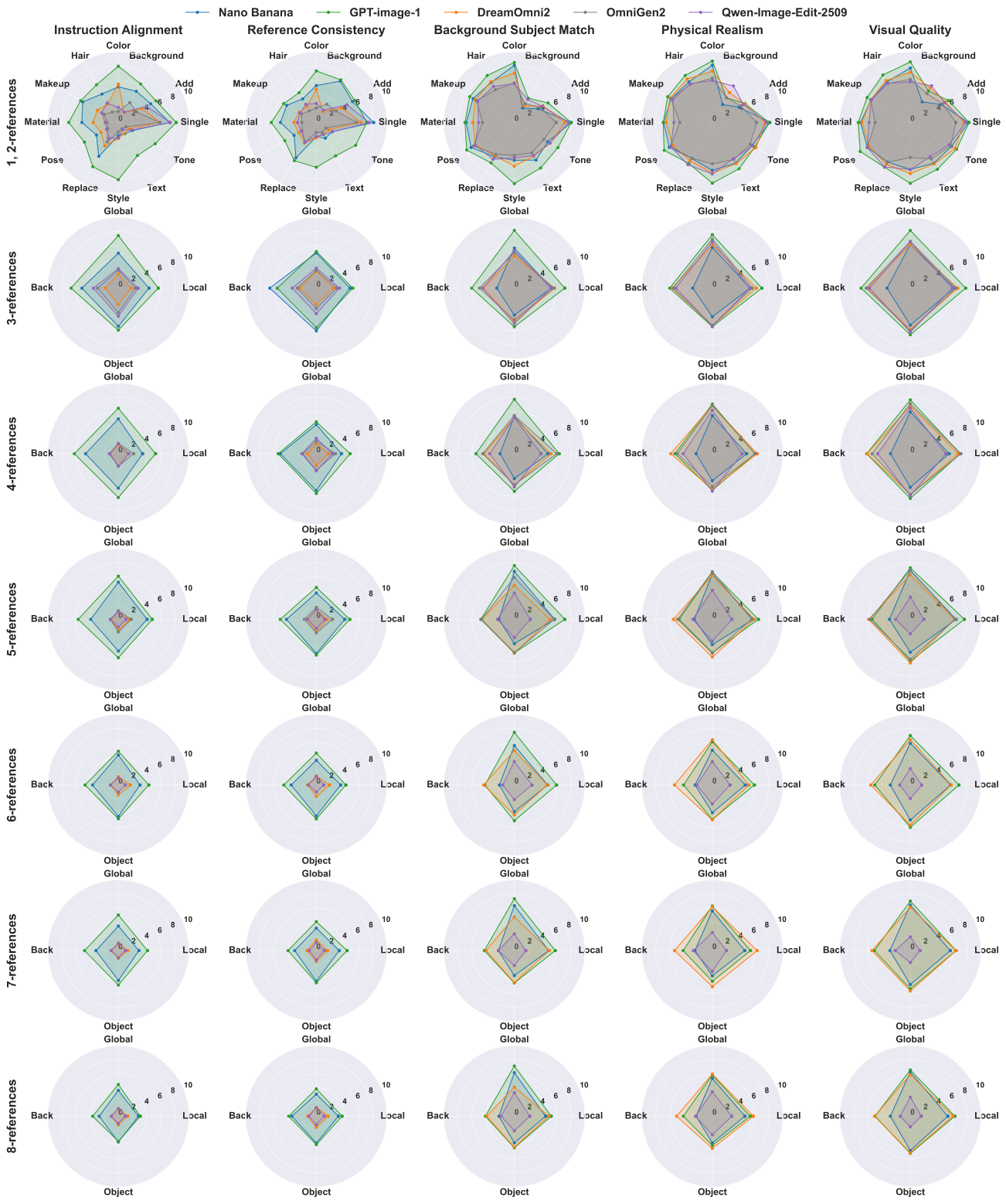

Figure 17. Average scores of each model across evaluation metrics for each task by the average of GPT and Gemini. The horizontal axis denotes the scores for the five evaluation criteria, and the vertical axis denotes the number of reference images.

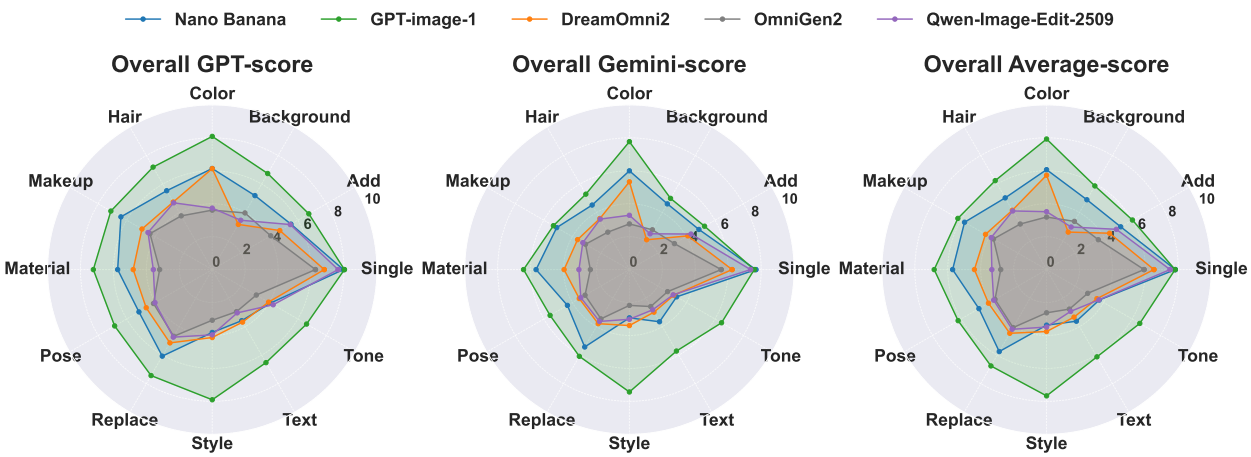

Figure 18. Total scores of each model for single and two-reference tasks by GPT, Gemini, and their average.

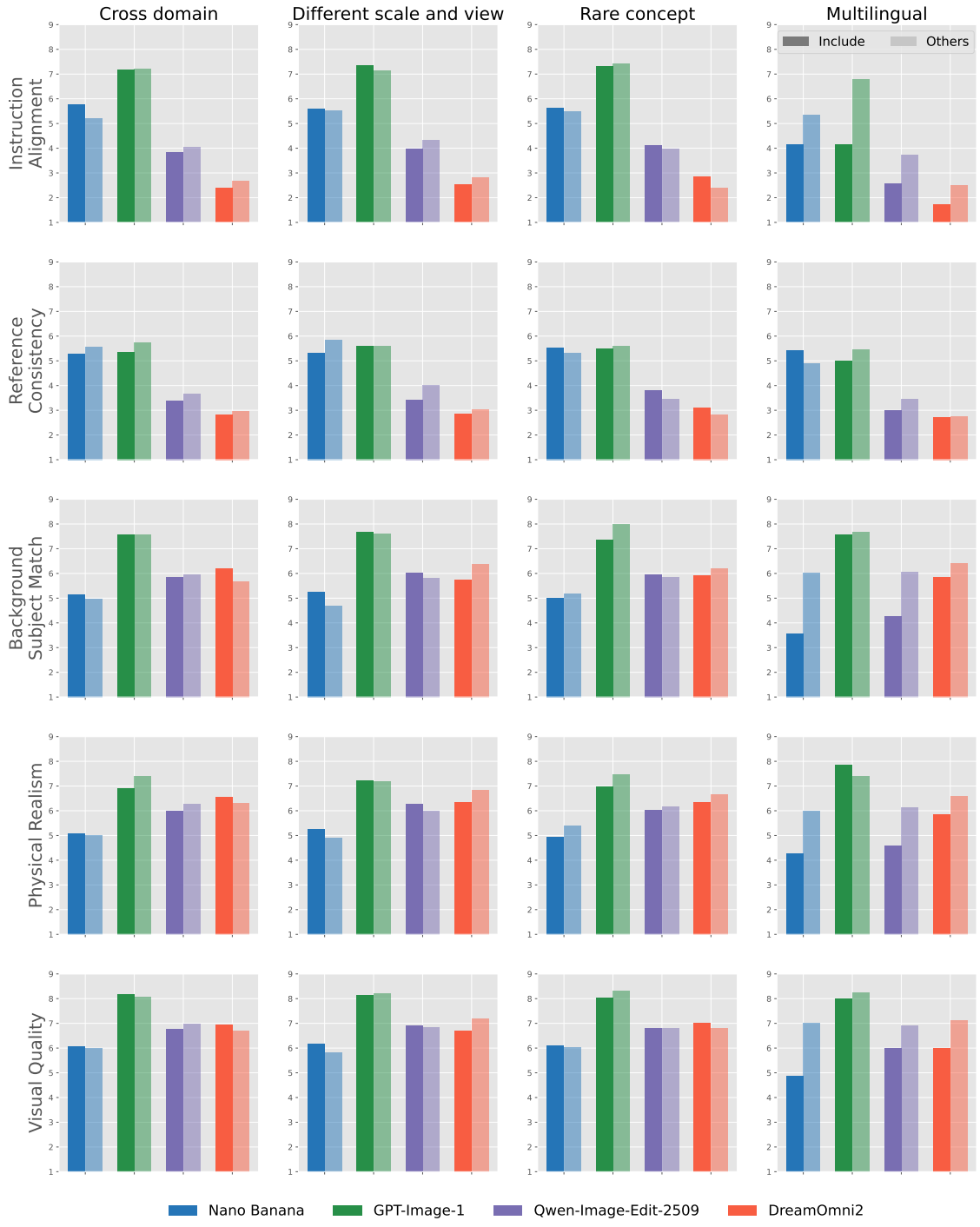

Figure 19. Total scores for each difficult reference combination across models. Darker colors represent the average score for tasks that include the corresponding combination, whereas lighter colors indicate the average score for tasks that do not include it.

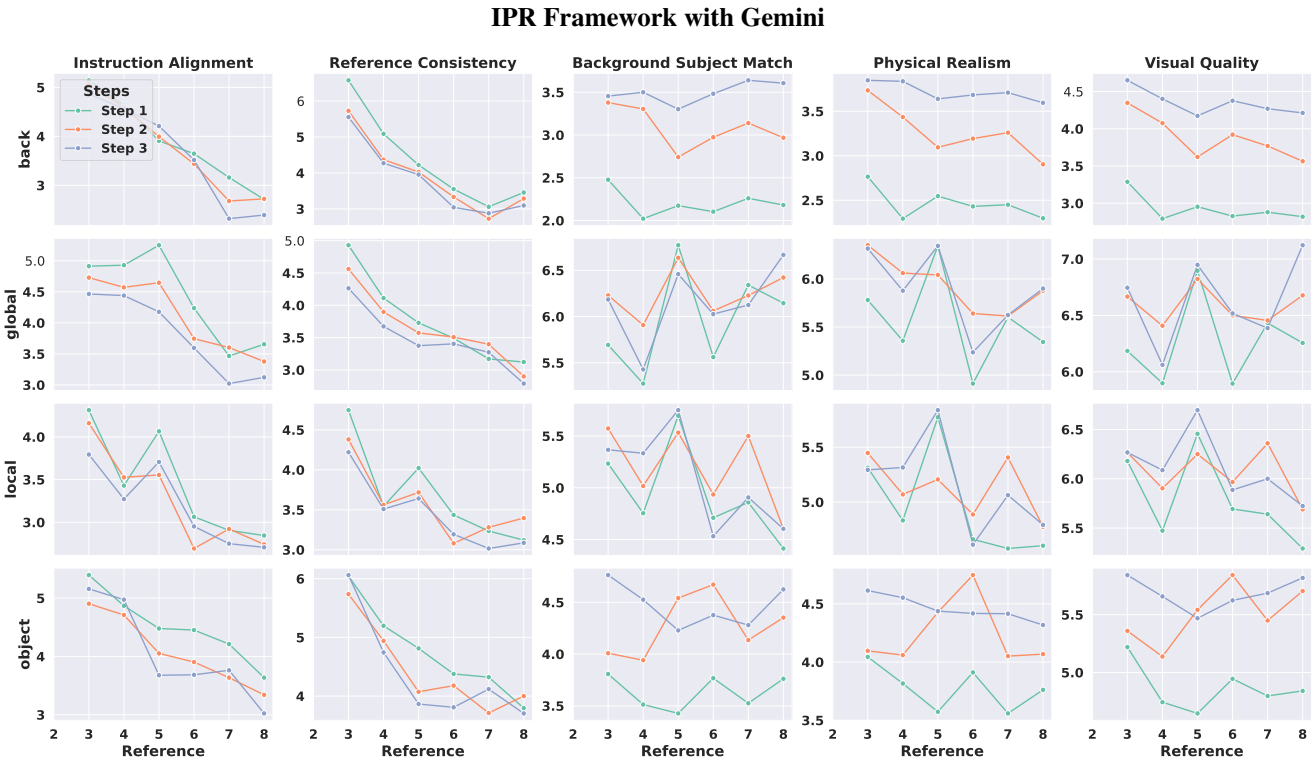

Figure 20. Detailed results of multi-reference image generation using the IPR framework with Gemini.

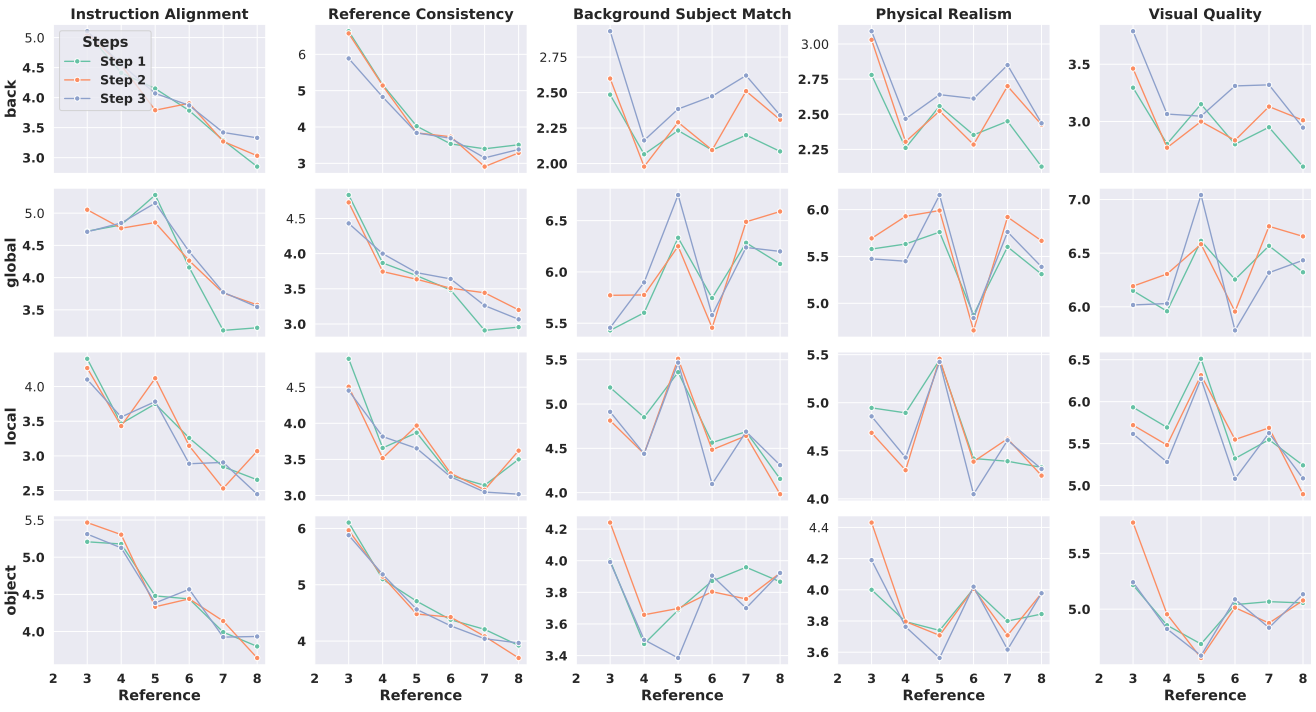

Figure 21. Detailed results of multi-reference image generation using the CAFG framework with Gemini.

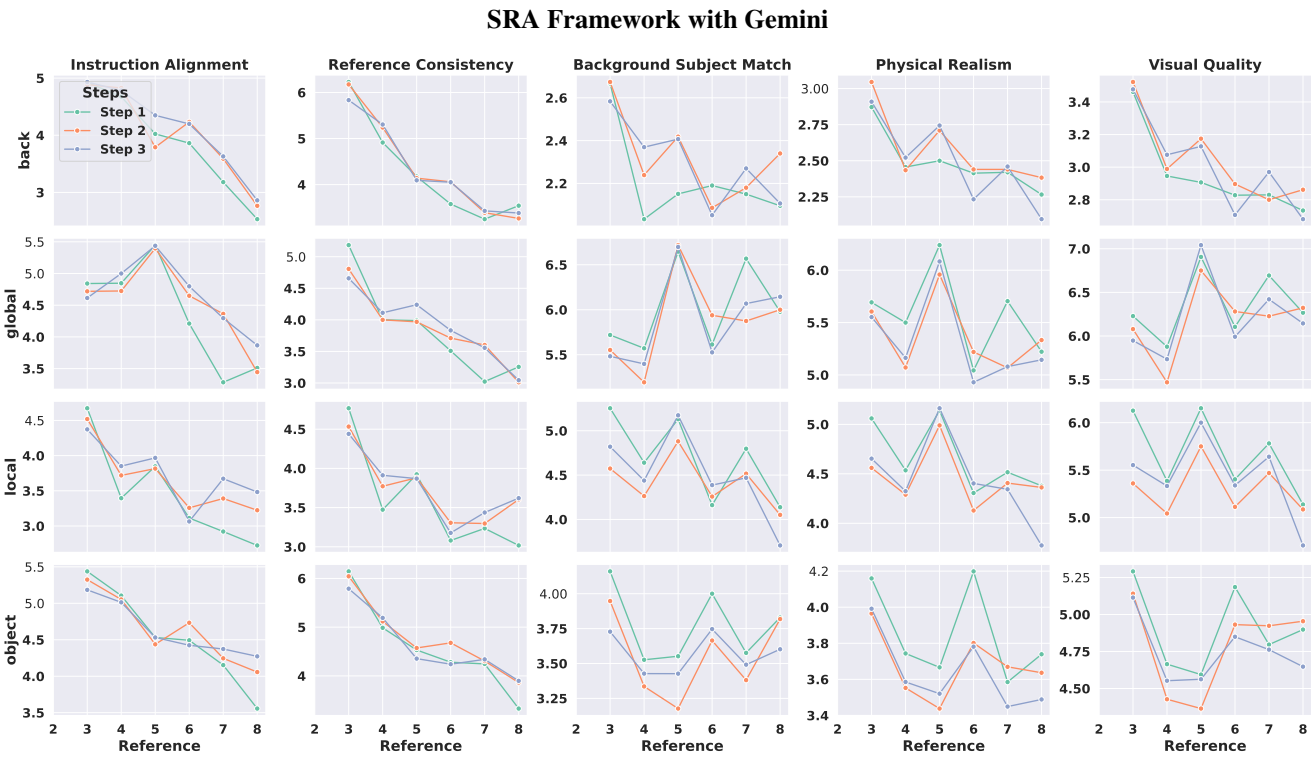

Figure 22. Detailed results of multi-reference image generation using the SRA framework with Gemini.

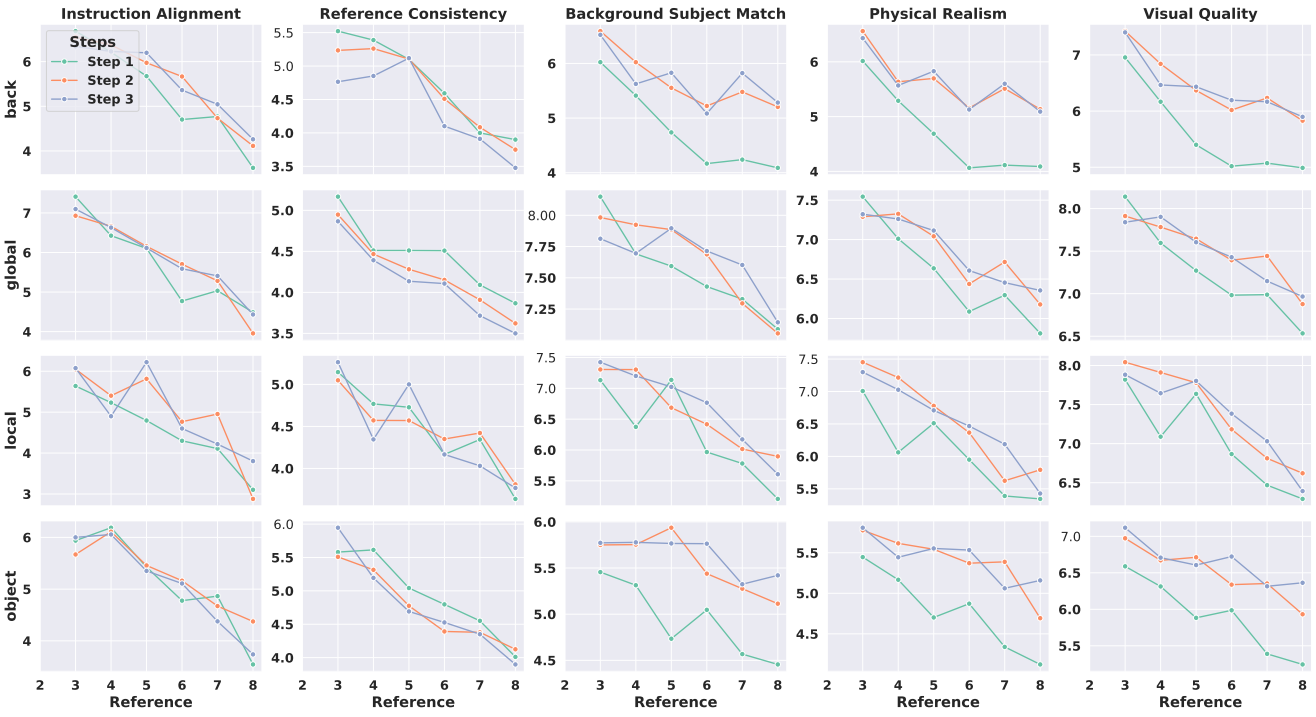

Figure 23. Detailed results of multi-reference image generation using the IPR framework with GPT.

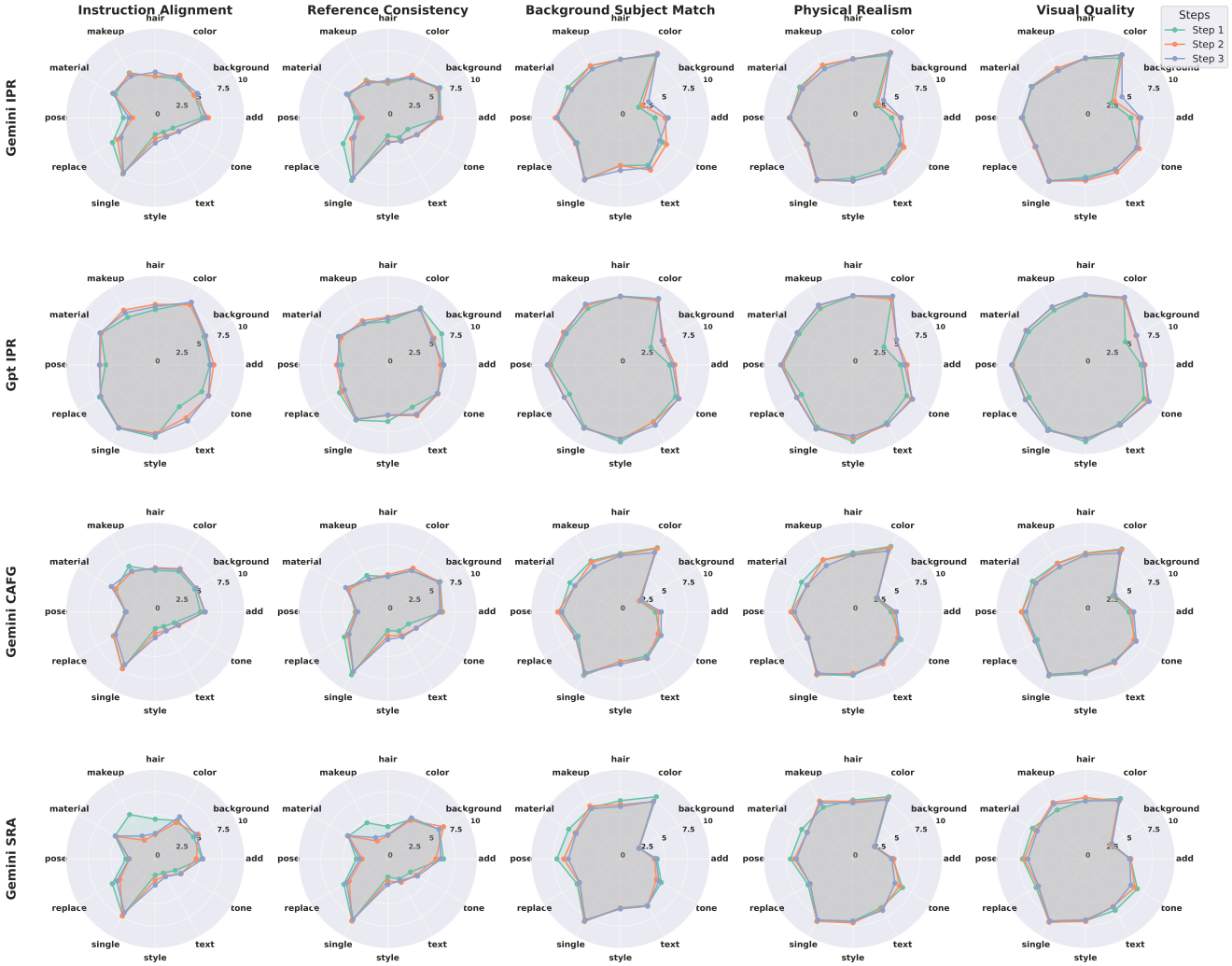

Figure 24. Detailed results of single and two-reference image generation using the agentic frameworks.

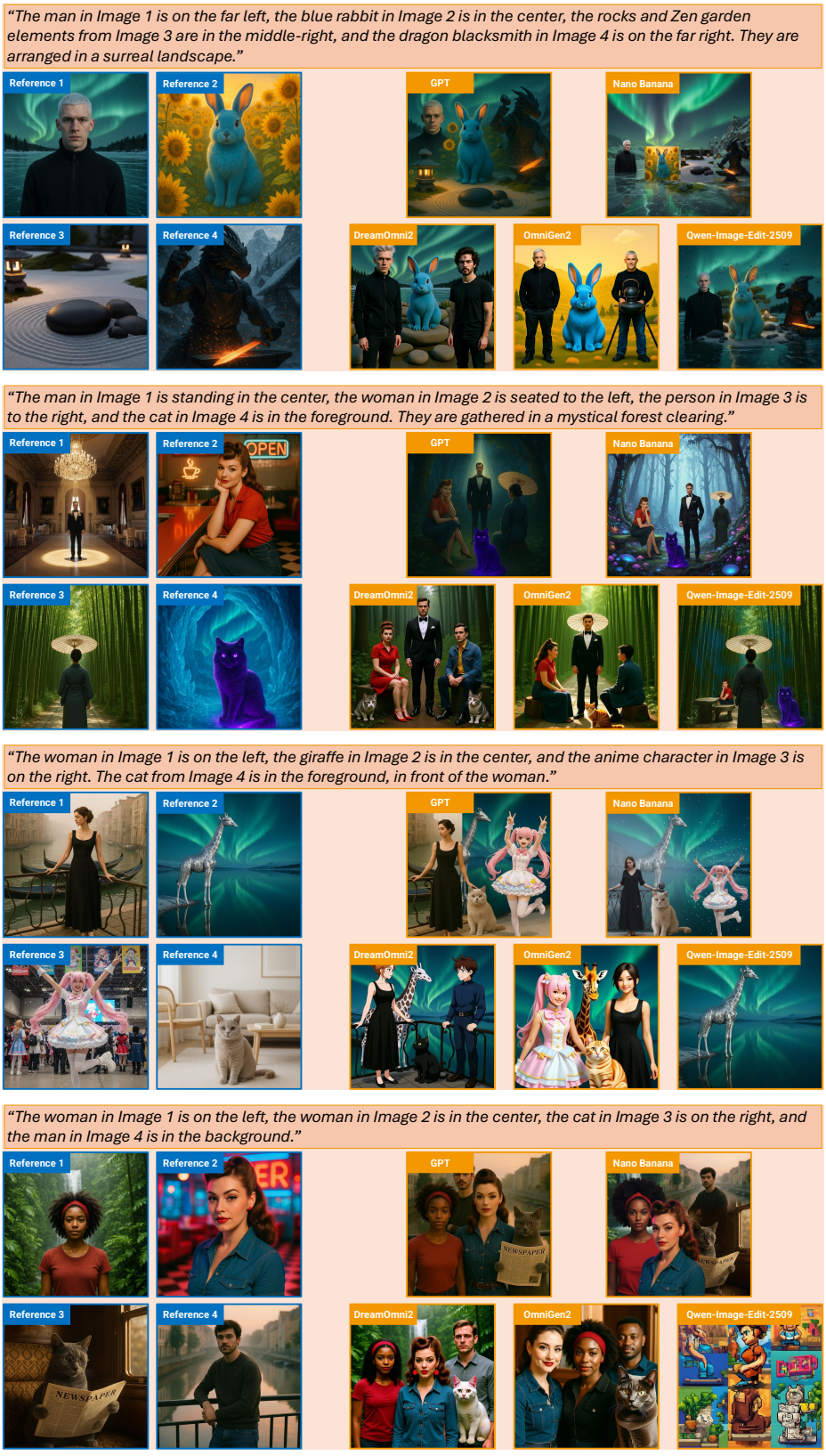

Figure 25. Qualitative example for 4-Object Tasks.

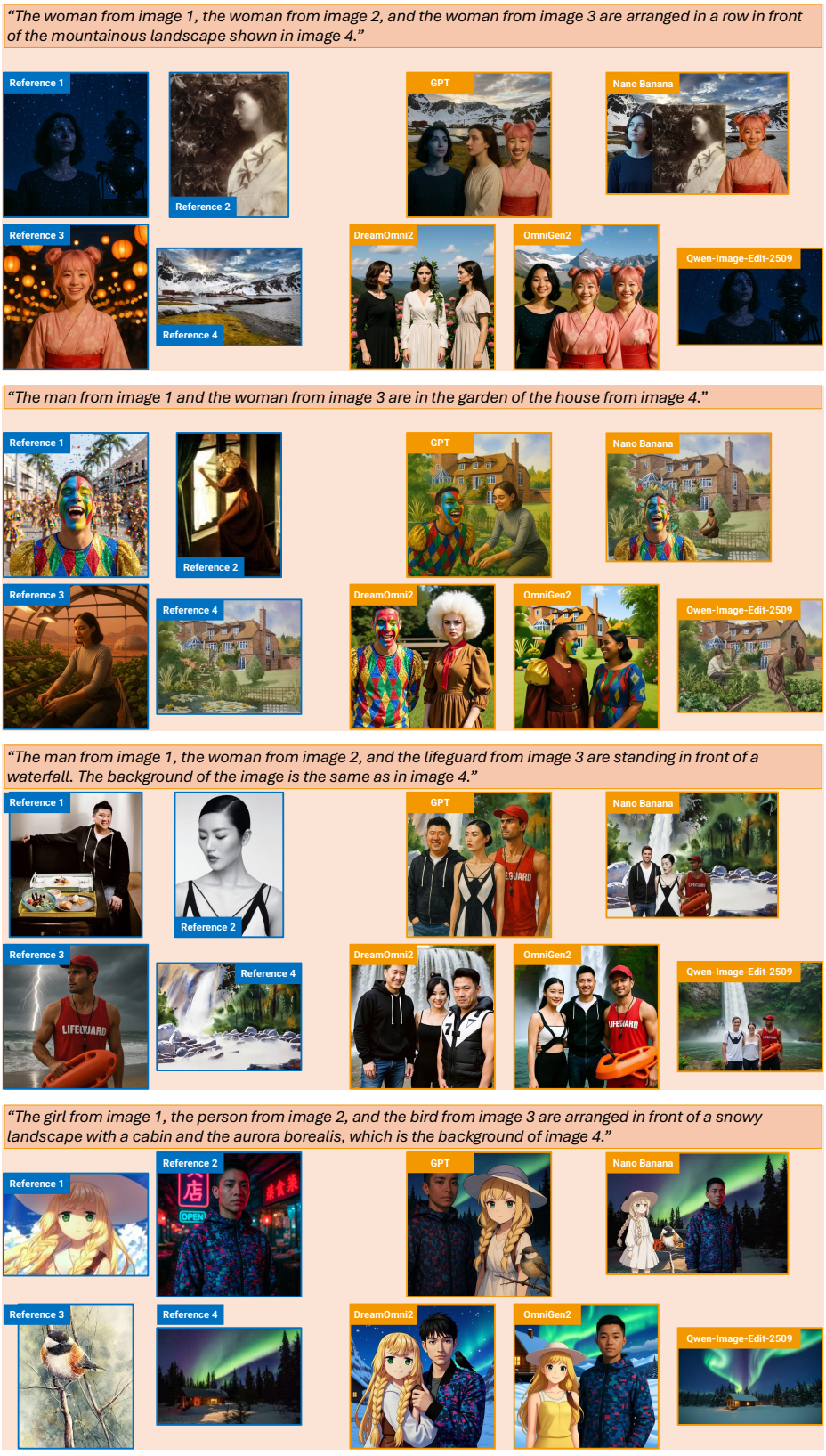

Figure 26. Qualitative example for 3-Object + Background Tasks.

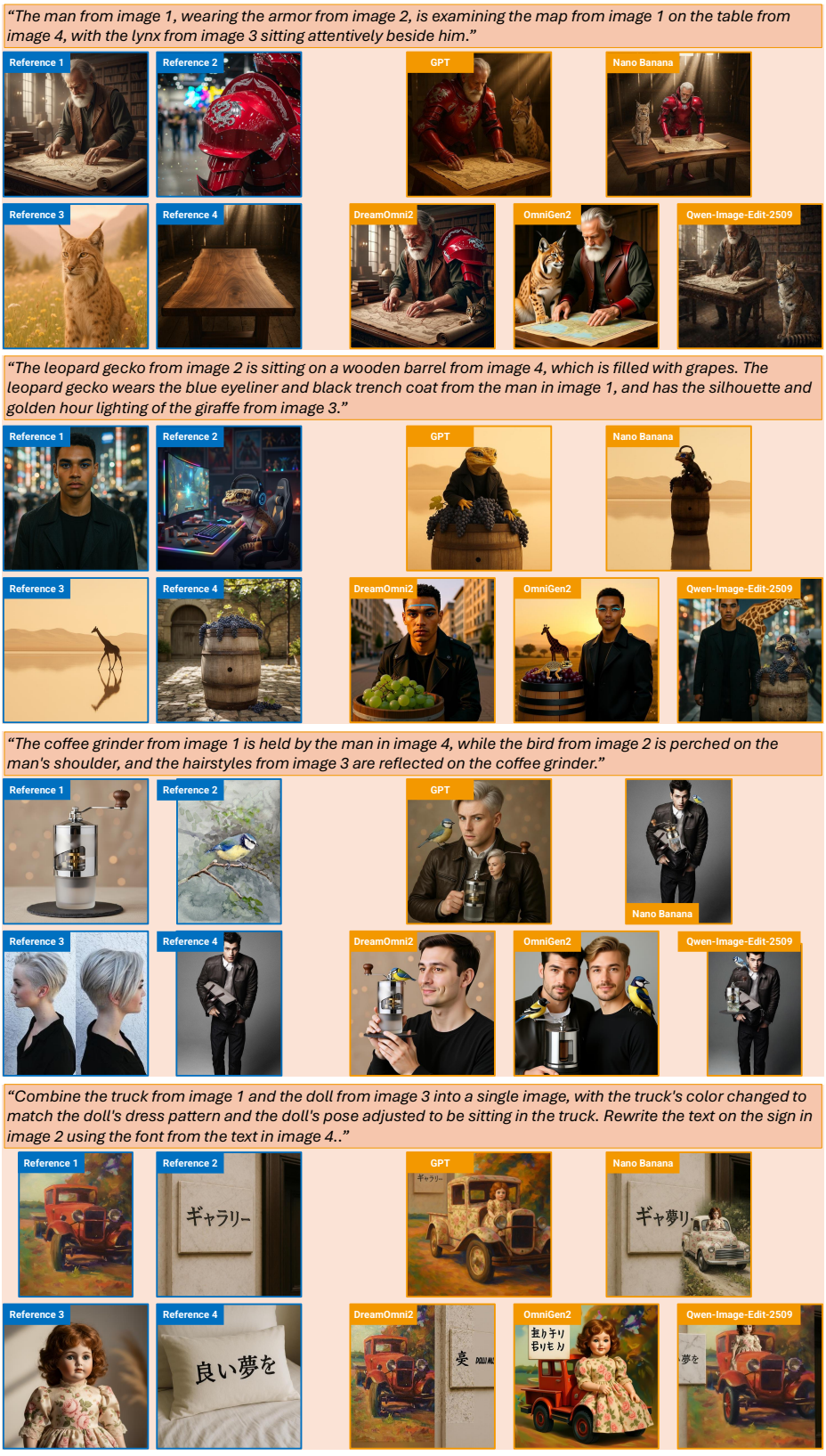

Figure 27. Qualitative example for 3-Object + Local Tasks.

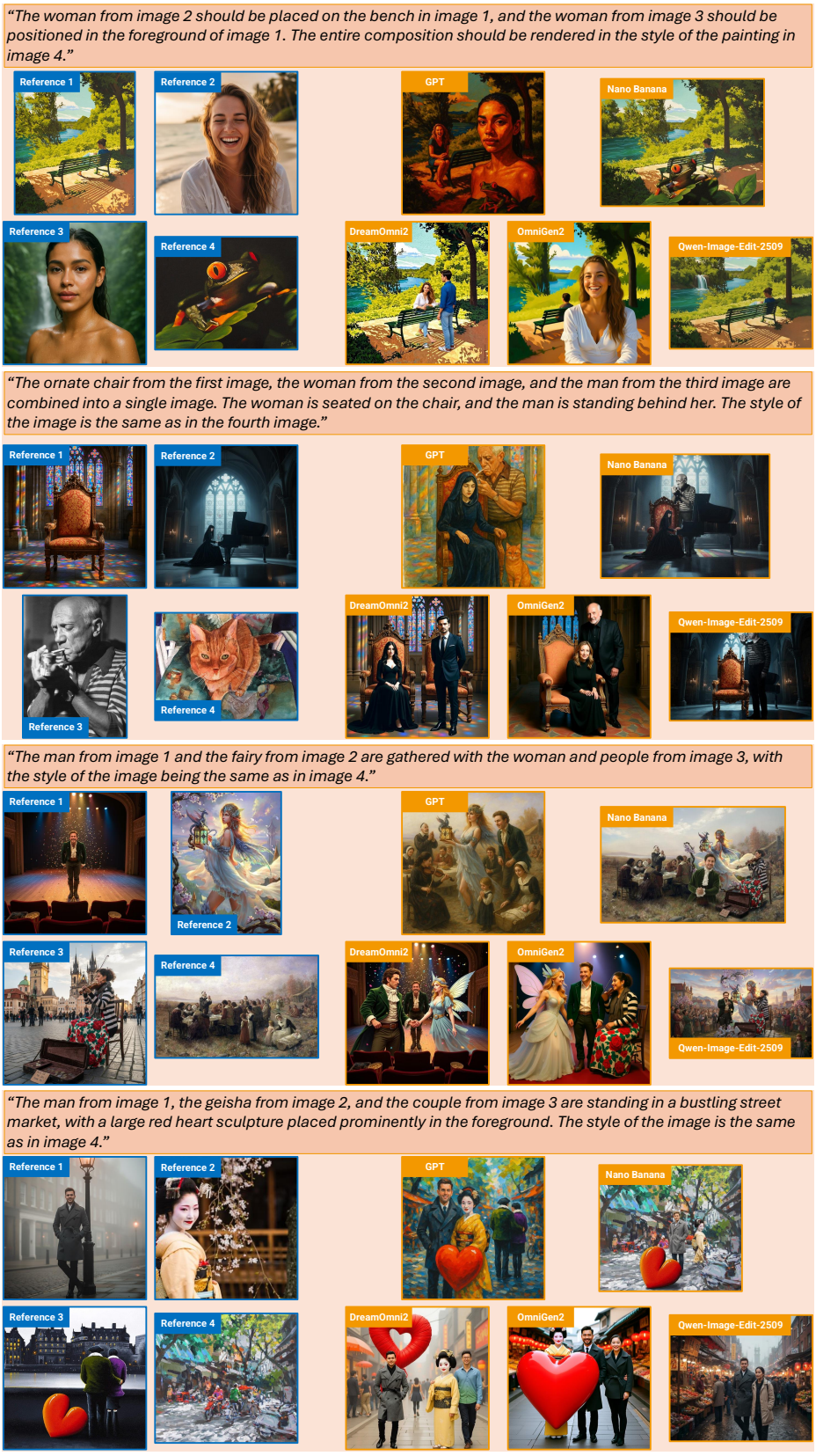

Figure 28. Qualitative example for 3-Object + Global Tasks.

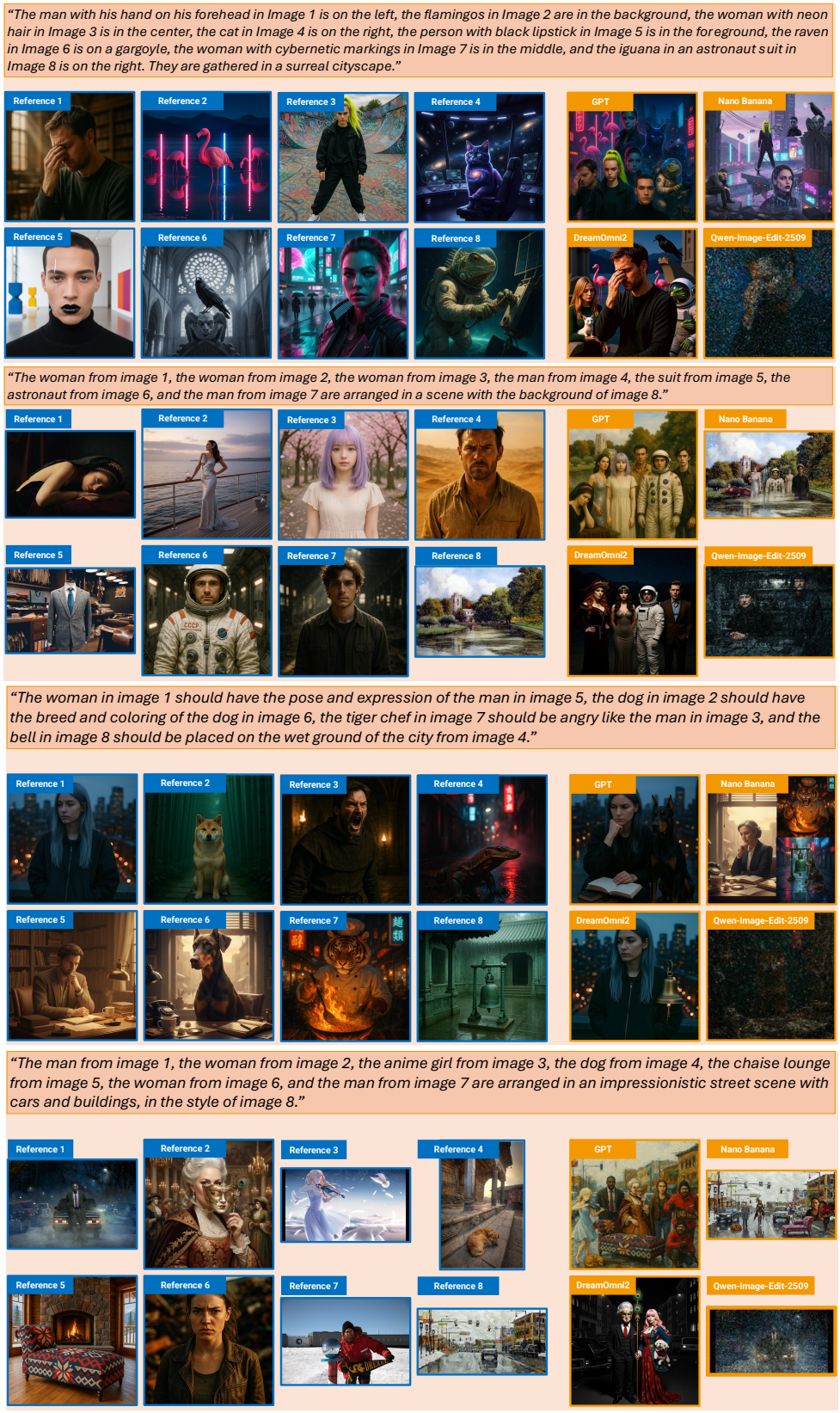

Figure 29. Qualitative example for Tasks with 8 references.