The Image Reconstruction Game: Drawing Common Ground Through Iterative Multimodal Dialogue

Sherzod Hakimov[1], Mattia D'Agostini[1], Ivan Samodelkin[1], David Schlangen[1,2]
[1]Computational Linguistics, Department of Linguistics
University of Potsdam, Germany
[2]German Research Center for Artificial Intelligence (DFKI), Berlin, Germany
{firstname.lastname}@uni-potsdam.de

Abstract

We introduce the Image Reconstruction Game, a fully automated benchmark in which a vision-language model issues corrective instructions to an image generator across multiple turns, making accumulated common ground directly observable as a rendered image. Benchmarking two Describer models crossed with two Generator models across seven image categories, we find that the describer is the dominant factor in reconstruction quality, while the generator determines whether iterative refinement helps or hurts. Mathematical and geometric images pose the greatest challenge. The describer's token budget strongly affects convergence: shorter budgets yield sparser first renderings with more room for visible improvement, while longer budgets raise absolute quality but leave less to fix. Stronger describers use a richer correction vocabulary spanning spatial, numeric, and structural categories, while weaker describers concentrate on surface properties and tend to stop after a few turns. Human validation shows that the best automated judge reaches only slight-to-fair agreement with human preferences, and automated scores require human recalibration to be used reliably.

1 Introduction

Imagine you have a picture "in your head" – a scene that you saw, a diagram that you think could explain something, etc. – that you want to see materialised. Image generation models aim to address this use case: You describe what is "in your head" using language, and the model generates an image. Where early models technically simply were functions from textual prompts to images, modern models set up this task in a more iterative, incremental and dialogic way, where earlier attempts can be corrected and modified. Using terminology from the study of conversations, this can be described as a process of *conversational grounding* (Clark, 1996), where common ground between the interactants is reached through an interactive process of clarification and correction. This paper uses this setup to test current models on the various components of this task: How good are current language and vision models at describing an image in such a way as to steer an image generation model to reproduce it, and how good are such image generation models at being steered? Figure 1 illustrates the outputs of three model pairs across turns to reconstruct the target image through dialogue.

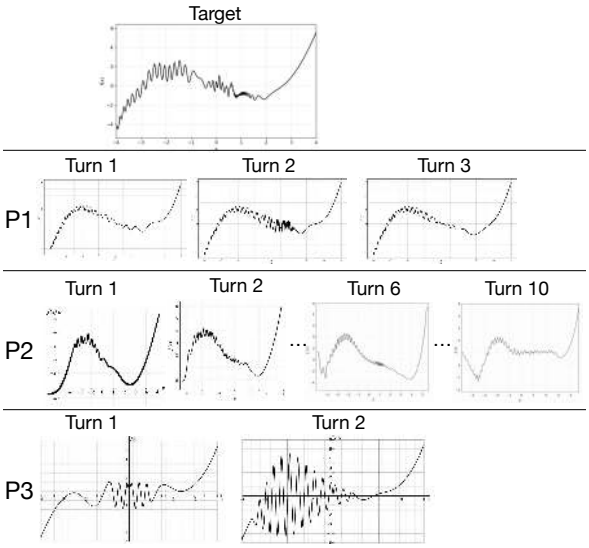


Figure 1: In the IR game, one model (the describer) tries to steer another (the generator) towards faithful reproduction of a target, using only natural language descriptions, over several turns. Figure shows example of how the generated image develops in response to the instructions (which are not shown), for three different model pairings and the same target.

This adds a new dimension to the testing of models that combine linguistic and visual capabilities, as commonly the focus in their evaluation is on non-interactive tasks—captioning, VQA, image–text alignment—that do not test the capacities (Shridhar et al., 2020). Settings such as those used in CoDraw (Kim et al., 2019) establish that iterative visual communication can be studied as a collab-

Findings of the Association for Computational Linguistics: IJCNLP-AACL 2026, pages 2561–2583
November 6-10, 2026 ©2026 Association for Computational Linguistics

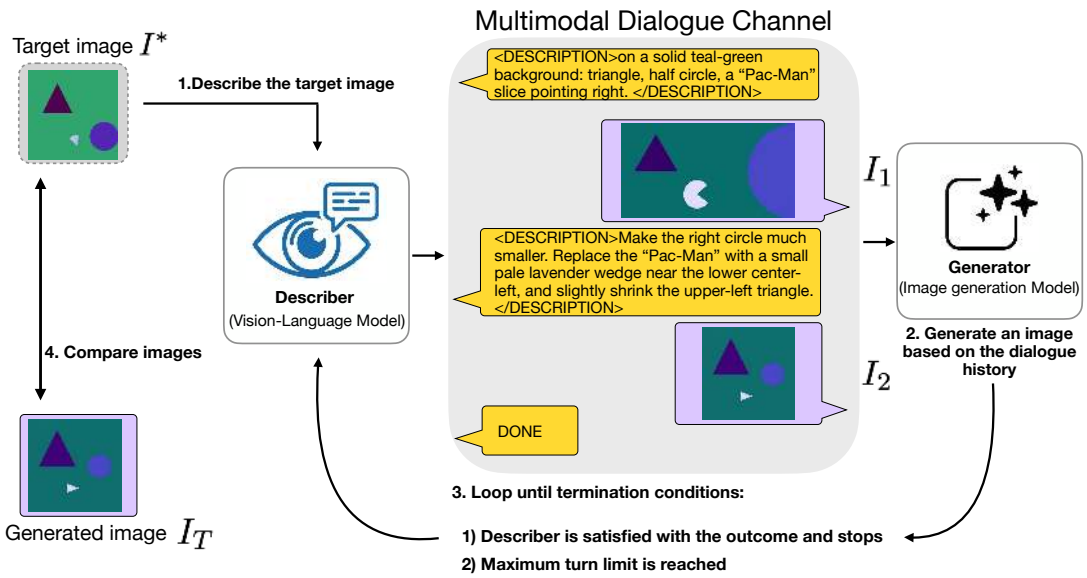


Figure 2: Overview of the Image Reconstruction Game. The *Describer* (a VLM) has access to the target image I^* and communicates with the *Generator* (an image model) exclusively through natural language. The Generator produces a rendering (I_1 or I_2) from the accumulated dialogue; the Describer observes both I^* and the rendering and issues a corrective instruction for the next turn. Each rendering makes the current state of common ground directly visible, and the loop continues until the Describer signals convergence or a turn limit is reached. Comparison is done between target image I^* and the final turn generated image I_T.

orative task with measurable outcomes, but rely on human players and fixed symbolic representations. Conversational image generation systems (El-Nouby et al., 2019; Huang et al., 2025; Ma et al., 2026) place the human in the describer role, targeting user intent rather than evaluating the describer itself. Neither line holds one component fixed while varying the other, and neither connects communication trajectories to human perceptual judgements of outcomes.

We introduce the **Image Reconstruction Game**, a fully automated multimodal dialogue game that fills these gaps (see Figure 2). A *describer* model (a VLM) observes a target image and must steer an *image generator* to reproduce it through a sequence of natural-language turns; at each turn the generator's rendering makes the current state of common ground directly visible, and the describer observes both *target* and all *renderings up to that point in the dialogue* before issuing a corrective instruction. No human is in the loop: the game is played entirely between models under controlled, reproducible conditions. This design lets us investigate four research questions: **RQ1** Can VLMs detect fine-grained perceptual differences between paired images and verbalise them as actionable, corrective instructions? **RQ2** Do those instructions measurably reduce the difference between the target and the current rendering over turns? **RQ3** Does the describer–generator pair build common ground progressively, as reflected in both automated similarity scores and human ratings? **RQ4** Which automated image-similarity metrics best track human perceptual judgements in this setting?

Addressing this last question is non-trivial: embedding-based similarity metrics are known to diverge substantially from human perceptual judgements (Ghazanfari et al., 2024), and LLM-based judges—while better correlated with human ratings than CLIP or BLIP in text-to-image settings (Lu et al., 2023)—cannot be assumed to carry over to image reconstruction without direct empirical validation (Fu et al., 2024; Marjieh et al., 2024). Our contributions are: (i) a new task formulation and instance set spanning seven image categories that vary the difficulty of perception and verbalisation; (ii) a benchmark comparing two describer models against two generators, isolating describer quality from generator variation; (iii) a human similarity rating study on a stratified subset, together with an analysis of which automated metrics are most reliable proxies; and (iv) a qualitative analysis of what kinds of differences are hard to perceive or verbalise[1].

[1]Source code: https://github.com/clp-research/image-reconstruction-game

2 Related Work

CoDraw (Kim et al., 2019) establishes the asymmetric Teller–Drawer paradigm for iterative visual reconstruction through multi-turn dialogue, and related work shows how common ground accumulates across turns when agents describe or identify shared images (Haber et al., 2019; Clark et al., 2021; Madureira and Schlangen, 2023; Testoni and Fernández, 2024; Han and Schlangen, 2017). Game-based frameworks have also been adopted as evaluation instruments for language and multimodal models: Chalamalasetti et al. (2023) systematise this for text-based chat models, and Hakimov et al. (2025) extend it to multimodal models with games probing reference resolution and image comparison (Zhong et al., 2021; Mohapatra, 2023; Li et al., 2024). Our game inherits this paradigm where players are VLM models as describer and generative image model as generator, making common ground directly observable as a rendered image at every turn.

A broad family of systems generates or refines images through multi-turn natural language (Ullah et al., 2022; El-Nouby et al., 2019; Lee et al., 2021; Wei et al., 2023; Wang et al., 2024b,a; Feng et al., 2025; Ma et al., 2025a; Huang et al., 2025; Wang et al., 2025; Ma et al., 2026; Hu et al., 2026; Ye et al., 2026; Ma et al., 2025b; Shen et al., 2026; Yao et al., 2026). In all cases a human user provides instructions; the research question is how to build a better generative pipeline, not how to evaluate the instruction-giver.

Co-creative drawing agents assist human sketchers in open-ended creative tasks (Davis et al., 2016, 2017; Fan et al., 2019; Huang et al., 2020; Zhang et al., 2021; Ibarrola et al., 2024; Davis and Rafner, 2025), while emergent communication research asks whether agents can evolve functional visual protocols through interaction (Fernando et al., 2020; Mihai and Hare, 2021; Qiu et al., 2022; Vinker et al., 2025). Both treat visual output as a communicative medium, but target artistic expression or protocol emergence rather than the fidelity of reconstruction by existing models.

Human evaluation of generated images typically takes the form of pairwise preference judgements or direct similarity ratings, and is used as the ground-truth reference against which automated metrics are calibrated. Standard metrics such as SSIM, LPIPS, DINO, and CLIP are known to diverge from human mid-level perceptual judgements (Ghazanfari et al., 2024), and multimodal LLMs similarly under-perform humans on explicit visual similarity tasks (Fu et al., 2024). LipSim (Ghazanfari et al., 2024) and LLMScore (Lu et al., 2023) address this by providing provably robust perceptual similarity estimates and aligning metrics to human quality ratings respectively, while Marjieh et al. (2024, 2022) show that LLMs can approximate human similarity ratings across perceptual domains, though with systematic biases. We extend this line to image reconstruction, using human ratings to validate LLM judges that also participated as Describer models.

Recent benchmarks specifically targeting mathematical and geometric visual perception—including MathOPEval (Li et al., 2025), GG-Bench (Wei et al., 2026), MATHGLANCE (Sun et al., 2025)—consistently show that diagram-level geometric perception is a critical bottleneck in current VLMs. Where these benchmarks measure whether a model can *read* a geometric figure, the Image Reconstruction Game asks whether a model can *articulate residual differences* between two such figures—a harder, relational task. Our finding that **Functions** and **Geometry** categories pose the greatest challenge aligns with and extends this body of evidence to corrective, goal-directed communication.

3 The Image Reconstruction Game

3.1 Task definition

Game description. The Image Reconstruction Game is a two-player dialogue game implemented within the `clem` framework for game-based model evaluation (Chalamalasetti et al., 2023; Hakimov et al., 2025). In this framework, a programmatic GAMEMASTER instantiates each game episode with a given target image, manages turn-by-turn prompting of the two players, enforces the communication protocol, and records the episode for scoring.

The two players occupy asymmetric roles with different information access and capabilities (see Figure 2). The *Describer* is a Vision-and-Language Model that has full access to the target image I^*. Its goal is to produce natural-language utterances that guide the Generator toward a faithful visual reproduction of I^*. The *Generator* is an image generation model that never observes I^* directly; it can only perceive the accumulated dialogue history—the sequence of Describer utterances—and must

render an image conditioned on that text.

The game unfolds over multiple turns: at each turn the Describer observes both I^* and the Generator's previous rendering I_{t-1} (t stands for the turn index), then issues a corrective instruction or signals convergence; the Generator produces a new image based on the dialogue and previous rendering.

This iterative structure operationalises the progressive building of *common ground* (Alikhani and Stone, 2020): each rendering I_t is a visible projection of the shared state constructed so far, and each Describer utterance is a grounding act that attempts to reduce the residual gap. The asymmetric information design—where only the Describer knows I^*— mirrors the Teller–Drawer setup of CoDraw (Kim et al., 2019) and multimodal dialogue games (referential, collaboration, spatial navigation) studied by Hakimov et al. (2025), while extending them to open-ended natural-language communication with modern generative image models.

Capability tested. The game is designed to probe two interleaved capabilities that standard VLM benchmarks (Agrawal et al., 2017; Yue et al., 2024) do not jointly assess. First, the Describer must *perceive* fine-grained differences between I^* and I_{t-1}— a visual discrimination task that is considerably harder than captioning a single image and that demands sensitivity to spatial, metric, and relational properties. Second, it must *verbalise* those differences as specific corrective instructions for the Generator to act on, which requires goal-directed language production rather than generic scene description. The Generator, in turn, must interpret and act on incremental instructions while maintaining coherence with what has already been rendered. Jointly, these demands test whether a VLM can function as a goal-directed communicative agent in a closed-loop, multimodal setting.

3.2 Prompts and communication protocol

Each player's messages must conform to the communication protocol so that the GAMEMASTER can validate, parse, and relay them between players. An episode continues until one of three stopping criteria is met: (1) Player A signals convergence by producing a DONE action; (2) the maximum turn limit of T=10 is reached; or (3) a player violates the protocol.

Player A (Describer). Player A is a VLM that receives the target image I^* and the most recent Generator rendering I_{t-1} at every turn (on turn 1, only I^* is available, as no rendering exists yet). The GAMEMASTER presents these images with a prompt that asks Player A either to describe I^* in sufficient detail for an image generator to reproduce it (turn 1), or to identify specific remaining differences between I^* and I_{t-1} and issue a targeted corrective instruction (turns $t > 1$). Player A must wrap its output in `<DESCRIPTION> ... </DESCRIPTION>` tags so that the GAMEMASTER can parse and relay the message to Player B; a response consisting solely of DONE (without tags) signals convergence and ends the episode. A token budget of 200 tokens per turn is imposed to enforce concise, targeted corrections rather than exhaustive re-descriptions.

The correction turns place a higher demand on visual reasoning than the initial description: corrections must be *relational*—specifying what is wrong relative to the current rendering—should focus on what to improve rather than descriptions of the image from scratch.

Player B (Generator). Player B is an image generation model that receives two inputs at each turn: the previous rendering I_{t-1} and the text message produced by Player A in the same turn. It does not have access to I^* at any point; its only signal is the accumulated dialogue history together with the visual context of its own previous output. The GAMEMASTER also passes the full sequence of prior Player A utterances alongside I_{t-1}, ensuring that compositional information established in early turns is not lost when later turns focus on local corrections. Player B's sole output is a new rendered image I_t, which is then forwarded to Player A as the basis for the next turn.

All prompt templates are given in Appendix C.

4 Experimental Setup

4.1 Models

Describer models. We evaluate a set of publicly available VLMs in the Describer role. QWEN3-VL-30B-A3B-INSTRUCT (Yang et al., 2025) was run locally on a single NVIDIA A100 GPU and GPT-5.2 model was accessed via API. A fixed token budget of *200 tokens* was used.

Generator models. We use two image generation models as generators: GEMINI-3.1-NANO-BANANA and GPT-IMAGE-1.5. Both are accessed via their respective image generation APIs

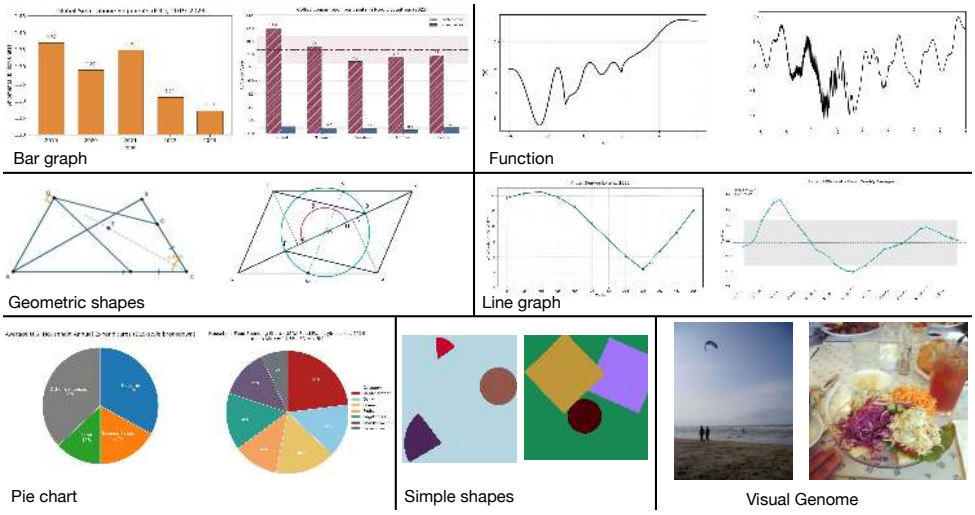


Figure 3: Representative target images (*easy & hard* level) for seven categories in the dataset

in *image-to-image conditioning* mode: at each turn the Generator receives the previous rendering I_{t-1} together with the accumulated dialogue history, not merely the latest instruction. Calls are made with default sampling parameters; seeds are not fixed across turns (as neither API exposes turn-level seed control), so observed score changes across turns reflect a combination of conditioning behaviour and sampling stochasticity. Each Describer is evaluated against each Generator, yielding a Describer $\times$ Generator grid of results; the Generator is held fixed within each grid cell so that outcome differences are attributable to the Describer's communicative quality rather than to rendering capability.

4.2 Benchmark conditions

The benchmark comprises **14 experiment conditions** obtained by crossing seven instance categories with two difficulty levels, *easy* and *hard*. Each condition contains **10 episodes**, yielding **140 episodes** in total. The six categories are generated synthetically by prompting an LLM to produce Python code, the code is executed to obtain the target image (see Figure 22 in Appendix C).

Figure 3 shows representative target images. **Simple Shapes** contain coloured geometric primitives at specified positions; differences are visually salient but require precise spatial and color language. **Geometry** diagrams depict labelled figures and constructions with subtle metric differences. **Functions** show mathematical function graphs; differences concern curve shape, scale, and axis labels. **Line Graph**, **Bar Graph**, and **Pie Graph** encode statistical data visually; corrections must be numerically precise. We also use real photographs from the **Visual Genome** (Krishna et al., 2017) dataset to compare the with synthetic categories.

Difficulty is defined consistently across synthetic categories: *easy* instances use fewer elements, simpler structure, and unambiguous layouts, while *hard* instances introduce additional complexity—more data series, finer geometric detail, denser labelling—that increases both perceptual and verbalisation demands. For Visual Genome, easy instances depict scenes with a small number of salient objects; hard instances are denser and more cluttered. All instances were checked and verified manually (synthetic ones were generated with the complexity included in the prompt and then confirmed, Visual Genome instances were manually selected). Hard instances receive consistently lower mean similarity scores than easy instances across all categories and model pairs (Table 3, Appendix B), confirming the intended difficulty ordering.

4.3 Tasks & Evaluation

We evaluate episode outcomes using two tasks.

Task 1: Pairwise Similarity Judges are shown the target image (*Image A*) and a Generator rendering (*Image B*) and asked to rate their visual and semantic similarity on a 0–10 scale:

> *How similar is Image B to Image A? Image A is the original. Image B was generated by a model. Please rate their overall visual & semantic similarity using the scale below:*
>
> **0** Completely different image
> **1–3** Very different; only a few elements in common
> **4–6** Partial match; some key elements captured, but important differences remain
> **7–9** Strong match; most elements correct, with minor differences in detail, colour, or placement
> **10** Identical or indistinguishable

Task 2: Triplet Preference Judges are shown the target image alongside two Generator renderings (from the first and the final turns) and asked to *select which rendering is more visually similar to the target*. This two-alternative forced-choice format abstracts away from absolute scale calibration and directly measures relative reconstruction quality—for example, whether a later-turn rendering is preferred over an earlier one considering the iterative improvement across turns.

Human judges. We collect human ratings for both tasks by asking them to annotate all 140 episodes. Each data point in both tasks is rated by three independent annotators with backgrounds in Computational Linguistics and Computer Science (who are also authors). The annotation guidelines and interfaces are given in Appendix A.

LLM judges. We use three LLMs as judges for both tasks. GPT-5.2 and QWEN3-VL-30B are drawn from the same model families used in the Describer role, allowing us to examine whether a model's ratings are consistent with—or systematically biased by—its prior exposure to the task as a generator of descriptions. Self-serving evaluation, in which a model rates its own outputs more favourably than an independent observer would, is a known failure mode of LLM-as-judge pipelines (Fu et al., 2024; Marjieh et al., 2024); including these models as judges provides a direct test of whether this bias is present in our setting. CLAUDE-4.5 SONNET was not used in experiments and therefore we included it as an independent, out-of-distribution LLM judge. Comparing the three judges against each other and against human ratings allows us to assess both the overall reliability of LLM-based evaluation in this setting and the degree to which judge scores are confounded by a model's participation in the task being evaluated. All three models receive the same prompt and rating scale as human annotators.

Automated metrics. LipSim (Ghazanfari et al., 2024) serves as the automated metric for Task 2; standard alternatives (SSIM, LPIPS, DINO, CLIP) are excluded as they are well-documented to diverge from human mid-level perceptual judgements (Ghazanfari et al., 2024; Fu et al., 2024). LipSim provides a provably robust perceptual similarity distance using Lipschitz-constrained neural networks, offering certified stability under input perturbations.

Table 1: Automatic judge calibration across annotation tasks (all model pairs). **Task 1** — Pairwise Similarity: Pearson correlation (r) and Spearman rank correlation (ρ) on matched instance–turn pairs. **Task 2** — Triplet Preference: pairwise agreement rate (%) and Cohen's κ on binary final-vs-turn-0 choices. LipSim is treated as an additional perceptual judge.

Task 1 — Pairwise Similarity			
Judge A	Judge B	r	ρ
GPT	Claude	0.76	0.74
GPT	Qwen3	0.65	0.65
Claude	Qwen3	0.63	0.65
Task 2 — Triplet Preference			
Judge A	Judge B	%	κ
GPT	Claude	62.28	0.24
GPT	Qwen	53.29	0.06
Claude	Qwen	49.10	-0.02
GPT	LipSim	62.57	0.22
Claude	LipSim	52.99	0.05
Qwen	LipSim	57.78	0.15

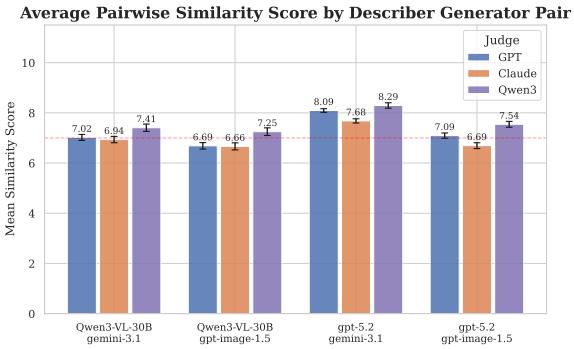


Figure 4: Task-1: Mean similarity scores for all four describer–generator pairs under LLM judges.

5 Results and Analysis

5.1 Automated judge calibration

Table 1 reports pairwise judge agreement for both evaluation tasks. Figure 4 shows that all three automated judges agree mainly on the overall ranking of model pairs, with GPT-5.2 + GEMINI-3.1 consistently highest and QWEN3 + GPT-IMAGE-1.5 consistently lowest among all judges.

Task 1 (pairwise similarity). GPT-5.2 and Claude-4.5 agree more closely ($r = 0.76$, $\rho = 0.74$), while Qwen-3 shows lower alignment with both ($r = 0.65$ / $\rho = 0.65$ vs. GPT-5.2; $r = 0.63$ / $\rho = 0.65$ vs. Claude).

Task 2 (triplet preference). GPT-5.2 and Claude-4.5 reach fair agreement (62.2%, $\kappa = 0.23$), while Qwen-3 shows lower alignment. The *LipSim* aligns more closely with GPT-5.2 (62.6%, $\kappa = 0.22$).

5.2 Overall Performance

Figure 4 shows mean similarity scores for all four describer–generator pairs (full breakdown in Table 3, Appendix B). **The describer is the dominant factor (RQ1)**: GPT-5.2 outperforms QWEN-3 by approximately one point across all conditions, with the ranking stable across all three judges. Within each describer, GEMINI-3.1 yields higher scores than GPT-IMAGE-1.5 (**RQ2**). Category-level results reveal two separable bottlenecks: *perception* limits scores on **Functions** and **Geometry** hard instances (fine-grained details such as curve shapes, axis ticks), while *verbalisation* is the separate challenge of converting perceived differences into actionable instructions.

5.3 Iterative Refinement

The iterative payoff (the difference in scores between the final turn image in comparison to the first turn image) is as follows and address the **RQ3**: $\Delta_Q = \text{score}_{I_T} - \text{score}_{I_1}$. GEMINI-3.1 pairings improve across turns; **GPT-IMAGE-1.5 pairings show near-zero or negative deltas—additional turns degrade rather than improve the reconstruction** (Figures 15 and 16, Appendix B). **The generator also determines whether iterative refinement is beneficial**. Figure 5 shows convergence outcomes per pair where each category refers to whether the first turn image has changed in scores in comparison to the final turn one. Qwen3-based pairs produce mostly single-turn episodes, leaving little room for improvement or regression; GPT-5.2/GPT-IMAGE-1.5 continues to the turn limit but still regresses in roughly a third of cases. More turns do not guarantee higher quality (see also Figure 19, Appendix B). Among all pairs, GPT-5.2 + GEMINI-3.1 shows the most consistent per-turn improvement.

5.4 Describer Token Budget

To analyse the effect of the token budget, we also ran GPT-5.2 + GPT-IMAGE-1.5 with 10, 50 tokens per turn to compare with 200 tokens experiments (see Figure 7). A shorter budget yields more visibly improved final images: at 10 tokens the first rendering I_1 is sparse, so corrections have substantial room to close the gap; at 200 tokens I_1 is already detailed, leaving little room for improvement—and more for regression. Absolute final-turn scores increase with budget while triplet preference for the final image decreases because it is easier to tell apart the first rendering from the final when the budget is low (full statistics in Figures 17 and 18, Appendix B). **This quality–stability trade-off is driven by the starting-point quality of I_1.**

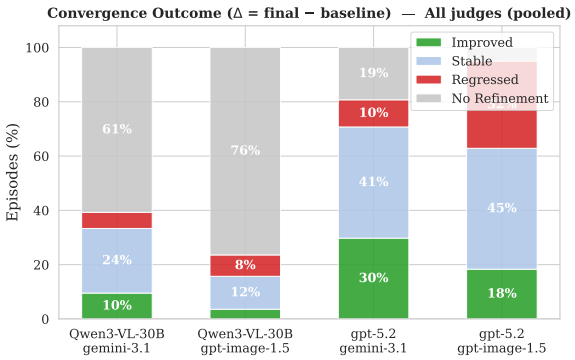


Figure 5: Convergence outcome distribution per model pair (pooled across all three LLM judges). *Improved* ($\Delta > 0$), *Stable* ($\Delta = 0$), *Regressed* ($\Delta < 0$), *No Refinement* (single-turn episode).

5.5 Correction Language

We collected a list of words for respective categories that the describer uses to correct the rendering on subsequent turns. Figure 8 summarises the per-category usage rate per model pair. GPT-5.2 corrections are lexically broad, spanning LABEL/TEXT, NUMERIC, COMPARISON, and POSITIVE ACKNOWLEDGEMENT in addition to the universal POSITION and COLOR axes. QWEN3-VL-30B corrections concentrate almost exclusively on COLOR and SHAPE, with near-zero NUMERIC use (GPT-5.2 uses coordinate notation such as *x=0*, *y=0*; Qwen3 contributes almost nothing) and zero POSITIVE ACKNOWLEDGEMENT. Qwen3-based pairs produce mostly single-turn episodes, so the language profile reflects only the coarsest initial instructions, with no opportunity for the structural and numeric precision that GPT-5.2 develops over longer dialogues (Table 5: full list of keywords, Figure 14: top-10 keywords).

5.6 Human Judgements

Table 2 reports human–automated judge calibration for the GPT-5.2 + GPT-IMAGE-1.5 pair. Human annotators showed moderate mutual agreement: mean $r = 0.48$ / $\rho = 0.49$ on Task 1 and mean $\kappa = 0.29$ on Task 2 (Table 4, Appendix A). **GPT-5.2 is the most human-aligned judge (RQ4) but still very low for practical applications**: Task-1 correlation $r = 0.56$ is the highest among all automated judges, and Task-2 triplet agreement is

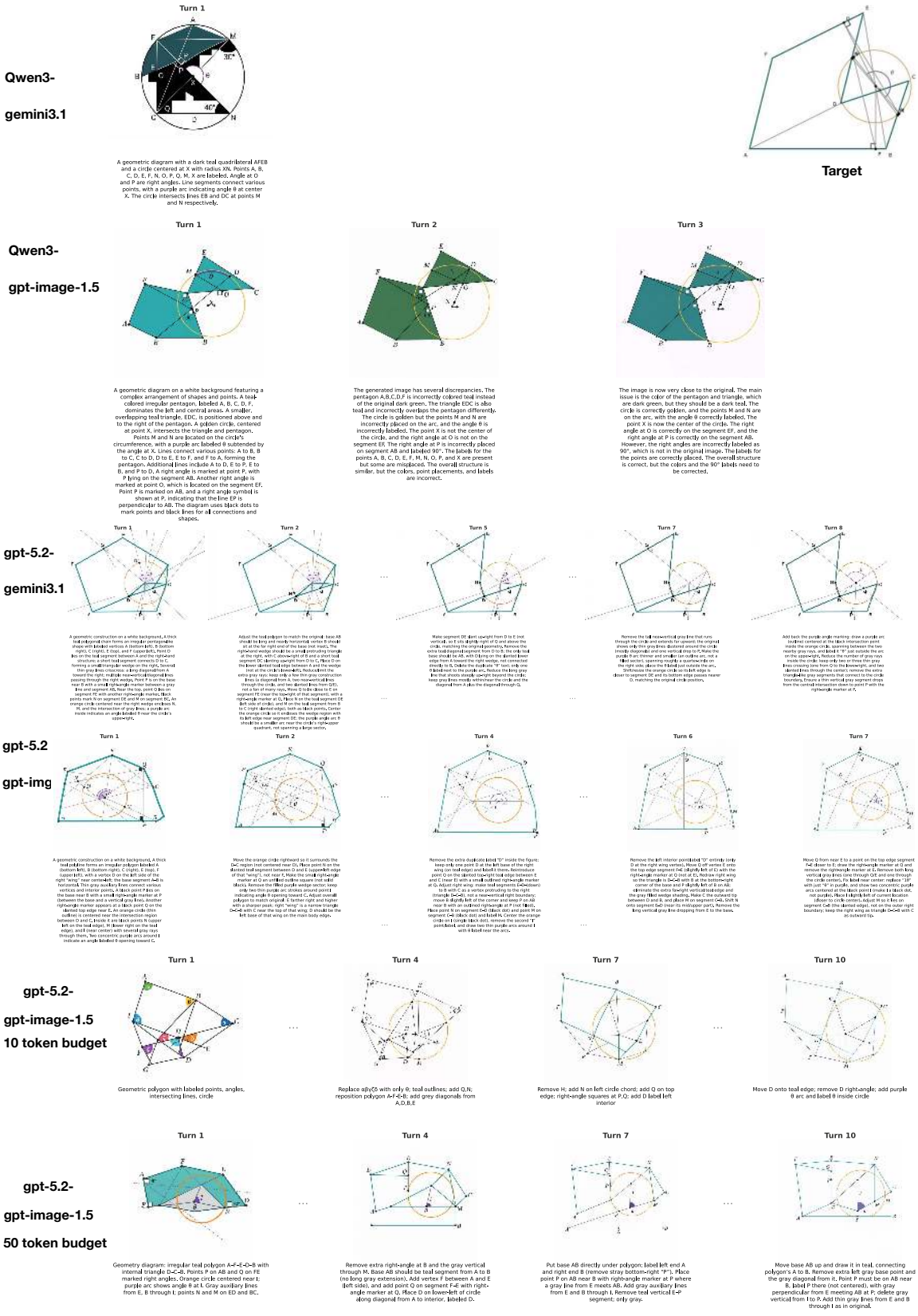


Figure 6: Qualitative samples for *Geometry* (hard, instance#3), model pairs and token budget experiments

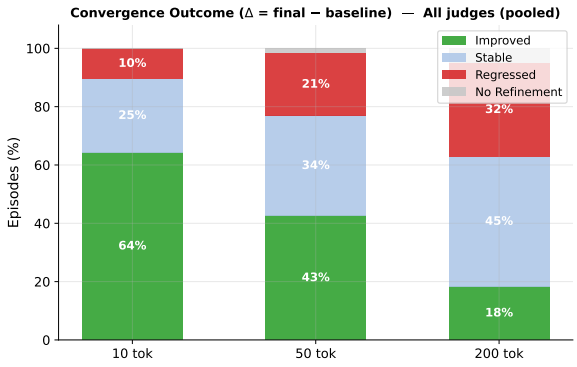


Figure 7: Convergence outcomes by token budget for GPT-5.2 + GPT-IMAGE-1.5. A shorter budget produces a sparser first rendering, giving corrections more visible room to improve it.

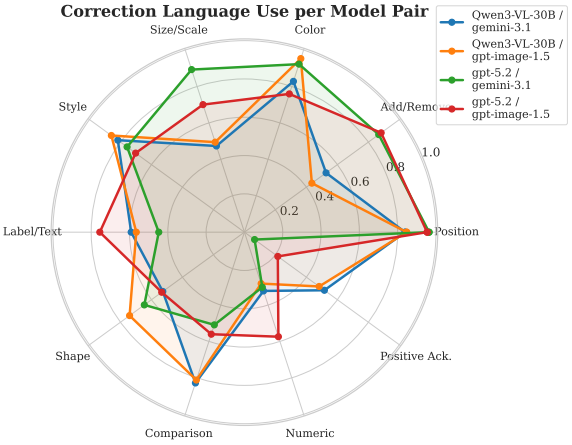


Figure 8: Correction language profiles per model pair.

$\kappa = 0.22$. All three automated judges are systematically more tolerant than humans: human annotators assign substantially lower mean similarity scores, with Qwen3 showing the largest upward bias. **Automated scores require human recalibration before they can be used to set quality thresholds.** Qwen3's near-chance triplet agreement with humans ($\kappa \approx 0$), despite competitive static benchmark performance, illustrates that game-based probing surfaces failure modes that single-turn evaluation misses. LipSim is a practical automated complement for Task-2, matching Claude's human agreement at lower inference cost.

5.7 Qualitative Analysis

Qualitative samples are available in Figure 6, Figure 9, and Figure 10 for *Geometry*, *Functions* and *Visual Genome* instances. Across all three category types, GPT-5.2 + GEMINI-3.1 produces the closest renderings; GPT-5.2 + GPT-IMAGE-1.5 runs to ten turns but still falls short on fine details. Lower token budgets and Qwen3-based pairs yield semantically weaker reconstructions, with key structural elements missing or substituted.

Table 2: Human vs automatic judge calibration (GPT-5.2/GPT-IMAGE-1.5). **Task 1** — Pairwise Similarity: Pearson correlation (r) and Spearman rank correlation (ρ). **Task 2** — Triplet Preference: agreement rate (%) and Cohen's κ between Human and each LLM judge and LipSim perceptual metric.

Task 1 — Pairwise Similarity			
Judge A	Judge B	r	ρ
Human	GPT	0.56	0.60
Human	Claude	0.51	0.54
Human	Qwen3	0.41	0.40
Task 2 — Triplet Preference			
Judge A	Judge B	%	κ
Human	GPT	60.90	0.22
Human	Claude	54.89	0.11
Human	Qwen	48.87	-0.01
Human	LipSim	54.89	0.06

6 Conclusion

We introduced the Image Reconstruction Game, a fully automated multimodal reference game that turns common ground into a directly observable artefact: the rendered image produced after each dialogue turn. Unlike static benchmarks, the game separately measures two distinct model competencies—the ability to perceive and verbalise residual differences (Describer), and the ability to incorporate targeted corrections without losing prior progress (Generator). Our experiments show that describer quality is the dominant factor in reconstruction fidelity, while the generator determines whether multi-turn interaction provides any benefit at all. Notably, stronger VLMs are capable of fine-grained visual articulation: even for mathematically complex and geometrically unusual images, they produce detailed, actionable correction sequences. Functions and geometry categories nonetheless pose the greatest challenge, confirming that fine-grained geometric and mathematical perception remains a critical bottleneck for current VLMs—consistent with recent geometric benchmarks. Human validation reveals that automated LLM judges systematically overestimate reconstruction quality, and even the best-aligned judge remains unreliable for episode-level decisions without human recalibration.

Limitations

The benchmark comprises 140 episodes across four Describer–Generator pairs (10 per category–difficulty condition)—sufficient for exploratory analysis but modest for robust category-level conclusions and cross-model generalisation; the observed rankings may shift with additional model pairings or larger episode counts. Human annotations are available for a single model pair (GPT-5.2 + GPT-IMAGE-1.5), limiting the scope of the human–automated correlation analysis; extending ratings to further pairs would validate whether iterative improvement observed by LLM judges is also perceived by human raters. All prompts and evaluations are in English; the game's generalisation to other languages is not assessed. The automated judges are themselves subject to the perceptual and linguistic biases of their training data, and their ratings should be interpreted alongside, rather than as a replacement for, human judgements. The benchmark currently covers seven image categories; extending it to broader visual domains, including natural photographs at higher resolution, remains future work.

Ethics Statement

All images in the dataset are either synthetically generated or drawn from Visual Genome (Krishna et al., 2017), a publicly available, consent-collected dataset. No personally identifiable information is present in the stimuli. Annotation tasks were designed to be low-risk and non-sensitive. The generated images produced by the game's Generator models may contain artefacts or unintended content. Automated evaluation with LLM judges carries the general risk that model-specific biases propagate into benchmark rankings; we mitigate this by using three judges from distinct model families and validating against human ratings. All game transcripts, model outputs, human annotation data, and evaluation code will be publicly released to support reproducible follow-up research.

References

Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Devi Parikh, and Dhruv Batra. 2017. VQA: visual question answering - www.visualqa.org. *Int. J. Comput. Vis.*, 123(1):4–31.

Malihe Alikhani and Matthew Stone. 2020. Achieving common ground in multi-modal dialogue. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts, ACL 2020, Online, July 5, 2020*, pages 10–15. Association for Computational Linguistics.

Kranti Chalamalasetti, Jana Götze, Sherzod Hakimov, Brielen Madureira, Philipp Sadler, and David Schlangen. 2023. clembench: Using game play to evaluate chat-optimized language models as conversational agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 11174–11219. Association for Computational Linguistics.

Christopher Clark, Jordi Salvador, Dustin Schwenk, Derrick Bonafilia, Mark Yatskar, Eric Kolve, Alvaro Herrasti, Jonghyun Choi, Sachin Mehta, Sam Skjonsberg, Carissa Schoenick, Aaron Sarnat, Hannaneh Hajishirzi, Aniruddha Kembhavi, Oren Etzioni, and Ali Farhadi. 2021. Iconary: A pictionary-based game for testing multimodal communication with drawings and text. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 1864–1886. Association for Computational Linguistics.

Herbert H. Clark. 1996. *Using Language*. Cambridge University Press, Cambridge.

Nicholas Davis, Chih-Pin Hsiao, Kunwar Yashraj Singh, Boyang Lin, and Brian Magerko. 2017. Quantifying collaboration with a co-creative drawing agent. *ACM Transactions on Interactive Intelligent Systems*, 7(4).

Nicholas Davis, Chih-Pin Hsiao, Kunwar Yashraj Singh, and Brian Magerko. 2016. Co-creative drawing agent with object recognition. In *Proceedings of the Twelfth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, AIIDE'16, pages 9–15. AAAI Press.

Nicholas Davis and Janet Rafner. 2025. AI drawing partner: Co-creative drawing agent and research platform to model co-creation. *CoRR*, abs/2501.06607.

Alaaeldin El-Nouby, Shikhar Sharma, Hannes Schulz, R. Devon Hjelm, Layla El Asri, Samira Ebrahimi Kahou, Yoshua Bengio, and Graham W. Taylor. 2019. Tell, draw, and repeat: Generating and modifying images based on continual linguistic instruction. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 10303–10311. IEEE.

Judith E. Fan, Monica Dinculescu, and David Ha. 2019. collabdraw: An environment for collaborative sketching with an artificial agent. In *Proceedings of the 2019 Conference on Creativity and Cognition*, C&C '19, page 556–561, New York, NY, USA. Association for Computing Machinery.

Yuheng Feng, Jianhui Wang, Kun Li, Sida Li, Tianyu Shi, Haoyue Han, Miao Zhang, and Xueqian Wang.

2025. TDRI: two-phase dialogue refinement and co-adaptation for interactive image generation. *CoRR*, abs/2503.17669.

Chrisantha Fernando, Daria Zenkova, Stanislav Nikolov, and Simon Osindero. 2020. From language games to drawing games. *CoRR*, abs/2010.02820.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. 2024. BLINK: multimodal large language models can see but not perceive. In *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part XXIII*, Lecture Notes in Computer Science, pages 148–166. Springer.

Sara Ghazanfari, Alexandre Araujo, Prashanth Krishnamurthy, Farshad Khorrami, and Siddharth Garg. 2024. Lipsim: A provably robust perceptual similarity metric. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Janosch Haber, Tim Baumgärtner, Ece Takmaz, Lieke Gelderloos, Elia Bruni, and Raquel Fernández. 2019. The photobook dataset: Building common ground through visually-grounded dialogue. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1895–1910. Association for Computational Linguistics.

Sherzod Hakimov, Yerkezhan Abdullayeva, Kushal Koshti, Antonia Schmidt, Yan Weiser, Anne Beyer, and David Schlangen. 2025. Using game play to investigate multimodal and conversational grounding in large multimodal models. In *Proceedings of the 31st International Conference on Computational Linguistics, COLING 2025, Abu Dhabi, UAE, January 19-24, 2025*, pages 5686–5718. Association for Computational Linguistics.

Ting Han and David Schlangen. 2017. Draw and tell: Multimodal descriptions outperform verbal- or sketch-only descriptions in an image retrieval task. In *Proceedings of the 8th International Joint Conference on Natural Language Processing (IJCNLP 2017), Volume 2: Short Papers*, pages 361–365. Asian Federation of Natural Language Processing.

Yujie Hu, Zecheng Tang, Xu Jiang, Weiqi Li, and Jian Zhang. 2026. Talkphoto: A versatile training-free conversational assistant for intelligent image editing. *CoRR*, abs/2601.01915.

Forrest Huang, Eldon Schoop, David Ha, and John Canny. 2020. Scones: Towards conversational authoring of sketches. In *Proceedings of the 25th International Conference on Intelligent User Interfaces*, pages 313–323. ACM.

Minbin Huang, Yanxin Long, Xinchi Deng, Ruihang Chu, Jiangfeng Xiong, Xiaodan Liang, Hong Cheng, Qinglin Lu, and Wei Liu. 2025. Dialoggen: Multi-modal interactive dialogue system with multi-turn text-image generation. In *Findings of the Association for Computational Linguistics: NAACL 2025, Albuquerque, New Mexico, USA, April 29 - May 4, 2025*, Findings of ACL, pages 411–426. Association for Computational Linguistics.

Francisco Ibarrola, Tomas Lawton, and Kazjon Grace. 2024. A collaborative, interactive and context-aware drawing agent for co-creative design. *IEEE Trans. Vis. Comput. Graph.*, 30(8):5525–5537.

Jin-Hwa Kim, Nikita Kitaev, Xinlei Chen, Marcus Rohrbach, Byoung-Tak Zhang, Yuandong Tian, Dhruv Batra, and Devi Parikh. 2019. Codraw: Collaborative drawing as a testbed for grounded goal-driven communication. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 6495–6513. Association for Computational Linguistics.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *Int. J. Comput. Vis.*, 123(1):32–73.

Hyunhee Lee, Gyeongmin Kim, Yuna Hur, and Heuiseok Lim. 2021. Visual thinking of neural networks: Interactive text to image synthesis. *IEEE Access*, 9:64510–64523.

Tao Li, Gang Li, Jingjie Zheng, Purple Wang, and Yang Li. 2024. MUG: interactive multimodal grounding on user interfaces. In *Findings of the Association for Computational Linguistics: EACL 2024, St. Julian's, Malta, March 17-22, 2024*, Findings of ACL, pages 231–251. Association for Computational Linguistics.

Xiaoyuan Li, Moxin Li, Wenjie Wang, Rui Men, Yichang Zhang, Fuli Feng, Dayiheng Liu, and Junyang Lin. 2025. Mathopeval: A fine-grained evaluation benchmark for visual operations of mllms in mathematical reasoning. *CoRR*, abs/2507.18140.

Yujie Lu, Xianjun Yang, Xiujun Li, Xin Eric Wang, and William Yang Wang. 2023. Llmscore: unveiling the power of large language models in text-to-image synthesis evaluation. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Shichao Ma, Yunhe Guo, Jiahao Su, Qihe Huang, Zhengyang Zhou, and Yang Wang. 2026. Talk2Image: A multi-agent system for multi-turn image generation and editing. In *Proceedings of the Fortieth AAAI Conference on Artificial Intelligence*. AAAI Press.

Shichao Ma, Xinfeng Zhang, Zeng Zhao, Bai Liu, Changjie Fan, and Zhipeng Hu. 2025a. DialogDraw:

Image generation and editing system based on multi-turn dialogue. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence*. AAAI Press.

Shichao Ma, Xinfeng Zhang, Zeng Zhao, Bai Liu, Changjie Fan, and Zhipeng Hu. 2025b. Dialogdraw: Image generation and editing system based on multi-turn dialogue. In *Thirty-Ninth AAAI Conference on Artificial Intelligence AAAI 2025, Philadelphia, PA, USA, February 25 - March 4, 2025*, pages 24795–24803. AAAI Press.

Brielen Madureira and David Schlangen. 2023. Instruction clarification requests in multimodal collaborative dialogue games: Tasks, and an analysis of the codraw dataset. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 2295–2311. Association for Computational Linguistics.

Raja Marjieh, Ilia Sucholutsky, Theodore R. Sumers, Nori Jacoby, and Thomas L. Griffiths. 2022. Predicting human similarity judgments using large language models. In *Proceedings of the 44th Annual Meeting of the Cognitive Science Society, CogSci 2022, Toronto, ON, Canada, July 27-30, 2022*. cognitivesciencesociety.org.

Raja Marjieh, Ilia Sucholutsky, Pol van Rijn, Nori Jacoby, and Thomas L Griffiths. 2024. Large language models predict human sensory judgments across six modalities. *Scientific Reports*, 14(1):21445.

Daniela Mihai and Jonathon S. Hare. 2021. Learning to draw: Emergent communication through sketching. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 7153–7166.

Biswesh Mohapatra. 2023. Conversational grounding in multimodal dialog systems. In *Proceedings of the 25th International Conference on Multimodal Interaction*, pages 706–710. ACM.

Shuwen Qiu, Sirui Xie, Lifeng Fan, Tao Gao, Jungseock Joo, Song-Chun Zhu, and Yixin Zhu. 2022. Emergent graphical conventions in a visual communication game. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Fei Shen, Chengyu Xie, Lihong Wang, Zhanyi Zhang, Xin Jiang, Xiaoyu Du, and Jinhui Tang. 2026. Imagagent: Orchestrating multi-turn image editing via constraint-aware planning and reflection. *CoRR*, abs/2603.29602.

Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. 2020. ALFRED: A Benchmark for Interpreting Grounded Instructions for Everyday Tasks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yanpeng Sun, Shan Zhang, Wei Tang, Aotian Chen, Piotr Koniusz, Kai Zou, Yuan Xue, and Anton van den Hengel. 2025. Math blind: Failures in diagram understanding undermine reasoning in mllms. *Preprint*, arXiv:2503.20745.

Alberto Testoni and Raquel Fernández. 2024. Asking the right question at the right time: Human and model uncertainty guidance to ask clarification questions. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2024 - Volume 1: Long Papers, St. Julian's, Malta, March 17-22, 2024*, pages 258–275. Association for Computational Linguistics.

Ubaid Ullah, Jeong-Sik Lee, Chang-Hyeon An, Hyeonjin Lee, Su-Yeong Park, Rock-Hyun Baek, and Hyun-Chul Choi. 2022. A review of multi-modal learning from the text-guided visual processing viewpoint. *Sensors*, 22(18):6816.

Yael Vinker, Tamar Rott Shaham, Kristine Zheng, Alex Zhao, Judith E. Fan, and Antonio Torralba. 2025. Sketchagent: Language-driven sequential sketch generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2025, Nashville, TN, USA, June 11-15, 2025*, pages 23355–23368. Computer Vision Foundation / IEEE.

Jianhui Wang, Yangfan He, Yan Zhong, Xinyuan Song, Jiayi Su, Yuheng Feng, Ruoyu Wang, Hongyang He, Wenyu Zhu, Xinhang Yuan, Miao Zhang, Keqin Li, Jiaqi Chen, Tianyu Shi, and Xueqian Wang. 2025. Twin co-adaptive dialogue for progressive image generation. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 3645–3653. ACM.

Zhenyu Wang, Aoxue Li, Zhenguo Li, and Xihui Liu. 2024a. Genartist: Multimodal LLM as an agent for unified image generation and editing. In *Advances in Neural Information Processing Systems 37: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Zhijie Wang, Yuheng Huang, Da Song, Lei Ma, and Tianyi Zhang. 2024b. PromptCharm: Text-to-image generation through multi-modal prompting and refinement. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 1–16. ACM.

Jingxuan Wei, Caijun Jia, Xi Bai, Xinglong Xu, Siyuan Li, Linzhuang Sun, Bihui Yu, Conghui He, Lijun Wu, and Cheng Tan. 2026. Ggbench: A geometric generative reasoning benchmark for unified multimodal models. *Preprint*, arXiv:2511.11134.

Jingxuan Wei, Shiyu Wu, Xin Jiang, and Yequan Wang. 2023. Dialogpaint: A dialog-based image editing model. *CoRR*, abs/2303.10073.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, and Binyuan Hui et al. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Mingde Yao, Zhiyuan You, King-Man Tam, Menglu Wang, and Tianfan Xue. 2026. Photoagent: Agentic photo editing with exploratory visual aesthetic planning. *CoRR*, abs/2602.22809.

Ruijie Ye, Jiayi Zhang, Zhuoxin Liu, Zihao Zhu, Siyuan Yang, Li Li, Tianfu Fu, Franck Dernoncourt, Yue Zhao, Jiacheng Zhu, Ryan Rossi, Wenhao Chai, and Zhengzhong Tu. 2026. Agent banana: High-fidelity image editing with agentic thinking and tooling. *CoRR*, abs/2602.09084.

Xiang Yue, Yuansheng Ni, Tianyu Zheng, Kai Zhang, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, and 3 others. 2024. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 9556–9567. IEEE.

Chao Zhang, Cheng Yao, Jianhui Liu, Zili Zhou, Weilin Zhang, Lijuan Liu, Fangtian Ying, Yijun Zhao, and Guanyun Wang. 2021. Storydrawer: A co-creative agent supporting children's storytelling through collaborative drawing. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI EA '21, New York, NY, USA. Association for Computing Machinery.

Victor Zhong, Austin W. Hanjie, Sida I. Wang, Karthik Narasimhan, and Luke Zettlemoyer. 2021. Silg: the multi-environment symbolic interactive language grounding benchmark. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA. Curran Associates Inc.

A Human Annotation Guidelines

Human annotations were collected for both evaluation tasks (Section 4.3) on the GPT-5.2 + GPT-IMAGE-1.5 model pair (140 episodes). Three annotators with backgrounds in Computer Science rated each episode independently; all 140 episodes were covered by all three annotators.

Task 1 — Pairwise Similarity Rating

Annotators saw the target image I^* alongside each Generator rendering and rated visual and semantic similarity on a 0–10 integer scale. Figure 11 shows the interface.

Task 2 — Triplet Preference

Annotators saw the target image alongside the turn-0 and final-turn renderings and selected which was more similar to the target. Figure 12 shows the interface.

B Additional Results

Per-category similarity scores

Table 3 reports mean similarity scores for every category at both difficulty levels across all four judges.

Iterative payoff and triplet preference

Figure 15 shows Δ_Q per model pair and judge: GEMINI-3.1 pairings yield consistent positive gains while GPT-IMAGE-1.5 pairings show near-zero or negative deltas. Figure 16 corroborates this via Task-2 preference rates, with GEMINI-3.1 pairings consistently above the 50% chance baseline and GPT-IMAGE-1.5 pairings at or below it.

Correction language analysis

Correction utterances (turns $\geq$ 2) were labelled across ten categories using stem patterns (`\b<stem>\w{0,6}\b`) and phrase patterns for multi-word expressions; a category is flagged if any pattern fires. Tables 4 and 5 appear together on the following page.

Figure 13 shows per-category usage rates across turns. GPT-5.2 pairs sustain broad-coverage profiles throughout all ten turns; QWEN3-VL-30B pairs show narrowing toward COLOR dominance at later turns, partly reflecting the small number of episodes that reach them. Figure 14 shows the top-10 keyword tokens per category: the sharpest contrasts are in NUMERIC (coordinate notation absent in Qwen3), LABEL/TEXT (structural references absent in Qwen3), and POSITIVE ACKNOWLEDGEMENT (zero counts for Qwen3).

Effect of describer token budget

Scores increase monotonically with budget (Figure 17), confirming that longer descriptions yield richer first renderings and higher absolute quality at episode end. Triplet preference for the final-turn image decreases monotonically (Figure 18): at 10 tokens the gap between I_1 and I_T is large and visually salient; at 200 tokens I_1 is already detailed, leaving little room for visible improvement.

Score trajectories by category

Figure 19 shows mean similarity score per turn for every category and model pair: GPT-5.2 pairings improve steadily across turns; QWEN3-VL-30B pairings plateau after turn 1.

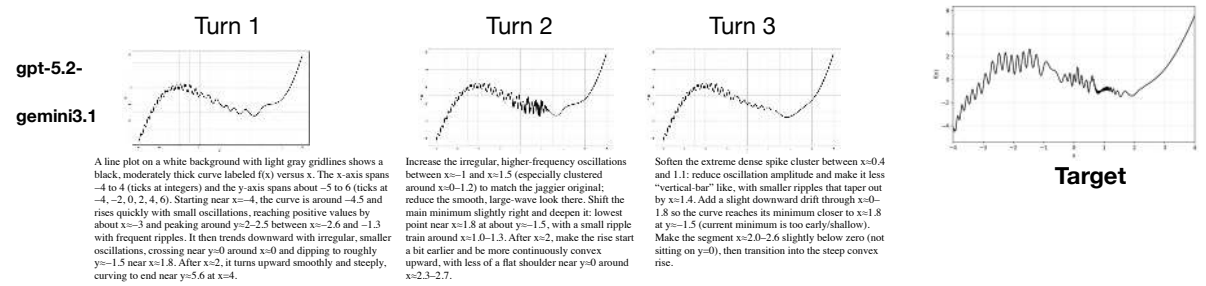


Figure 9: Qualitative samples for *Functions* (hard, instance#2) four model pairs and including token budget experiments

C Prompt Templates

The prompt templates are given in Figure 20, 21. The prompt template for generating synthetic images is given in Figure 22. We used *GPT-5.2* for generating the synthetic images.

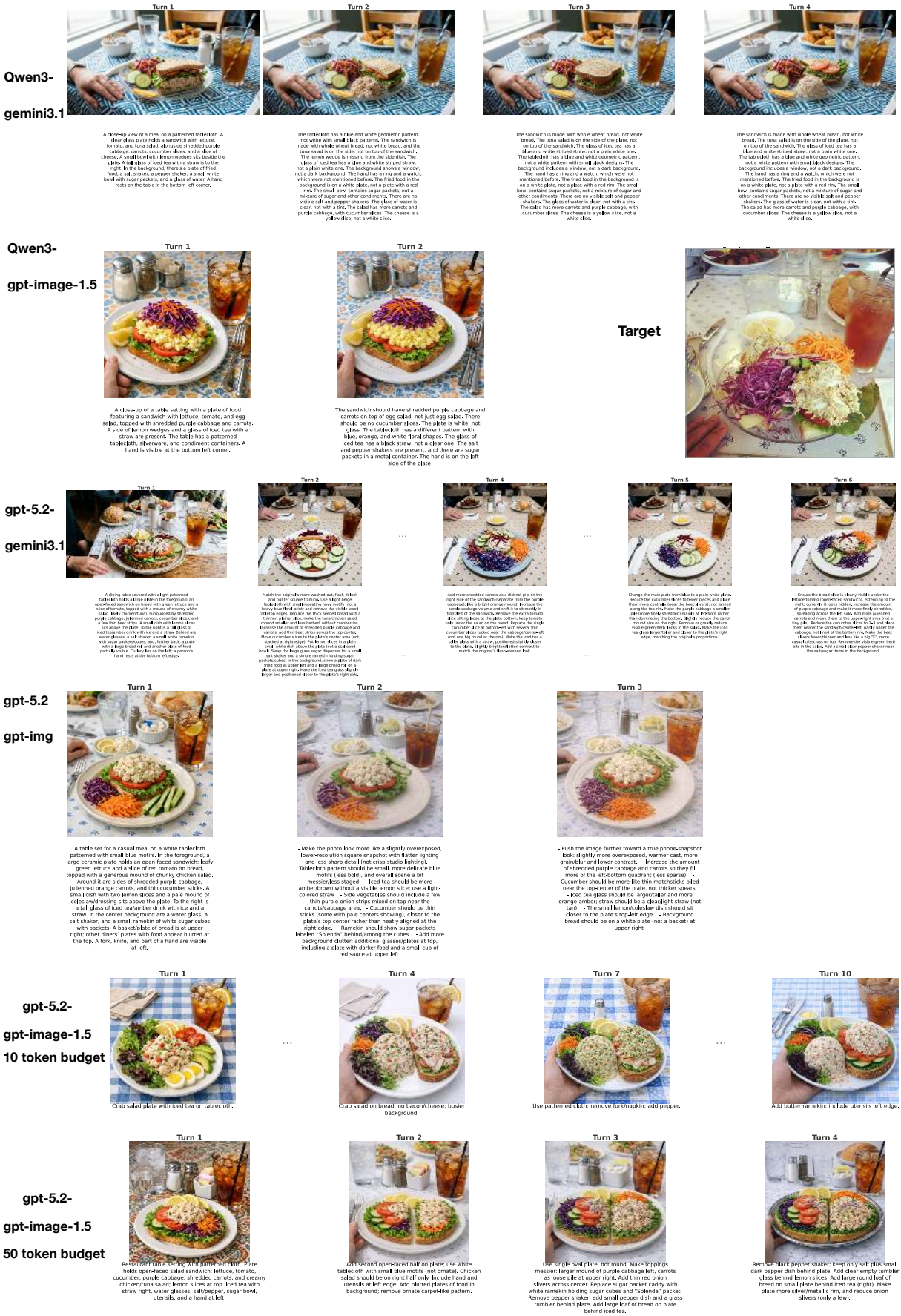


Figure 10: Qualitative samples for *Visual Genome* (hard, instance#4) four model pairs and including token budget experiments

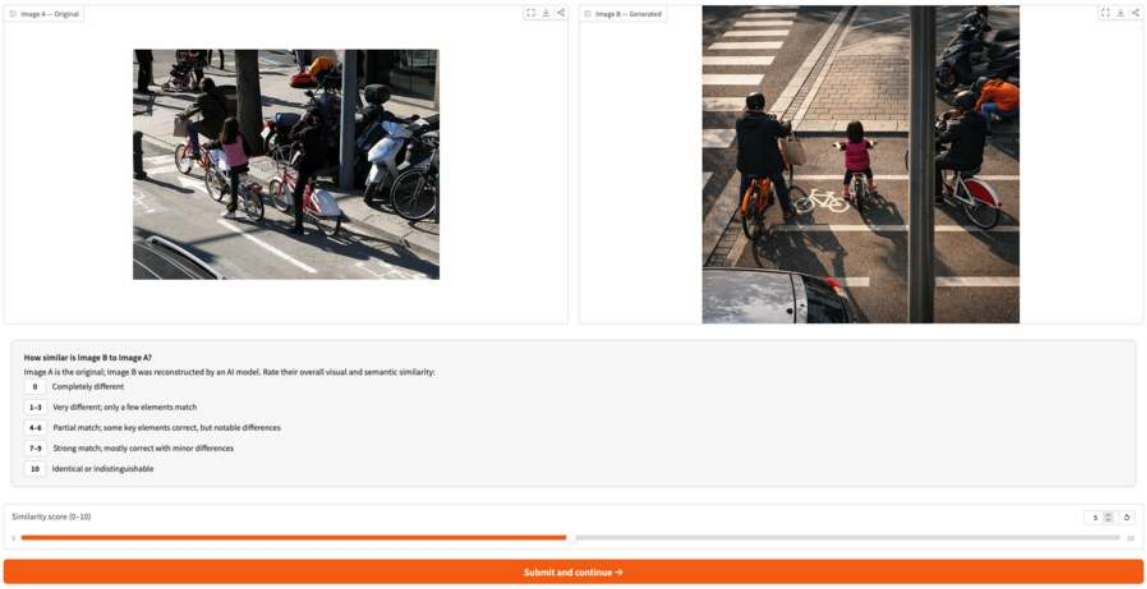


Figure 11: Task 1 interface: target (*Image A*) and Generator rendering (*Image B*) side by side; anchor labels calibrate the 0–10 slider.

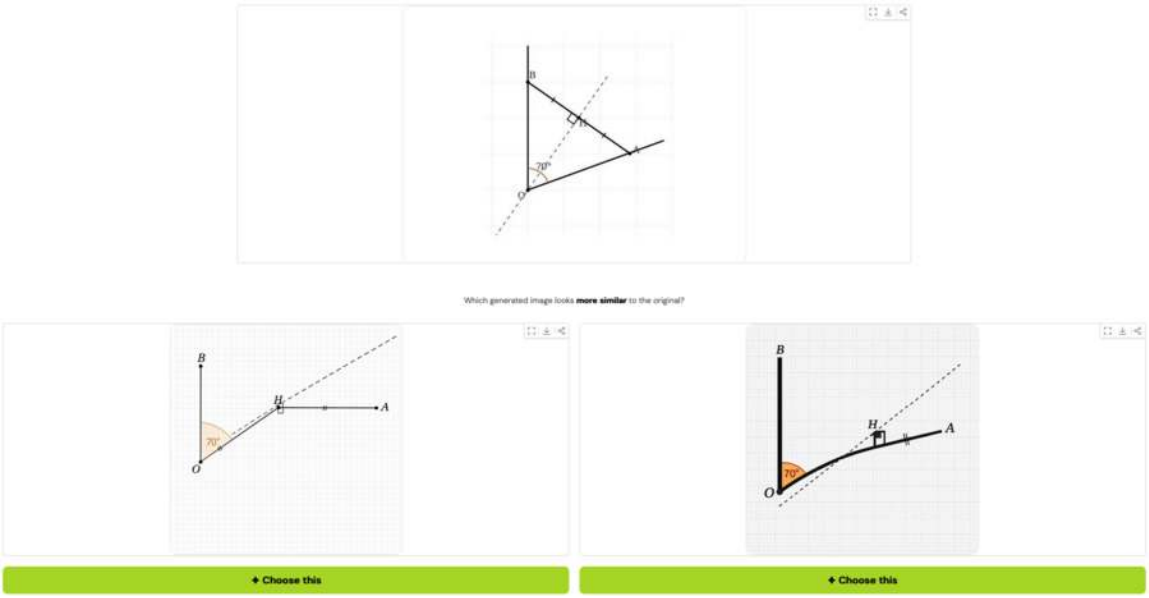


Figure 12: Task 2 interface: annotators choose which of two renderings (first vs. final turn) is more similar to the target.

Table 3: Mean final-turn similarity score (1–10) per model pair, image category, and difficulty level, broken down by judge. LipSim reports % of instances where the final-turn image was preferred over turn-0 (triplet task). Human annotations are available for the gpt-5.2 / gpt-image-1.5 pair only; '–' indicates no annotation.

Model Pair	Category	Diff.	GPT	Claude	Qwen3	LipSim (%)	Human
Qwen-3 / gemini-3.1	Bar Graph	easy	8.70	8.70	9.00	—	–
	Bar Graph	hard	8.50	8.10	8.80	0.0%	–
	Functions	easy	6.10	6.70	7.40	100.0%	–
	Functions	hard	5.20	5.60	6.00	40.0%	–
	Geometry	easy	7.20	6.40	6.50	40.0%	–
	Geometry	hard	5.20	5.00	5.10	33.3%	–
	Line Graph	easy	8.00	7.90	8.50	50.0%	–
	Line Graph	hard	6.90	6.90	7.10	0.0%	–
	Pie Graph	easy	7.90	7.90	8.90	66.7%	–
	Pie Graph	hard	8.10	8.30	8.40	33.3%	–
	Simple Shapes	easy	6.50	5.90	6.70	66.7%	–
	Simple Shapes	hard	7.00	6.30	7.50	33.3%	–
	Visual Genome	easy	6.40	6.70	7.30	55.6%	–
	Visual Genome	hard	6.60	6.70	6.50	44.4%	–
Qwen-3 / gpt-image-1.5	Bar Graph	easy	7.70	7.60	8.80	—	–
	Bar Graph	hard	7.60	7.10	8.20	—	–
	Functions	easy	4.90	4.40	6.20	100.0%	–
	Functions	hard	4.80	5.90	7.40	100.0%	–
	Geometry	easy	6.30	5.90	5.70	0.0%	–
	Geometry	hard	5.20	5.10	5.30	100.0%	–
	Line Graph	easy	7.10	7.50	8.00	—	–
	Line Graph	hard	6.70	7.20	7.20	100.0%	–
	Pie Graph	easy	8.20	8.10	8.40	0.0%	–
	Pie Graph	hard	8.20	8.00	7.90	0.0%	–
	Simple Shapes	easy	6.80	6.60	7.10	40.0%	–
	Simple Shapes	hard	6.60	6.60	6.50	16.7%	–
	Visual Genome	easy	7.20	7.00	8.10	0.0%	–
	Visual Genome	hard	6.30	6.30	6.70	40.0%	–
gpt-5.2 / gemini-3.1	Bar Graph	easy	8.90	8.80	9.20	66.7%	–
	Bar Graph	hard	8.80	8.60	9.20	100.0%	–
	Functions	easy	7.30	7.00	7.20	88.9%	–
	Functions	hard	6.50	7.20	6.70	66.7%	–
	Geometry	easy	8.70	7.70	8.80	87.5%	–
	Geometry	hard	7.90	6.50	7.50	88.9%	–
	Line Graph	easy	8.50	8.00	9.10	83.3%	–
	Line Graph	hard	8.00	8.10	8.60	88.9%	–
	Pie Graph	easy	8.80	8.70	9.20	87.5%	–
	Pie Graph	hard	8.80	8.60	8.80	90.0%	–
	Simple Shapes	easy	7.60	6.60	8.00	50.0%	–
	Simple Shapes	hard	7.70	7.00	7.80	40.0%	–
	Visual Genome	easy	7.80	7.40	8.00	40.0%	–
	Visual Genome	hard	8.00	7.30	8.00	80.0%	–
gpt-5.2 / gpt-image-1.5	Bar Graph	easy	7.90	7.40	8.60	40.0%	6.00
	Bar Graph	hard	7.40	7.00	8.50	40.0%	5.17
	Functions	easy	6.10	5.90	7.20	80.0%	5.10
	Functions	hard	4.80	5.20	6.80	70.0%	5.33
	Geometry	easy	7.50	6.80	7.10	80.0%	5.63
	Geometry	hard	6.60	5.70	6.20	100.0%	4.37
	Line Graph	easy	6.60	6.70	8.20	80.0%	6.37
	Line Graph	hard	6.00	6.40	6.90	80.0%	5.40
	Pie Graph	easy	8.30	8.00	8.40	70.0%	6.93
	Pie Graph	hard	7.60	7.30	7.80	30.0%	5.97
	Simple Shapes	easy	7.50	6.30	7.10	42.9%	6.40
	Simple Shapes	hard	7.50	6.80	7.20	37.5%	7.17
	Visual Genome	easy	7.90	7.10	7.90	40.0%	7.47
	Visual Genome	hard	7.60	7.10	7.70	55.6%	6.83

Table 4: Human inter-annotator agreement across both tasks (140 episodes, GPT-5.2 / GPT-IMAGE-1.5 pair). Task 1: Pearson r and Spearman ρ on 0–10 similarity scores. Task 2: agreement rate and Cohen's κ on binary triplet choices.

	Task 1		**Task 2**	
Pair	r	ρ	Agree. (%)	κ
A vs. B	0.421	0.447	63.6	0.247
A vs. C	0.562	0.546	65.7	0.308
B vs. C	0.454	0.465	66.4	0.316
Mean	*0.479*	*0.486*	*65.2*	*0.290*

Table 5: Correction language categories with representative stem roots and phrase patterns. Stems are matched as `\b<stem>\w{0,6}\b` to cover inflected forms; a category is flagged if any pattern fires.

Category	Representative stem roots	Representative phrase patterns
Position	*mov, shift, left, right, top, bottom, center, centr, above, below, higher, lower, align, overlap, corner, diagonal, horizon, vertic, perpendicular, parallel*	"move to the left/right/top/bottom"; "upper-left"; "top-right corner"; "to the left of"
Size/Scale	*enlarg, larg, bigg, small, shrink, reduc, widen, narrow, thin, thick, tall, shorter, resize, rescal, upscal, downscal*	"too big/small/wide/narrow"; "scale up/down"; "make it larger/smaller"; "slightly narrower"
Color	named colours (*red, blue, green, orange, purple, yellow, cyan, teal, gray, black, white, . . .*); *darker, lighter, brighter, saturate, desaturat, muted, vivid, pale*	"color/colour"; "shade of"; "hue"; "more saturated/vivid"; "make it darker/lighter"
Add/Remove	*add, remov, delet, insert, includ, exclud, omit, missing, extra, eliminat, replac, substitut, absent, introduc, eras, drop*	"should not be there"; "add/insert a"; "remove/delete the"; "no longer visible"
Shape	*round, circular, ellips, oval, squar, rectangl, triangl, polygon, diamond, arc, curv, straight, angl, vertex, wedge, convex, concave, symmet, wavy*	"should be a circle/square/triangle"; "more rounded/pointed"; "right-angle"
Label/Text	*label, mislabel, title, annot, font, legend, caption, tick, axis, note, header, bold, italic, underlin, misalign, misplac*	"x/y-axis"; "axis label/tick/title"; "missing/wrong label"; "font size/weight/style"
Style	*gradient, shadow, blur, smooth, texture, flat, solid, dashed, dotted, stroke, opaci, transparent, fill, outlin, border, grid, background, blend*	"flat color/fill"; "solid/dashed/dotted line"; "line weight/thickness"; "stroke width"
Comparison	*incorrect, inaccurat, erroneous, mismatch, discrepancy, differ, wrong, misrepresent*	"instead of"; "rather than"; "should be"; "doesn't match"; "compared to the original"
Numeric	*approximat, exactl, precisel, percent, proportion*	digits with units (%, px, pt, degrees); coordinate notation (x=3.2, y≈1); "about/around/exactly *N*"
Positive Ack.	*correct, accurat, almost, nearly, well*	"close/similar enough"; "otherwise correct/good"; "other than that"; "now correct"

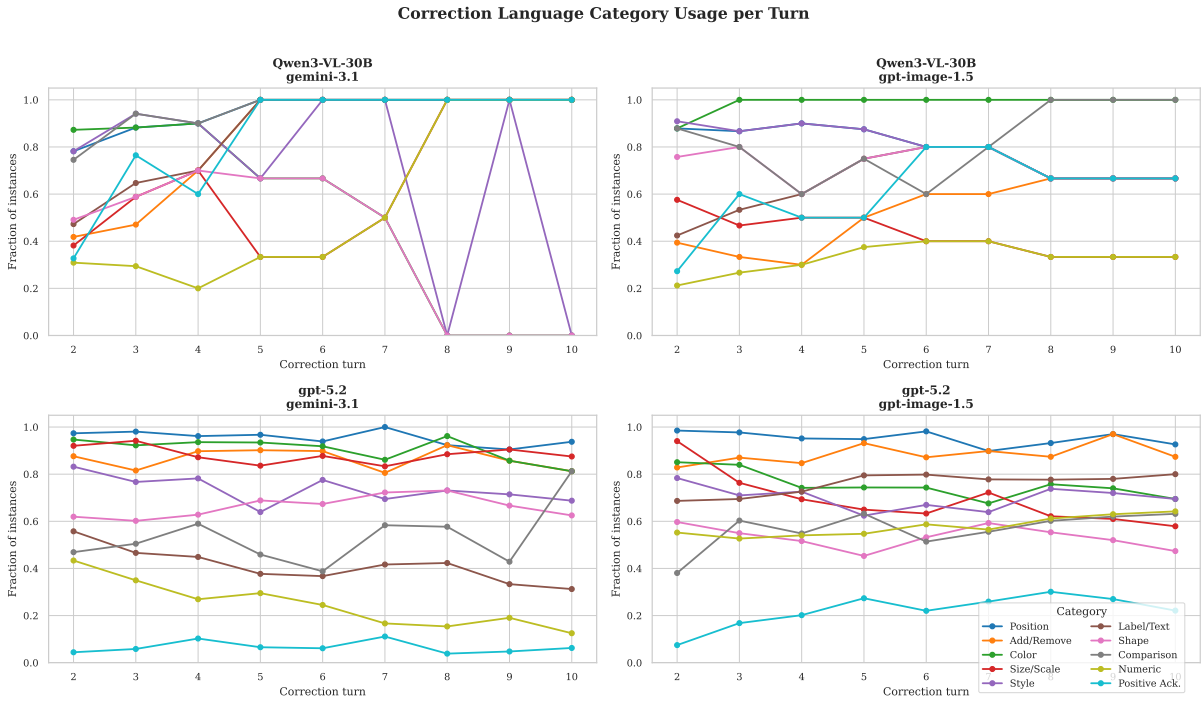


Figure 13: Correction language category usage per turn for each model pair. GPT-5.2 pairs (bottom row) sustain consistent multi-category coverage; QWEN3-VL-30B pairs (top row) show higher variance and sparse later-turn coverage.

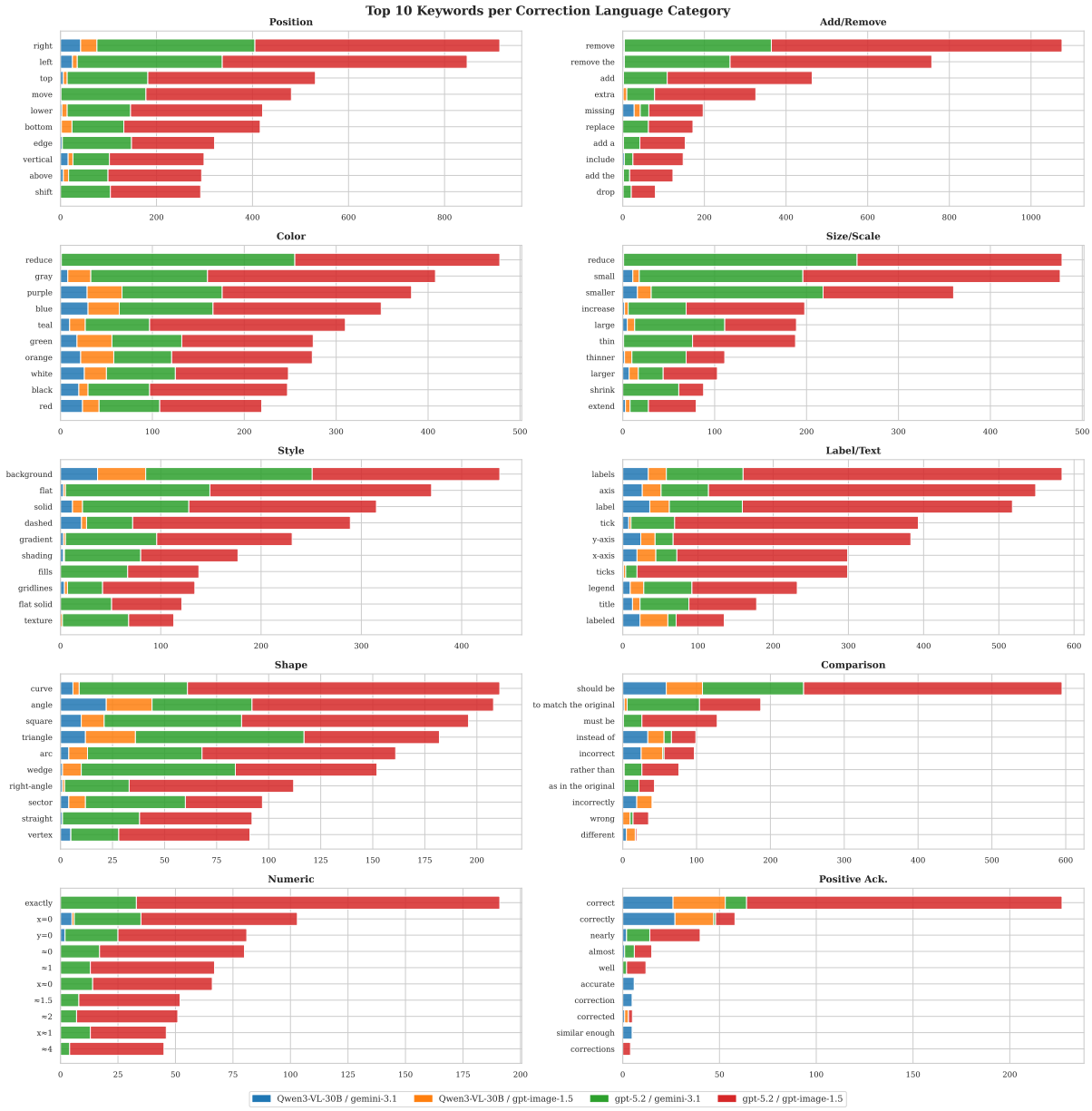


Figure 14: Top-10 matched keyword tokens per correction language category by model pair. GPT-5.2 pairings dominate NUMERIC, LABEL/TEXT, and POSITIVE ACK.; Qwen3-VL-30B pairings lead in COLOR and SHAPE and are absent from NUMERIC and POSITIVE ACK.

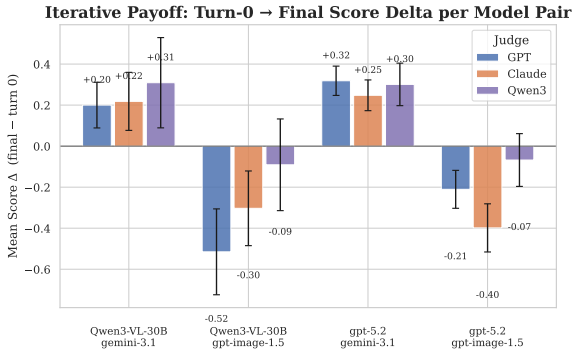


Figure 15: Task-1: Mean iterative payoff $\Delta_Q = \text{score}_{I_T} - \text{score}_{I_1}$ per model pair and judge (error bars: SEM). The dashed line marks the no-improvement baseline.

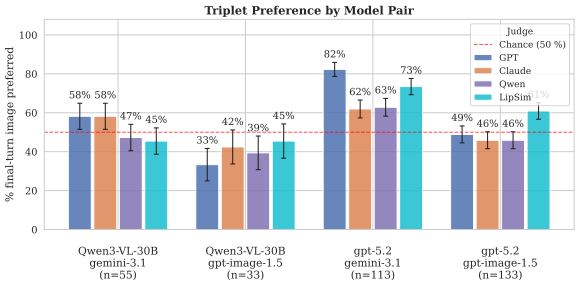


Figure 16: Task-2: Percentage of multi-turn episodes where the final-turn image is preferred over the first-turn image, by model pair and judge. The red dashed line marks the 50% chance baseline.

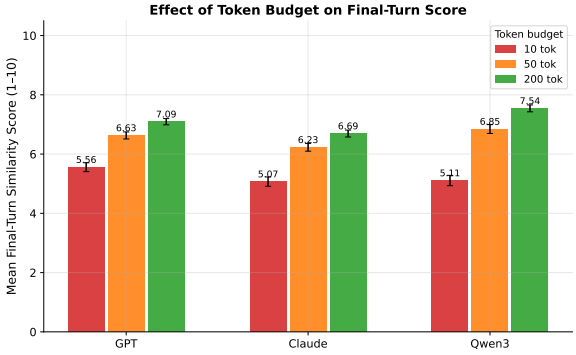


Figure 17: Mean final-turn similarity score for GPT-5.2 + GPT-IMAGE-1.5 at three token budgets, by judge (error bars: SEM). Absolute quality increases monotonically with budget.

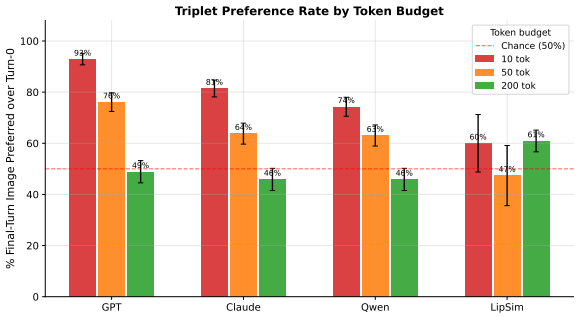


Figure 18: Triplet preference rate for GPT-5.2 + GPT-IMAGE-1.5 at three token budgets, by judge and LipSim. Preference for the final-turn image decreases with budget; all conditions remain above the 50% chance baseline (dashed line).

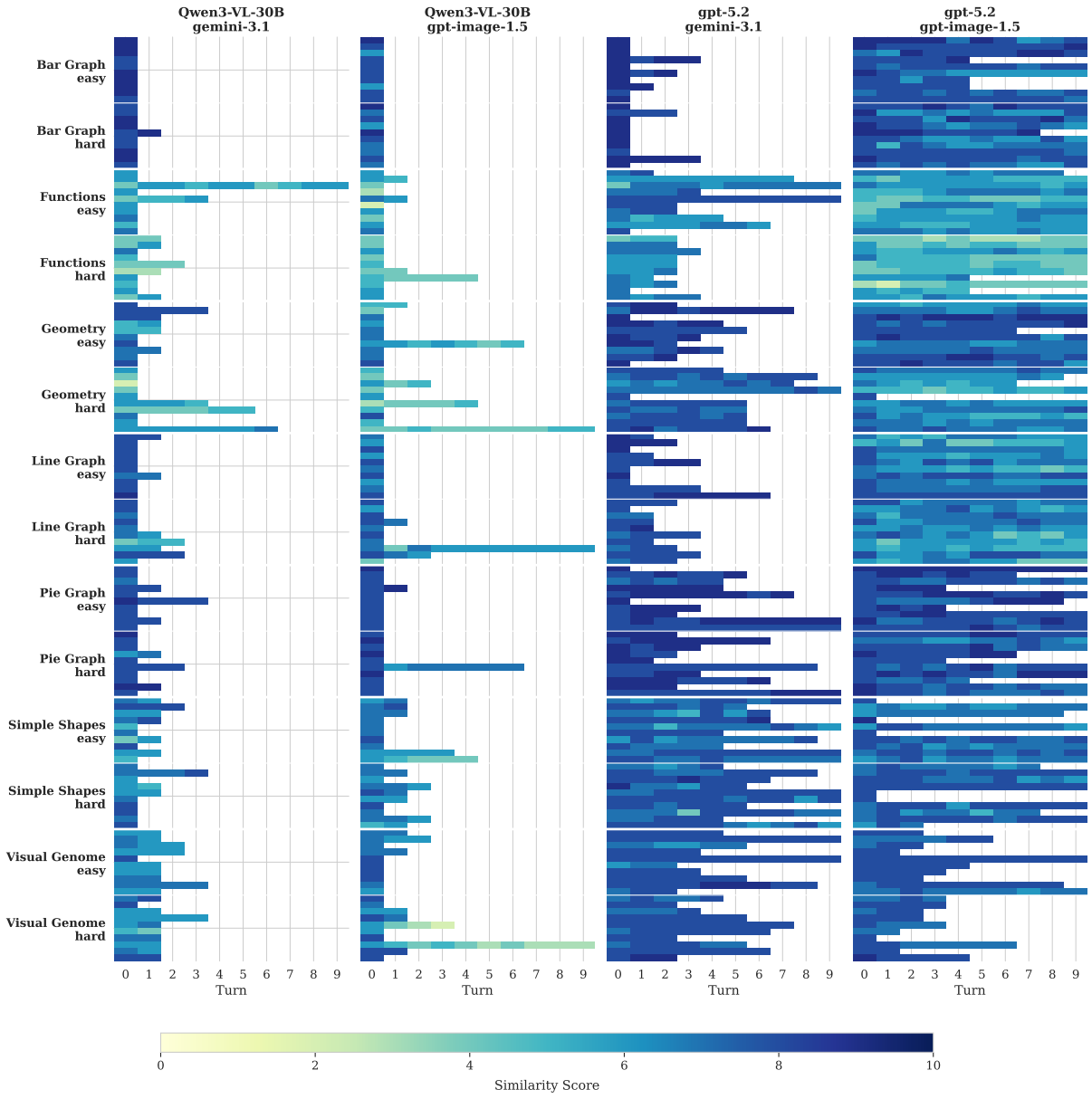


Figure 19: Score trajectory heatmap: mean similarity score per turn, image category, and describer–generator pair. Rows are categories; columns are turns; colour encodes mean similarity. The turn scores are assigned by GPT-5.2.

TEMPLATE C.0.1

```
Let's play a game. I will feed you an image and you will produce the most accurate description possible. The description should capture every detail and be extremely precise. Your description must be at most 200 tokens long.

The output should consist of an initial tag <DESCRIPTION> followed by the description.

Reply just with the two tags and their related content.
```

TEMPLATE C.0.2

```
You will now receive an image that was generated using your initial description. The goal is to generate an image as similar as possible to the original one. If you deem that the generated image is close enough to the original one, reply with "<DONE>".

If not, you will reply with the changes needed to correct the image. The updated description should ONLY contain the changes/edits that have to be made to the previous image to make it look like the original image. Do NOT provide the full description again. Your description must be at most 200 tokens long. The output should consist of an initial tag <DESCRIPTION> ,followed by the changes or just a tag word <DONE> if you deem that the images are similar enough. Reply just with the two tags and the content related.

Then these steps will repeat until you find the generated image similar enough to the original image.

<GENERATED_IMAGE>
```

Figure 20: Prompt templates for Describer player. The first template is for the initial turn and the next template is for consecutive turns.

TEMPLATE C.0.3

```
<DESCRIPTION WHAT TO GENERATE>
```

TEMPLATE C.0.4

```
<DESCRIPTION WHAT TO GENERATE>

<PREVIOUS_GENERATED_IMAGE>
```

Figure 21: Prompt templates for Generator player. The first template is for the initial turn and the next template is for consecutive turns.

TEMPLATE C.0.5
Generate a description for a simple geometry diagram.

Requirements:

- The diagram should be completely different from the ones generated early,vary the graphical settings and colours as well.
- The diagram described should be similar to ones found in geometry book
- The diagram should be simple and easily understandable

STRICT OUTPUT FORMAT REQUIREMENT (MANDATORY):
Output ONLY:

<DESCRIPTION> (technical construction details only) </DESCRIPTION>
<SUMMARY> (exactly two lines describing the resulting diagram) </SUMMARY>

TEMPLATE C.0.6
When provided with an image description, generate the Python code required to visualize the image.

Requirements:

- Output should consist only of the required code, placed between two <CODE> tags.
- Do not display the plot using plt.show();
- Do not include any title or legend in the picture.
- !IMPORTANT! All figures should fit entirely in the image, adjust sizes if necessary
- !IMPORTANT! THE CODE MUST CONSISTS OF A FUNCTION NAMED "GENERATE" THAT RETURNS THE PLOT OBJECT WHEN CALLED AND THAT TAKES NO INPUT
- !IMPORTANT! code will be run using the exec command so make sure to include all required imports
- !IMPORTANT! DO NOT use ax or subplots
- !IMPORTANT: THE CODE MUST BE BETWEEN TWO <CODE TAGS> as in the following example

  <CODE> (code goes here) </CODE>

Figure 22: Prompts used to generate synthetic data for the experiment using GPT-5.2. The first prompt produces the diagrams descriptions (here is shown an example for the easy geometry experiment), the second one produces the code to generate images with Matplotlib. The first prompt was sent to the LLM alongside with all the previously generated images summaries.