CONSIDERING THE FOREST AND EVERY LEAF: VARIOUS INTERLEAVED VISUAL INPUTS FOR ABSTRACTIVE ANALYSIS

Anonymous authors
Paper under double-blind review

ABSTRACT

Abstractive visual reasoning (AVR) requires multi-modal large language models (MLLMs) to interpret human-defined visual representations, such as mind maps and relational diagrams. Existing AVR benchmarks primarily study individual abstract images, leaving open how models connect concrete visual evidence with abstract relational structures distributed across multiple images. We introduce VIVIANA, a benchmark for reasoning over sequences composed of entity images, visualized knowledge triples, and subgraph diagrams. VIVIANA contains 22,568 instances across 14 tasks, ranging from visual grounding to cross-image relational and graph reasoning. We also construct local-to-global chain-of-thought supervision and propose Gated Knowledge-informed GRPO (GKGRPO), which combines answer rewards with local and global knowledge rewards through an answer-conditioned, two-level gate. On Qwen3-VL models, GKGRPO achieves the highest overall score among the evaluated post-training methods. The resulting models also improve on established multi-image understanding and abstract visual reasoning benchmarks, indicating that structured cross-image supervision can transfer beyond the visual knowledge graph setting.

1 INTRODUCTION

Multi-modal large language models (MLLMs) have made rapid progress in understanding concrete images of people, objects, and scenes. This ability captures only one part of visual intelligence. **Abstractive visual reasoning (AVR)** (Zhang et al., 2026a) instead studies images whose semantics are encoded through human-defined representational conventions.

Such representations include geometric diagrams (Zhang et al., 2024), statistical charts (Masry et al., 2022), and mind maps (Zhang et al., 2025). Unlike natural images, they express meaning through symbols, layout, and explicit relations. Reasoning over them therefore requires a model to recover both the visual language and the content it encodes. Mind maps are particularly useful for studying this capability because they connect multi-modal perception with structured knowledge reasoning. Existing work (Zhang et al., 2025; 2026a) has advanced benchmark construction, synthetic data generation, and post-training for individual relational diagrams. However, many practical reasoning problems distribute evidence across several images: a model may need to recognize an entity in a photograph, align it with a visualized fact, and then follow relations in a larger subgraph. This setting requires both image-wise grounding and cross-image composition.

Table 1 summarizes the closest benchmarks. MathVerse (Zhang et al., 2024) studies mathematical diagrams, VisuLogic (Xu et al., 2025) studies visual logic puzzles, and M3STR (Zhang et al., 2025) and STAR (Zhang et al., 2026a) study structured relational knowledge images. QUASAR (Zhang et al., 2026b) extends AVR to compositional queries over multiple abstract inputs, whereas MMIU (Meng et al., 2025) evaluates broad multi-image understanding. In contrast, we focus on a progression from same-granularity inputs to sequences that explicitly connect concrete entity images, visualized facts, and abstract subgraphs, with training supervision for this setting.

We address this setting with VIVIANA, short for Various Interleaved Visual Inputs for Abstractive Analysis. It contains 22,568 instances organized into 6 categories and 14 tasks. The inputs cover three semantic granularities: concrete entity images, visualized knowledge triples, and subgraph di-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Table 1: Comparison with representative visual reasoning benchmarks. "Novel post-training" indicates that the work introduces a new optimization objective, reward mechanism, or learning algorithm, rather than directly applying existing training methods.

Benchmark	Visual Form	Multi-image	Cross-image Reasoning	Concrete–Abstract Interleaving	Structured Knowledge	CoT-supervised Training Data	Novel Post-training
MathVerse (Zhang et al., 2024)	Math diagrams	✗	✗	✗	✗	✗	✗
M3STR (Zhang et al., 2025)	Mind maps	✗	✗	✗	✓	✗	✗
STAR (Zhang et al., 2026a)	Relational knowledge images	✗	✗	✗	✓	✓	✓
QUASAR (Zhang et al., 2026b)	Complex visual queries	✓	✓	✗	✓	✓	✗
VisuLogic (Xu et al., 2025)	Visual logic puzzles	✗	✗	✗	✗	✗	✗
MMIU (Meng et al., 2025)	Diverse visual inputs	✓	✓	✗	✗	✗	✗
VIVIANA (Ours)	**Entities, facts, and subgraphs**	✓	✓	✓	✓	✓	✓

agrams. Some tasks isolate a single granularity, while others combine granularities to test grounding and relational composition. We construct the data from multi-modal knowledge graphs (MMKGs) through image filtering, graph sampling and visualization, task-specific composition, and local-to-global chain-of-thought (CoT) synthesis. We further introduce **gated knowledge-informed GRPO (GKGRPO)** for post-training. Standard GRPO optimizes outcome rewards without directly assessing the reasoning process. KGRPO adds keyword-based local and global knowledge rewards, but these recall-oriented signals may also reward irrelevant keyword insertion. GKGRPO conditions the weight of the knowledge reward on answer correctness: fully correct trajectories receive a larger coefficient, while other trajectories retain only a smaller coefficient. Experiments on VIVIANA show the best overall results at both evaluated model scales, and evaluation on external benchmarks measures transfer to broader multi-image and abstract visual reasoning. Our contributions are three-fold:

- We introduce VIVIANA, a benchmark with 22,568 instances and 14 tasks that organizes visual reasoning across entity, fact, and subgraph granularities, with an emphasis on interleaved concrete and abstract visual inputs.
- We develop an MMKG-based construction pipeline that combines task-specific graph sampling and visualization with local-to-global CoT synthesis, providing structured supervision for multi-image grounding and relational reasoning.
- We propose GKGRPO, an answer-conditioned knowledge-reward method for post-training MLLMs, which provides better process rewards from the supervision of structured knowledge. It achieves the highest Overall score among the evaluated training methods at both model scales and transfers positively to the evaluated multi-image and abstract visual reasoning benchmarks.

2 RELATED WORKS

Abstractive Visual Reasoning for MLLMs. Abstractive visual reasoning (AVR) (Zhang et al., 2025) is currently a popular research topic for MLLMs, which focuses on enhancing and evaluating the reasoning capability on images with human-defined abstractive semantics, such as math (Zhang et al., 2024), charts (Masry et al., 2022), mind maps (Zhang et al., 2026a), and neuro-symbolic queries (Zhang et al., 2026b). These benchmarks typically use a single abstractive image as visual input, with the goal of having the model understand the abstract semantic information contained within it. In this work, we further combine abstractive and concrete images into sequences and explore ways to enhance the multi- and cross-image abstractive reasoning capabilities of MLLMs. Meanwhile, we attempt to achieve co-design for both benchmark data and post-training algorithms.

Multi-image Evaluation Benchmarks for MLLMs. Most existing MLLM benchmarks (Fu et al., 2024a) are in a single-image setting, which has just one visual input for each data instance. Multi-image benchmarks (Suhr et al., 2019; Jiang et al., 2024) provide multiple heterogeneous visual information for MLLMs to evaluate their comprehensive understanding and reasoning capabilities across multiple images, which are more challenging for MLLMs. MUIRBench (Wang et al., 2025) and MMIU (Meng et al., 2025) are two representative benchmarks with diverse data sources and subtasks. In this work, we also propose a new multi-image benchmark with mind maps as a special visual signal, which brings abstractive visual reasoning into an interleaved multi-image setting.

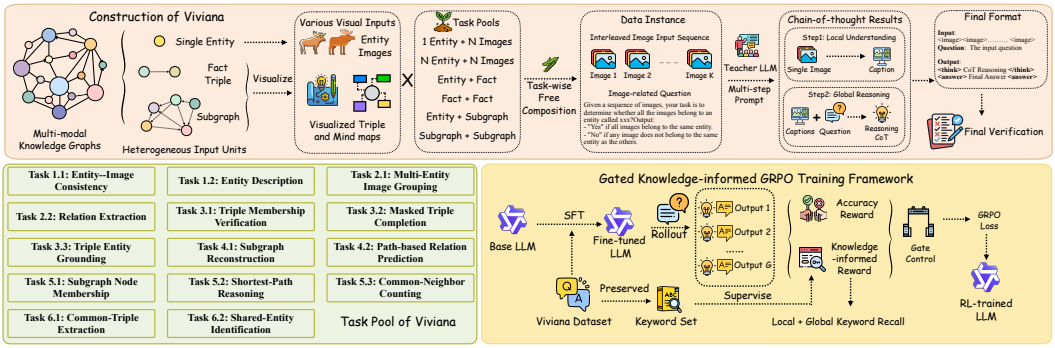


Figure 1: Overview of VIVIANA construction and GKGRPO post-training. Entity images, visualized triples, and subgraph diagrams are composed into task instances with local-to-global CoT supervision; answer-conditioned knowledge rewards are then used for post-training.

3 CONSTRUCTION PROCESS OF VIVIANA

3.1 DATASET OVERVIEW

Existing multi-image benchmarks (Meng et al., 2025) primarily use concrete images. VIVIANA complements this setting with **V**arious **I**nterleaved **VI**sual Inputs for Abstractive **ANA**lysis. Here, *interleaved* means that concrete entity images and abstract visual representations of facts or subgraphs are placed within a single ordered visual context, requiring a model to align evidence across representational forms rather than process each input in isolation. The task suite progresses from same-granularity multi-image understanding to cross-granularity sequences that foreground this interleaved reasoning capability.

Data Format and Data Source The i-th instance is $(\mathcal{S}_i, \mathcal{Q}_i, \mathcal{A}_i)$, where $\mathcal{S}_i = (\mathcal{V}_1, \ldots, \mathcal{V}_{N_i})$ is an ordered sequence of $N_i \geq 2$ concrete or synthetically rendered images, $\mathcal{Q}_i$ is the textual question, and $\mathcal{A}_i$ contains the CoT and final answer. In an interleaved instance, the elements of $\mathcal{S}_i$ may encode complementary evidence at different semantic granularities, and the question requires their joint interpretation. Following existing work, we use mind maps (Zhang et al., 2025) as the primary abstract representation. The underlying MMKGs are VisualSemantic (Alberts et al., 2020), FB15K-237 (Bollacker et al., 2008), and MKG-Y (Xu et al., 2022).

Task Design We define 14 tasks in 6 categories according to the semantic granularity of their inputs. An image may depict an entity, render one factual triple, or visualize a subgraph, corresponding to a node, an edge, and a subgraph in the MMKG. The six categories are multiple images of one entity, images of multiple entities, entity + fact, multiple facts, entity + subgraph, and subgraph + subgraph. The first categories test grounding within one granularity, whereas the cross-granularity categories require concrete–abstract alignment and relational composition. The definitions of all 14 tasks of VIVIANA are presented in Appendix A.1.1 .

Dataset Statistics The benchmark contains 22,568 instances and 89,788 image occurrences. Each instance contains 2–10 images. We randomly split instances into training, validation, and test sets at an approximate 8:1:1 ratio, yielding 18,054, 2,256, and 2,258 instances, respectively. Because this is an instance-level rather than entity-level split, entities, source images, or triples may occur in more than one split. The test set therefore measures generalization to new task instances rather than strictly unseen entities. Table 3 reports the per-task distribution.

3.2 CONSTRUCTION PIPELINE

We construct VIVIANA from a given MMKG in four stages: image preparation, task-wise composition, CoT synthesis, and final verification.

Image Preparation For an MMKG $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T})$, let $\mathcal{V}_e$ be the source-image set for entity $e \in \mathcal{E}$. Because source images vary in relevance and quality, we compute the CLIP similarity between each image and a text prompt containing the entity name, rank the images, and preferentially sample higher-scoring candidates. This procedure produces the entity-level visual inputs. Fact and subgraph images are rendered rather than taken from the source data. A fact is a triple (h, r, t), and a sampled subgraph is defined as:

$$\mathcal{G}_s = (\mathcal{E}_s, \mathcal{R}_s, \mathcal{T}_s), \quad \mathcal{E}_s \subseteq \mathcal{E},\ \mathcal{R}_s \subseteq \mathcal{R},\ \mathcal{T}_s = \{(h, r, t) \in \mathcal{T} \mid h \in \mathcal{E}_s \wedge t \in \mathcal{E}_s \wedge r \in \mathcal{R}_s\} \quad (1)$$

where $\mathcal{T}_s$ contains the retained triples among the selected entities and relations. We render $\mathcal{G}_s$ as a mind map or flowchart with the Graphviz API (Ellson et al., 2004). Nodes represent entities and may include a representative image; labeled directed edges represent relations. A triple image is the special case with two entities and one edge.

Task-wise Data Composition Each task uses a fixed prompt template and a task-specific input composition rule. Starting from a randomly sampled entity, we perform a mixed depth- and breadth-first traversal of neighboring nodes and retain the entities, triples, or subgraphs required by the task. Subgraphs contain at most eight entities. We render the selected structures, combine them with any required entity images, shuffle inputs when specified by the task, and derive the gold answer directly from the sampled graph data.

CoT Data Synthesis We synthesize a local-to-global CoT with Qwen3.6 as the teacher. First, the teacher produces an analytic caption for each image. It then receives the complete image sequence and local captions and generates cross-image reasoning leading to an answer. We compare the generated answer with the graph-derived gold answer and regenerate unsuccessful outputs. The released response places reasoning inside `<think>...</think>` and the final answer inside `<answer>...</answer>` to support deterministic parsing.

Final Verification Finally, we combine MLLM-based filtering with human review to remove instances containing politics-related topics, violent, sexually explicit, or discriminatory content. The resulting dataset contains 22,568 instances focused on general knowledge reasoning.

4 Training MLLMs with Gated Knowledge-informed GRPO

4.1 General Post-training Pipeline

Our post-training pipeline has two stages. We first perform supervised fine-tuning (SFT) on the training split and then optimize the resulting model with GRPO. To construct the second-stage corpus, the SFT model performs inference on the training set; every instance on which it fails to obtain the full task score is retained as a hard case. The vanilla GRPO objective can be denoted as:

$$\begin{aligned}\mathcal{J}_{\mathrm{GRPO}}(\theta) =& \mathbb{E}_{q,\{o_i\}_{i=1}^{G}\sim\pi_{\theta_{\mathrm{old}}}}\Bigg[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\min\Big(\rho_{i,t}(\theta)A_i,\ \mathrm{clip}\big(\rho_{i,t}(\theta),\\ & 1-\varepsilon, 1+\varepsilon\big)A_i\Big)\Bigg] - \beta\,\mathbb{D}_{\mathrm{KL}}(\pi_\theta \,\|\, \pi_{\mathrm{ref}}), \quad \rho_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_{\theta_{\mathrm{old}}}(o_{i,t} \mid q, o_{i,<t})}\end{aligned} \quad (2)$$

Here $\rho_{i,t}(\theta)$ is the importance-sampling ratio, ε is the clipping coefficient, and β controls the KL penalty against the reference model π_{ref}. A_i is the relative advantage of the i-th rollout response obtained by normalizing the corresponding reward r_i within the group. As for vanilla GRPO, we employ accuracy reward for r_i, determined by the output correctness in the `<answer>` mark. Given that the MLLM has fit the output format after SFT, we do not include a format reward in GRPO. Meanwhile, tasks in VIVIANA have different criterions for accuracy. We present the detailed of how the accuracy reward is calculated for each task in Appendix A.4.

4.2 Knowledge-informed Rewards for Viviana Tasks

An answer-only reward does not distinguish faithful reasoning from a rollout that reaches the answer through unsupported intermediate claims. Following KGRPO (Zhang et al., 2026a), we add a

knowledge-informed reward:

$$r_{\text{kgrpo}} = \alpha r_{\text{acc}} + (1-\alpha) r_{\text{know}}, \quad r_{\text{know}} = \frac{1}{2}(r_{\text{local}} + r_{\text{global}}) \tag{3}$$

The knowledge reward has two components: local understanding of individual images and global reasoning across images. Let $\mathcal{K}_i$ be the reference keyword set for image i and s_i its parsed local description. After lowercasing and whitespace normalization, a keyword matches if it occurs as a substring of the prediction. Image-level keyword recall is

$$r_{\text{local},i} = \frac{1}{|\mathcal{K}_i|} \sum_{k \in \mathcal{K}_i} \mathbb{I}[k \subseteq s_i], \quad r_{\text{local}} = \frac{1}{M} \sum_{i=1}^{M} r_{\text{local},i}. \tag{4}$$

where $\mathbb{I}[\cdot]$ is the indicator function and M is the number of valid image–description pairs. Let $\mathcal{K}_{\text{global}}$ contain task-relevant reference keywords and let s_{global} denote the parsed cross-image reasoning. The global reward uses the same recall criterion:

$$r_{\text{global}} = \frac{1}{|\mathcal{K}_{\text{global}}|} \sum_{k \in \mathcal{K}_{\text{global}}} \mathbb{I}[k \subseteq s_{\text{global}}]. \tag{5}$$

We set $r_{\text{know}} = \frac{1}{2}(r_{\text{local}} + r_{\text{global}})$. If either component cannot be parsed or its required keyword set is empty, the knowledge reward is zero. Task metadata changes the keyword sets but not the reward computation.

4.3 Control the Knowledge Reward with Hard Gate

Because the knowledge reward is recall-based, a model can increase it by inserting additional entity names without improving the final answer. We therefore extend KGRPO with an answer-conditioned hard gate. Here "hard" denotes a discrete choice between two fixed coefficients:

$$r_{\text{final}} = \begin{cases} r_{\text{acc}} + \beta_1 \, r_{\text{know}}, & \text{if } r_{\text{ans}} == 1, \\ r_{\text{acc}} + \beta_2 \, r_{\text{know}}, & \text{if } r_{\text{ans}} == 0. \end{cases} \tag{6}$$

We use $\beta_1 = 0.2$ and $\beta_2 = 0.05$. Thus, only a fully correct rollout receives the larger knowledge-reward coefficient; partially correct and incorrect rollouts use the smaller coefficient. Because accuracy is difficult to compute reliably for the open-ended outputs of T1.2 and T4.1, we use only r_{know} for these two tasks during RL training. This design reduces the incentive to optimize keyword recall without solving the task, although it does not eliminate all possible reward exploitation.

5 Experiments and Analysis

We evaluate post-training performance on VIVIANA, analyze data scaling and cross-task transfer, ablate the CoT and reward design, and measure transfer to established multi-image and abstract visual reasoning benchmarks.

5.1 Experiment Settings

Baseline Methods We compare GKGRPO with SFT; preference-optimization methods DPO (Rafailov et al., 2023), ORPO (Hong et al., 2024), and SimPO (Meng et al., 2024); and RL methods GRPO (Guo et al., 2025), GDPO (Liu et al., 2026), DAPO (Yu et al., 2025), GSPO (Zheng et al., 2025), and KGRPO (Zhang et al., 2026a). All post-training methods start from the same SFT checkpoint and use the same hard-case corpus under comparable training budgets. For preference optimization, the reference CoT and answer form the chosen response, and the model's incorrect rollout forms the rejected response.

Evaluation Metrics Task-specific exact-match and partial-credit rules are given in Appendix A.4. For the two open-ended tasks, T1.2 and T4.1, Qwen3.6 serves as a fixed judge and assigns a score from 0 to 100 based on factual correctness and completeness. All task scores are reported as percentages. The Overall column is the unweighted macro-average across the 14 tasks.

Table 2: Results on the VIVIANA tasks. Scores are percentages. Overall is the unweighted macro-average across tasks. Gold shading marks the top three results within each Qwen3-VL model.

Method	T1.1	T1.2	T2.1	T2.2	T3.1	T3.2	T3.3	T4.1	T4.2	T5.1	T5.2	T5.3	T6.1	T6.2	Overall
Closed-source Top-tier Models															
Kimi-k2.6	78.15	82.01	85.51	1.31	74.74	83.89	1.50	98.92	95.17	77.56	52.34	84.25	78.01	79.85	69.52
Claude-Opus-5	80.79	82.48	89.13	0.00	80.89	85.23	25.56	99.43	95.17	79.21	74.22	84.93	95.04	90.30	75.88
GPT-5.5	76.16	84.43	85.51	0.21	77.13	85.23	15.04	99.36	96.55	81.85	63.28	79.45	92.91	87.31	73.17
Gemini-3.5-flash	78.15	70.97	91.30	0.00	80.89	91.95	0.75	99.47	95.86	79.54	75.78	87.67	93.62	91.79	74.12
Trained on Qwen3-VL-2B															
SFT	70.20	69.34	36.38	7.45	70.51	67.25	37.44	83.04	93.24	82.38	26.72	64.11	30.92	67.46	57.60
DPO	73.51	71.23	46.38	12.92	69.28	71.14	54.14	88.66	94.48	87.13	35.16	71.92	44.68	77.61	64.94
SimPO	71.52	71.79	45.65	13.95	69.28	67.11	54.14	87.50	95.17	87.13	33.59	73.29	42.55	78.36	64.49
ORPO	72.85	72.33	47.10	13.54	71.67	70.47	54.14	88.35	96.55	88.12	35.94	76.71	46.81	76.12	65.05
GRPO	74.50	72.50	47.28	16.65	71.59	72.65	55.64	91.18	95.17	89.77	46.68	76.54	45.21	77.43	66.63
GDPO	72.19	72.77	52.17	13.33	71.33	72.48	53.38	90.08	95.17	89.11	38.28	73.29	42.55	80.60	65.48
DAPO	68.87	72.70	46.38	16.84	70.65	75.17	54.89	88.52	95.86	89.44	36.72	74.66	43.26	79.10	65.22
GSPO	74.17	72.67	45.65	14.78	71.33	72.48	52.63	90.50	95.17	89.77	40.62	76.03	42.55	78.36	65.48
KGRPO	72.85	72.61	57.25	14.09	73.72	75.17	53.38	88.80	95.86	90.76	42.19	71.23	42.55	79.85	66.45
GKGRPO	73.51	72.11	53.62	16.49	73.72	75.84	57.14	88.51	95.17	89.11	39.84	79.45	46.10	79.10	67.12
Trained on Qwen3-VL-8B															
SFT	77.09	80.57	64.78	13.88	76.93	79.60	63.46	95.89	95.72	85.08	48.28	78.49	62.41	79.85	71.57
DPO	80.13	80.94	70.29	16.63	81.91	80.54	74.44	97.50	95.86	89.77	57.03	82.88	77.30	88.06	77.56
SimPO	80.79	82.64	71.01	16.22	83.96	81.21	73.68	97.80	95.86	89.77	56.25	82.19	78.01	86.57	77.60
ORPO	82.78	82.33	68.84	18.08	84.30	82.55	75.94	98.05	96.55	89.77	59.38	81.51	78.72	88.06	77.63
GRPO	82.78	82.20	73.62	20.82	83.96	81.61	78.05	98.01	96.28	89.97	61.09	79.45	78.16	85.82	77.99
GDPO	83.44	82.88	71.74	20.27	84.30	84.56	74.44	97.96	96.55	91.09	54.69	83.56	78.01	87.31	77.92
KGRPO	80.79	81.95	71.74	19.31	84.64	82.55	78.95	98.09	96.55	90.10	62.50	80.82	80.85	85.07	78.14
DAPO	82.12	82.33	68.84	19.59	86.01	77.85	74.44	98.19	96.55	91.75	59.38	80.14	80.85	87.31	77.52
GSPO	79.47	82.67	73.91	17.04	86.69	83.22	77.44	97.86	96.55	85.48	60.16	76.03	82.27	85.07	77.42
GKGRPO	82.78	82.30	72.46	17.66	82.59	85.91	78.95	97.61	97.93	92.08	63.28	81.51	80.85	88.81	78.91

Implementation Details We conduct experiments with Qwen3-VL-2B and Qwen3-VL-8B. SFT is implemented with LLaMA-Factory (Zheng et al., 2024) using LoRA (Hu et al., 2022) of rank 32 for three epochs. We select the learning rate from $\{1 \times 10^{-5}, 1 \times 10^{-4}, 5 \times 10^{-4}\}$ and set the maximum sequence length to 32,768. RL training is implemented with verl (Sheng et al., 2024) for three epochs, with a batch size of 16, four rollout responses per prompt, maximum prompt and response lengths of 8,192 tokens, and an actor learning rate of 1×10^{-6}. For GKGRPO, we set the knowledge-reward coefficients to $\beta_1 = 0.2$ and $\beta_2 = 0.05$. All experiments run on one node with 8× NVIDIA A100 GPUs. Appendix B.1 reports the remaining optimization and rollout settings.

5.2 MAIN EXPERIMENTS

Table 2 reports results on VIVIANA. GKGRPO achieves the highest Overall score at both model scales: 67.12 for Qwen3-VL-2B and 78.91 for Qwen3-VL-8B. These scores improve over SFT by 9.52 and 7.34 points, respectively, and exceed the Overall scores of the evaluated preference-optimization and RL baselines. The 8B GKGRPO model also obtains a higher Overall score than the evaluated closed-source models under the same task protocol.

At the 2B scale, GKGRPO is best or tied-best on T3.1, T3.2, T3.3, and T5.3, which involve aligning visual entities with facts or aggregating graph neighborhoods. At the 8B scale, it leads on T3.2, T3.3, T4.2, T5.1, T5.2, and T6.2. The task-level pattern is consistent with gains in both entity–fact grounding and topology-aware reasoning, although GKGRPO does not lead on every task.

The 8B model shows leading results on a broader set of subgraph-level tasks than the 2B model. This observation suggests that greater model capacity may help exploit structured supervision, but the present comparison does not isolate model scale as a causal factor. T2.2 remains difficult for every evaluated model, including the closed-source systems, identifying open-ended relation extraction as an unresolved challenge.

5.3 FURTHER EXPLORATION ON VIVIANA

Loc In this section, we conduct more experiments on the VIVIANA tasks to further explore their own properties. The experiments consists of two parts: data scalability and cross-task generalization.

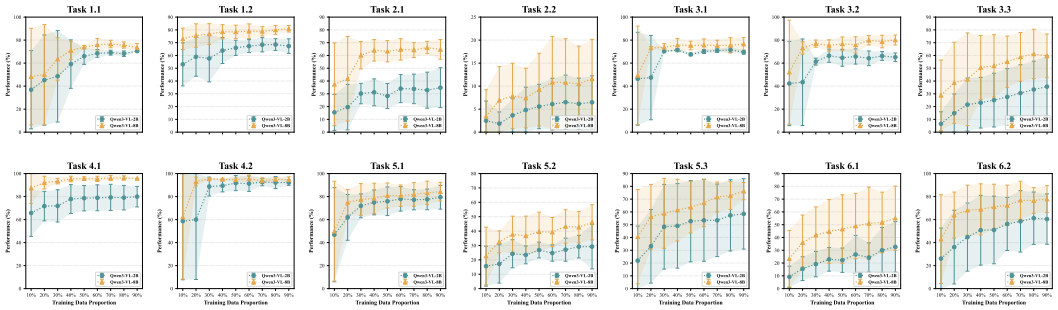


Figure 2: Per-task data scaling on VIVIANA for the Qwen3-VL-2B and 8B models. Scores are reported as the proportion of task data increases.

Data Scalability. Figure 2 shows favorable scaling behavior for models trained with GKGRPO: performance generally improves as more task data are used, and the 8B model consistently outperforms its 2B counterpart. T3.1, T3.2, T4.1, and T4.2 saturate relatively early, whereas more challenging tasks, including T2.2, T3.3, T5.2, T6.1, and T6.2, continue to benefit from additional data. The reduced variance at higher data proportions further demonstrates the stability of GKGRPO and the scalability of VIVIANA.

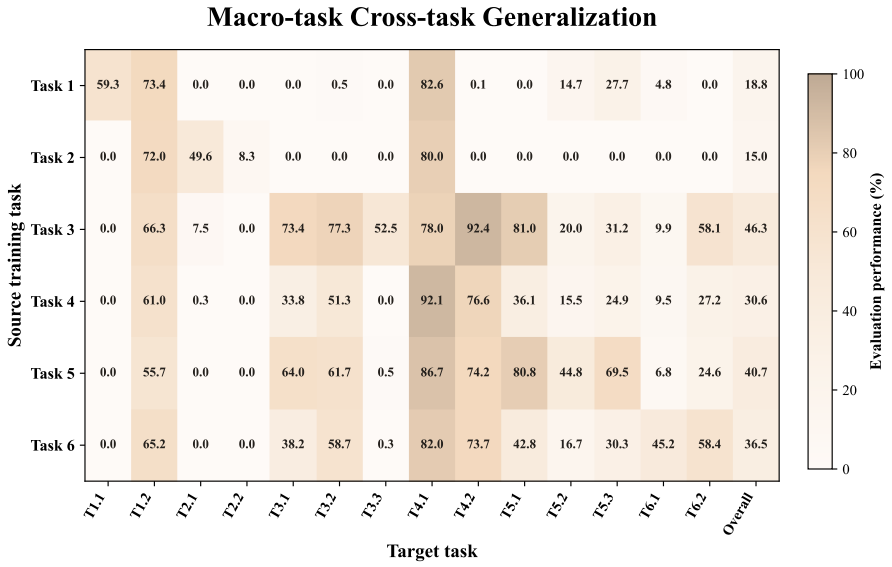


Figure 3: Cross-task generalization on VIVIANA. Each row denotes the training category and each column the evaluation category; scores are reported on the 8B model.

Cross-task Generalization. Figure 3 shows strongly asymmetric transfer across task categories. Training on Task Category 3 gives the highest Overall score (46.3). In addition to retaining strong performance on its own entity–fact tasks (73.4, 77.3, and 52.5 on T3.1–T3.3), it transfers well to path-based relation prediction (92.4 on T4.2), subgraph node membership (81.0 on T5.1), and shared-entity identification (58.1 on T6.2). This breadth suggests that aligning entity images with visualized triples provides an intermediate form of supervision that is useful for both lower-level grounding and subsequent graph composition. Category 5 provides the second-highest Overall score (40.7). Besides strong within-category results, it transfers to entity–fact grounding (64.0 and 61.7 on T3.1 and T3.2) and multiple-fact reasoning (86.7 and 74.2 on T4.1 and T4.2). Category 6 has a lower but still broad Overall score of 36.5, with comparatively strong transfer to T3.2 (58.7), both T4 tasks (82.0 and 73.7), and its own shared-entity task (58.4). Category 4 is more specialized: it

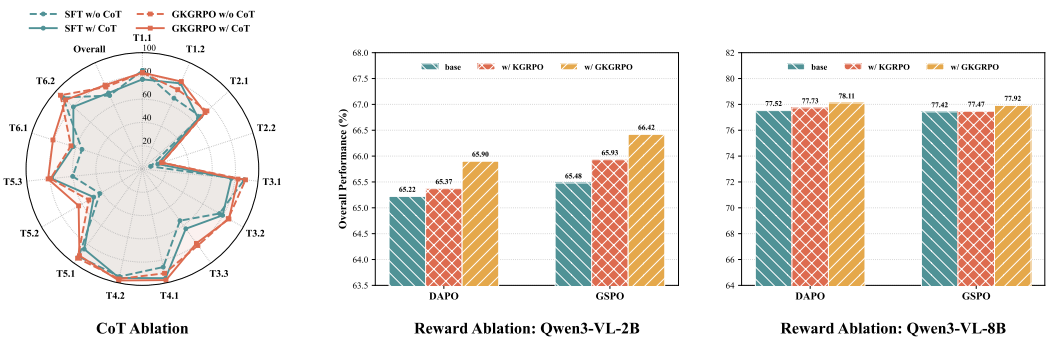


Figure 4: Ablation study on the CoT and reward design.

performs strongly on its two source-aligned tasks (92.1 and 76.6) but reaches an Overall score of 30.6 because transfer to several entity-identification and overlap tasks remains limited.

In contrast, training on Categories 1 and 2 yields Overall scores of only 18.8 and 15.0. Both retain high scores on entity description and subgraph reconstruction, but most exact-match relational tasks remain near zero. The transfer is therefore directional: supervision that explicitly connects entities, facts, or subgraphs transfers to several simpler grounding and description tasks, whereas entity-only or grouping supervision does not provide the relational operations needed by the more compositional tasks. T3.3 and T6.1 are particularly resistant to transfer from unrelated categories, indicating that exact cross-image entity indexing and common-triple extraction require more task-specific supervision. Taken together, these results support joint training across heterogeneous categories while showing that the benefit arises from complementary, rather than interchangeable, task supervision.

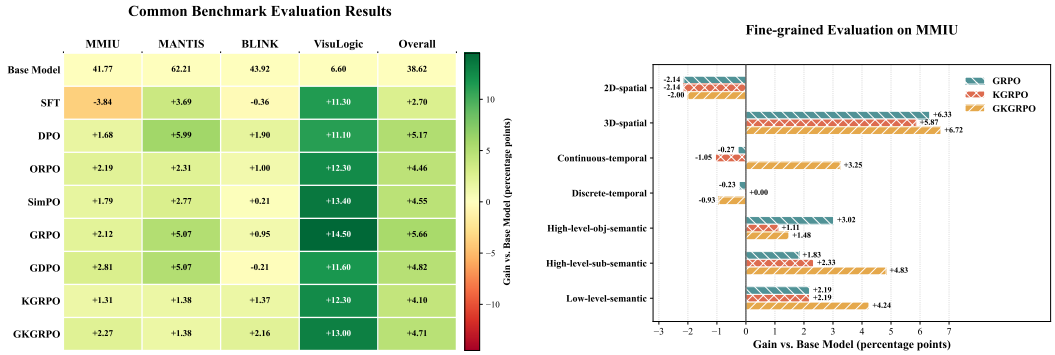
5.4 Further Exploration on GKGRPO

In this section, we present more results on the design of GKGRPO, especially on the synthesised CoT and the global and the knowledge-informed rewards. As shown in Figure 4, the left panel shows that CoT supervision consistently improves both SFT and GKGRPO across most tasks, with particularly clear gains on the more compositional T3–T6 tasks, validating its role in decomposing multimodal graph reasoning into explicit intermediate steps. The middle and right panels further demonstrate that the proposed reward design generalizes beyond GRPO. On Qwen3-VL-2B, incorporating GKGRPO improves DAPO from 65.22 to 65.90 and GSPO from 65.48 to 66.42; on Qwen3-VL-8B, it improves DAPO from 77.52 to 78.11 and GSPO from 77.42 to 77.92. In all four settings, GKGRPO also outperforms the variant with KGRPO alone, showing that the full knowledge-informed reward provides consistent benefits across different RL optimization algorithms and model scales.

5.5 General Benchmark Evaluation

Figure 5 evaluates whether the knowledge-structured training of GKGRPO transfers beyond Viviana to established multi-image MLLM and abstract reasoning benchmarks. The results show consistent improvements over the base model across all evaluated benchmarks: GKGRPO improves MMIU (Meng et al., 2025), MANTIS (Jiang et al., 2024), BLINK (Fu et al., 2024b), and VisuLogic (Xu et al., 2025) by 2.27, 1.38, 2.16, and 13.00 points, respectively, yielding a 4.71-point average gain. The substantial improvement on VisuLogic is particularly notable, indicating that the benefits of GKGRPO are not limited to the visual knowledge-graph format used during training, but also transfer to abstract visual reasoning scenarios that require identifying latent relational rules. Meanwhile, the positive gains on MMIU, MANTIS, and BLINK demonstrate that the proposed training improves general multi-image understanding, including the integration of information distributed across multiple images and the resolution of inter-image relations. Compared with KGRPO, the full GKGRPO further improves the overall gain from 4.10 to 4.71 points, suggesting that its complete reward formulation provides additional transferable supervision.

The fine-grained MMIU results in Figure 5b further connect these gains to the capabilities targeted by Viviana. The entity-centric tasks (T1 and T3) require models to consolidate visual evidence



(a) Common benchmark evaluation.

(b) Detailed evaluation on MMIU subtasks.

Figure 5: Evaluation results on the common benchmark and MMIU subtasks.

and ground entities in factual triples, while the subgraph-oriented tasks (T4–T6) require composing these grounded entities through relational paths and graph structures. This progression from local grounding to relational composition naturally transfers to MMIU's semantic categories: GKGRPO improves low-level semantic and high-level sub-semantic reasoning by 4.24 and 4.83 points, substantially exceeding GRPO and KGRPO on the latter. It also yields large gains on 3D-spatial (+6.72) and continuous-temporal (+3.25) reasoning, where solving a query similarly requires integrating relational evidence across multiple views or observations rather than recognizing a single image in isolation. In contrast, the slight drops on 2D-spatial and discrete-temporal tasks suggest that exact metric localization and strict event-order judgments are less directly captured by the current entity–relation–subgraph supervision. The transfer pattern is consistent with the design of VIVIANA: it most strongly improves multi-image grounding and compositional relational abstraction, while identifying precise geometry and discrete temporal ordering as complementary capabilities.

6 Conclusion

In this work, we investigate abstractive visual reasoning in multi-image settings, where models must jointly interpret heterogeneous sequences of concrete entity images, visualized facts, and structured mind maps. To support this study, we introduce VIVIANA, a benchmark comprising 22K instances across 14 tasks that span visual grounding, cross-image relational inference, and subgraph-level reasoning. We further propose GKGRPO, a gated knowledge-informed GRPO framework that combines outcome supervision with local and global knowledge rewards while down-weighting process rewards for trajectories that produce incorrect answers. Experiments on two model scales demonstrate consistent improvements over supervised fine-tuning, preference-optimization, and reinforcement-learning baselines. The resulting models also transfer favorably to established multi-image understanding and abstract visual reasoning benchmarks, suggesting that learning to connect concrete visual evidence with structured abstractions benefits broader multimodal reasoning. We hope that VIVIANA and GKGRPO will facilitate further research on MLLMs that can reason reliably across diverse, interleaved visual representations.

AI Use Statement

We used generative AI tools to polish the language of author-written manuscript drafts and to assist in synthesizing the experimental dataset, including the local image descriptions and reasoning annotations described in the data-construction pipeline. The authors reviewed the AI-assisted writing and verified the synthesized data against the corresponding task inputs and graph-derived answers. We did not use generative AI tools to write experimental code or to generate experimental results; all reported results were obtained by running the implemented methods and evaluation procedures. The authors take full responsibility for the final manuscript, dataset, analyses, and reported findings.

ETHICS STATEMENT

This work uses publicly available datasets and open-source models for dataset construction and model training; closed-source models are included only as evaluation baselines. We followed the applicable terms of use for these resources and did not collect data through interventions involving human participants. During dataset construction, we combined automated filtering with human review to remove violent, sexually explicit, discriminatory, and otherwise inappropriate content. Although these procedures reduce potential risks, automatically generated annotations and source data may still contain residual errors or biases. We therefore encourage users of VIVIANA to apply appropriate safeguards and to use the dataset only for lawful and responsible research. To the best of our knowledge, the experiments and reporting in this work comply with established academic research ethics.

REPRODUCIBILITY STATEMENT

We provide the information needed to reproduce the principal results of this work. The main paper describes the VIVIANA construction pipeline, task composition procedure, CoT synthesis process, and GKGRPO objective. The appendix provides per-task dataset statistics and definitions, the complete task instruction templates, the LLM-as-a-judge evaluation prompt and decoding settings, task-specific scoring rules, and detailed SFT and RL hyperparameters. Together, these materials specify the data construction, training, and evaluation procedures used in our experiments.

REFERENCES

Houda Alberts, Teresa Huang, Yash Deshpande, Yibo Liu, Kyunghyun Cho, Clara Vania, and Iacer Calixto. Visualsem: a high-quality knowledge graph for vision and language. *CoRR*, abs/2008.09150, 2020.

Kurt D. Bollacker, Colin Evans, Praveen K. Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *SIGMOD Conference*, pp. 1247–1250. ACM, 2008.

John Ellson, Emden R. Gansner, Eleftherios Koutsofios, Stephen C. North, and Gordon Woodhull. Graphviz and dynagraph - static and dynamic graph drawing tools. In *Graph Drawing Software*, pp. 127–148. Springer, 2004.

Chaoyou Fu, Yifan Zhang, Shukang Yin, Bo Li, Xinyu Fang, Sirui Zhao, Haodong Duan, Xing Sun, Ziwei Liu, Liang Wang, Caifeng Shan, and Ran He. Mme-survey: A comprehensive survey on evaluation of multimodal llms. *CoRR*, abs/2411.15296, 2024a.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. BLINK: multimodal large language models can see but not perceive. In *ECCV (23)*, volume 15081 of *Lecture Notes in Computer Science*, pp. 148–166. Springer, 2024b.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, Hao Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, Tao Wang,

Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nat.*, 645(8081):633–638, 2025.

Jiwoo Hong, Noah Lee, and James Thorne. ORPO: monolithic preference optimization without reference model. In *EMNLP*, pp. 11170–11189. Association for Computational Linguistics, 2024.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *ICLR*. OpenReview.net, 2022.

Dongfu Jiang, Xuan He, Huaye Zeng, Cong Wei, Max Ku, Qian Liu, and Wenhu Chen. Mantis: Interleaved multi-image instruction tuning. *Trans. Mach. Learn. Res.*, 2024, 2024.

Shih-Yang Liu, Xin Dong, Ximing Lu, Shizhe Diao, Peter Belcak, Mingjie Liu, Min-Hung Chen, Hongxu Yin, Yu-Chiang Frank Wang, Kwang-Ting Cheng, Yejin Choi, Jan Kautz, and Pavlo Molchanov. GDPO: group reward-decoupled normalization policy optimization for multi-reward RL optimization. *CoRR*, abs/2601.05242, 2026.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq R. Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *ACL (Findings)*, Findings of ACL, pp. 2263–2279. Association for Computational Linguistics, 2022.

Fanqing Meng, Jin Wang, Chuanhao Li, Quanfeng Lu, Hao Tian, Tianshuo Yang, Jiaqi Liao, Xizhou Zhu, Jifeng Dai, Yu Qiao, Ping Luo, Kaipeng Zhang, and Wenqi Shao. MMIU: multimodal multi-image understanding for evaluating large vision-language models. In *ICLR*. OpenReview.net, 2025.

Yu Meng, Mengzhou Xia, and Danqi Chen. Simpo: Simple preference optimization with a reference-free reward. In *NeurIPS*, 2024.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*, 2023.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*, 2024.

Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs. In *ACL (1)*, pp. 6418–6428. Association for Computational Linguistics, 2019.

Fei Wang, Xingyu Fu, James Y. Huang, Zekun Li, Qin Liu, Xiaogeng Liu, Mingyu Derek Ma, Nan Xu, Wenxuan Zhou, Kai Zhang, Tianyi Lorena Yan, Wenjie Jacky Mo, Hsiang-Hui Liu, Pan Lu, Chunyuan Li, Chaowei Xiao, Kai-Wei Chang, Dan Roth, Sheng Zhang, Hoifung Poon, and Muhao Chen. Muirbench: A comprehensive benchmark for robust multi-image understanding. In *ICLR*. OpenReview.net, 2025.

Derong Xu, Tong Xu, Shiwei Wu, Jingbo Zhou, and Enhong Chen. Relation-enhanced Negative Sampling for Multimodal Knowledge Graph Completion. In *ACM Multimedia*, pp. 3857–3866. ACM, 2022.

Weiye Xu, Jiahao Wang, Weiyun Wang, Zhe Chen, Wengang Zhou, Aijun Yang, Lewei Lu, Houqiang Li, Xiaohua Wang, Xizhou Zhu, Wenhai Wang, Jifeng Dai, and Jinguo Zhu. Visulogic: A benchmark for evaluating visual reasoning in multi-modal large language models. *CoRR*, abs/2504.15279, 2025.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Weinan Dai, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. DAPO: an open-source LLM reinforcement learning system at scale. *CoRR*, abs/2503.14476, 2025.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Yu Qiao, Peng Gao, and Hongsheng Li. MATHVERSE: does your multi-modal LLM truly see the diagrams in visual math problems? In *ECCV (8)*, Lecture Notes in Computer Science, pp. 169–186. Springer, 2024.

Yichi Zhang, Zhuo Chen, Lingbing Guo, Yajing Xu, Min Zhang, Wen Zhang, and Huajun Chen. Abstractive visual understanding of multi-modal structured knowledge: A new perspective for mllm evaluation, 2025. URL https://arxiv.org/abs/2506.01293.

Yichi Zhang, Zhuo Chen, Lingbing Guo, Wen Zhang, and Huajun Chen. Structured and abstractive reasoning on multi-modal relational knowledge images. In *ACL (Findings)*, pp. 15518–15540. Association for Computational Linguistics, 2026a.

Yichi Zhang, Jingdian Lu, Zhuo Chen, Lingbing Guo, Jun Xu, Wen Zhang, and Huajun Chen. Symbolic and abstractive reasoning with complex visual queries. *CoRR*, abs/2606.09195, 2026b.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, Jingren Zhou, and Junyang Lin. Group sequence policy optimization. *CoRR*, abs/2507.18071, 2025.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand, 2024. Association for Computational Linguistics. URL http://arxiv.org/abs/2403.13372.

A Appendix

A.1 Detailed Information of Viviana

A.1.1 Task Definitions

We formulate 14 tasks under 6 broad categories according to the semantic granularity expressed by the visual inputs. At the atomic level, an image may represent a concrete entity, a factual relation, or an entity-centered mind map, corresponding respectively to a node, a directed edge, and a subgraph in an MMKG. By composing these atomic visual inputs into heterogeneous image sequences, we obtain tasks that require models to ground entities, recover relations, and reason over graph structures. The six categories and their associated tasks are as follows:

Multiple Images of a Single Entity. This category evaluates whether a model can consolidate multiple visual observations of the same entity.

- **Task 1.1: Entity–Image Consistency.** Given an entity name and a sequence of images, the model determines whether every image depicts the specified entity, producing a binary *Yes*/*No* answer.
- **Task 1.2: Entity Description.** Given several images of an entity, the model generates a coherent description that summarizes its visual attributes and salient semantic characteristics.

Table 3: Per-task statistics for VIVIANA. "Images" counts image occurrences across instances.

Task	Task Type	Samples	Images	Train	Valid	Test
1.1	Entity-Image Consistency	1,415	6,977	1,123	141	151
1.2	Entity Description	1,654	6,714	1,333	162	159
2.1	Multi-Entity Image Grouping	1,350	9,713	1,086	126	138
2.2	Relation Extraction	1,021	5,931	824	100	97
3.1	Triple Membership Verification	3,049	6,098	2,438	318	293
3.2	Masked Triple Completion	1,330	7,331	1,060	121	149
3.3	Triple Entity Grounding	1,385	9,855	1,106	146	133
4.1	Subgraph Reconstruction	1,453	6,403	1,151	161	141
4.2	Path-based Relation Prediction	1,577	4,637	1,274	158	145
5.1	Subgraph Node Membership	2,948	5,896	2,361	284	303
5.2	Shortest-Path Reasoning	1,329	3,987	1,059	142	128
5.3	Common-Neighbor Counting	1,410	4,230	1,126	138	146
6.1	Common-Triple Extraction	1,448	2,896	1,166	141	141
6.2	Shared-Entity Identification	1,199	9,120	947	118	134
	Total	**22,568**	**89,788**	**18,054**	**2,256**	**2,258**

Images of Multiple Entities. This category requires models to distinguish several entities and infer their relationships from mixed visual evidence.

- **Task 2.1: Multi-Entity Image Grouping.** Given a shuffled collection of images and a set of entity names, the model groups the images according to their corresponding entities.
- **Task 2.2: Relation Extraction.** Given images of multiple named entities, the model identifies all supported relations among them and outputs the resulting knowledge-graph triples.

Entity + Fact. This category combines entity photographs with visualized knowledge-graph triples, testing cross-modal entity grounding and factual reasoning.

- **Task 3.1: Triple Membership Verification.** Given an entity image and a visualized triple, the model determines whether the depicted entity is the head or tail entity in the triple.
- **Task 3.2: Masked Triple Completion.** Given an incomplete triple with a masked head or tail and several candidate entity images, the model selects the entity that correctly completes the triple.
- **Task 3.3: Triple Entity Grounding.** Given one or more triple visualizations and candidate entity images, the model identifies the image indices corresponding to the head and tail of each triple.

Multiple Facts. This category presents several factual triple visualizations and requires structural organization or relational inference across them.

- **Task 4.1: Subgraph Reconstruction.** Given shuffled triple visualizations, the model reorganizes them into a coherent subgraph and describes its structure.
- **Task 4.2: Path-based Relation Prediction.** Given a multi-hop path of visualized triples and a queried entity pair, the model infers the relation between the two queried entities.

Entity + Subgraph. This category pairs one or two entity images with a visualized subgraph, requiring node-level and topology-aware reasoning.

- **Task 5.1: Subgraph Node Membership.** Given an entity image and a subgraph visualization, the model determines whether the entity appears as a node in the subgraph.
- **Task 5.2: Shortest-Path Reasoning.** Given two entity images and a subgraph containing both entities, the model calculates the number of edges in their shortest connecting path.
- **Task 5.3: Common-Neighbor Counting.** Given two entity images and a subgraph, the model counts the entities that are directly connected to both queried entities.

Subgraph + Subgraph. This category compares two or more subgraph visualizations and tests set-level reasoning over their nodes and edges.

- **Task 6.1: Common-Triple Extraction.** Given two subgraphs, the model identifies and outputs every triple shared by both subgraphs, or outputs `None` when no common triple exists.
- **Task 6.2: Shared-Entity Identification.** Given multiple subgraphs and candidate entity images, the model selects the candidate entity that appears in every subgraph.

A.2 Task Prompt Templates

The templates of the 14 different tasks are shown below:

Task instruction templates used in VIVIANA

Task 1.1: Entity–Image Consistency.

`<image> ...`
Given a sequence of images, your task is to determine whether all the images belong to an entity called `[ENTITY]`.
Output:
- "Yes" if all images belong to the same entity.
- "No" if any image does not belong to the same entity as the others.

Task 1.2: Entity Description.

`<image> ...`
Given a sequence of images, your task is to describe the entity called `[ENTITY]`.
Please carefully examine the images and provide a description of the entity based on what you observe.

Task 2.1: Multi-Entity Image Grouping.

`<image> ...`
Given a sequence of images, these images belong to `[NUM_ENTITIES]` different entities: `[ENTITY_LIST]`.
Your task is to group the images by entity, and identify which images belong to which entity.

Task 2.2: Relation Extraction.

`<image> ...`
Given the images of `[NUM_ENTITIES]` entities:
Image 1: `[ENTITY_1]`
...
Please identify all the relations between these entities based on their images.
For each related entity pair, output the head entity, tail entity, and their relationship.

Task 3.1: Triple Membership Verification.

`<image><image>`
You are given two images:
Image 1: A photo of an unknown entity.
Image 2: A knowledge graph showing a relationship triple between two entities.
Based on the visual appearance of the entity in Image 1, does this entity appear in the knowledge graph shown in Image 2?
Please answer with "Yes" or "No".

Task 3.2: Masked Triple Completion.

`<image> ...`
You are given several images:
The first is a knowledge triple visualization showing an incomplete triple, where one entity is masked as `[MASK]`.
The others are photos of candidate entities.
The incomplete triple is missing one entity.
Based on the visual appearance of the candidates, which entity should fill in the `[MASK]` position?

A: [CANDIDATE_A]
...
Please answer with a single letter (A, B, C or D).

Task 3.3: Triple Entity Grounding.

<image> ...
You are given several images:
The first image is a knowledge graph visualization showing a triple.
The others are photos of candidate entities.
For each triple visualization, identify which entity photos correspond to its head and tail entities.
Please answer with the image indices (start from 0) for each triple's head and tail, in the following format (one line, no spaces):
[[head_index,tail_index],...]

Task 4.1: Subgraph Reconstruction.

<image> ...
You are given several images:
The images are visualized knowledge triples, each showing a relation triple between two entities.
There are [NUM_TRIPLES] triples in total, but they are given in a shuffled order.
Please reorganize these triples to form a coherent subgraph and describe it.
Please describe the subgraph structure.

Task 4.2: Path-based Relation Prediction.

<image> ...
You are given several images, each visualizing a knowledge triple that forms a reasoning path between entities.
There are [NUM_TRIPLES] triples in total, forming a path:
[ENTITY_A] -[RELATION_1]-> ... -[RELATION_N]-> [ENTITY_B]
Based on the reasoning path shown above, what is the most likely relation between "[ENTITY_A]" and "[ENTITY_B]"?
A. [RELATION_A]
...
Please answer with a single letter (e.g. A, B, C, ...).

Task 5.1: Subgraph Node Membership.

<image><image>
You are given two images:
The first image is a photo of an unknown entity.
The second image is a visualized knowledge subgraph showing relations between entities.
Does the entity shown in the first image appear as a node in the subgraph shown in the second image?
Please answer with "Yes" or "No".

Task 5.2: Shortest-Path Reasoning.

<image><image><image>
You are given three images:
The first image shows entity A.
The second image shows entity B.
The third image is a visualized knowledge subgraph showing relations between entities, which includes both entity A and entity B.
Given the subgraph, what is the length of the shortest path (number of edges) between entity A and entity B in the subgraph?

Task 5.3: Common-Neighbor Counting.

<image><image><image>
You are given three images:
The first image shows entity A.
The second image shows entity B.

The third image is a visualized knowledge subgraph showing relations between entities, which includes both entity A and entity B.

How many common neighbor entities do entity A and entity B have in the subgraph? Note: A common neighbor is an entity that is directly connected to both entity A and entity B in the subgraph.

Task 6.1: Common-Triple Extraction.

`<image><image>`

You are given two images:
The first image is a visualized knowledge subgraph showing relations between entities.
The second image is another visualized knowledge subgraph showing relations between entities.

Identify all the triples (head, relation, tail) that appear in both subgraphs. List each triple in the format: "head - relation - tail".

Please list all overlapping triples, one per line. If there are no overlapping triples, output "None".

Task 6.2: Shared-Entity Identification.

`<image> ...`

You are given a series of knowledge subgraphs and a set of candidate entities.

The first `[NUM_SUBGRAPHS]` images are visualized knowledge subgraphs showing relations between entities.
The following `[NUM_CANDIDATES]` images are candidate entities, labeled with letters A, B, C, ...

Which candidate entity appears in ALL of the subgraphs? Note: An entity appears in a subgraph if it is either the head or tail of any triple in that subgraph.

Please answer with the letter of the correct candidate entity.

A.3 LLM-AS-A-JUDGE EVALUATION PROMPT

We use the following prompt to evaluate the open-ended outputs of Tasks 1.2 and 4.1.

Evaluation prompt for Tasks 1.2 and 4.1

You are a strict text similarity evaluation expert.

Your task is to evaluate the semantic similarity between the Prediction and the Label.

Scoring Rules

- Return ONLY a single integer between 0 and 100, nothing else
- 0 means completely dissimilar, 100 means perfectly identical
- Evaluate based on the following dimensions:
  1. Semantic Consistency: whether the core meaning is the same
  2. Key Information Coverage: whether all important information is included
  3. Factual Accuracy: whether there are any errors or contradictions
  4. Completeness: whether any critical content is missing

Input

Label:
{}
Prediction:
{}

Output Requirement

Output a single integer only, range 0-100, with no explanation, punctuation, or any other characters.

We run Qwen3.6 with a temperature of 0 and a maximum generation length of 8,192 tokens. Under this deterministic decoding setting, the model returns a single integer between 0 and 100 as the evaluation score. For the small number of outputs that do not follow the required format, we rerun the evaluation with the same configuration.

A.4 Accuracy Reward

Below, task identifiers follow the renumbered task taxonomy used in this work. Each prediction must use the format `<think>...</think><answer>...</answer>`; the task-specific content described below is placed in the `<answer>` field. Below, task identifiers follow the renumbered task taxonomy used in this work.

- **Task 1.1: Entity–Image Consistency.** The answer must be `Yes` or `No`. Full accuracy reward is given only for an exact match with the reference label.
- **Task 1.2: Entity Description.** The answer is a free-form textual description of the entity inferred from the image sequence. No answer-accuracy reward is assigned for this open-ended generation task; it is rewarded only through the CoT component.
- **Task 2.1: Multi-Entity Image Grouping.** The answer must be a dictionary mapping each entity name to its image indices, e.g., {`'entity A': [1, 3], 'entity B': [2]`}. Partial credit is the proportion of reference entity keys whose complete index list is exactly recovered.
- **Task 2.2: Relation Extraction.** The answer must be a list of relation dictionaries, each containing `head`, `tail`, and `relation` fields. Partial credit is the proportion of reference triples contained in the predicted triple set.
- **Task 3.1: Triple Membership Verification.** The answer must be `Yes` or `No`, indicating whether the pictured entity occurs as the head or tail of the displayed triple. It is scored by exact match.
- **Task 3.2: Masked Triple Completion.** The answer must be the letter of the selected candidate, such as `A`, `B`, `C`, or `D`. It is scored by exact match.
- **Task 3.3: Triple Entity Grounding.** The answer must be a list of ordered index pairs in the form `[[head_index, tail_index], ...]`. Partial credit is the proportion of correctly recovered head–tail index pairs.
- **Task 4.1: Subgraph Reconstruction.** The answer is a free-form description of the reconstructed subgraph and its structure. No answer-accuracy reward is assigned; this task is rewarded only through the CoT component.
- **Task 4.2: Path-based Relation Prediction.** The answer must be the letter of the selected relation candidate. It is scored by exact match.
- **Task 5.1: Subgraph Node Membership.** The answer must be `Yes` or `No`, indicating whether the entity is a node in the subgraph. It is scored by exact match.
- **Task 5.2: Shortest-Path Reasoning.** The answer must be an integer representing the shortest-path length, measured by the number of edges. It is scored by exact match.
- **Task 5.3: Common-Neighbor Counting.** The answer must be an integer giving the number of nodes directly connected to both queried entities. It is scored by exact match.
- **Task 6.1: Common-Triple Extraction.** The answer must list each shared triple on a separate line using the format `head - relation - tail`; when no overlap exists, it must output `None`. Partial credit is the proportion of reference overlapping triples that are correctly output.
- **Task 6.2: Shared-Entity Identification.** The answer must be the letter of the candidate entity that appears in every subgraph. It is scored by exact match.

B Experiments

B.1 Implementation Details

For SFT, we adopt LoRA for parameter-efficient adaptation, with rank 32 and all model modules as LoRA targets. For each model size, we tune the learning rates in 1×10^{-4}, 5×10^{-4}, and 1×10^{-5}. Training is conducted for three epochs using the cosine learning-rate scheduler with a warm-up ratio of 0.1. We use a per-device batch size of 1 and gradient accumulation of 8, resulting in an effective

batch size of 8 per device. The maximum sequence length is set to $32{,}768$ and all experiments use bfloat16 precision.

We further optimize the fine-tuned MLLMs (after merging the LoRA) with GKGRPO. Training uses a batch size of 16 and runs for three epochs on a single node with eight GPUs. For each prompt, we sample 4 responses, with the maximum prompt and response lengths both set to $8{,}192$ tokens; overlong prompts are filtered and truncation is treated as an error. The actor is optimized with a learning rate of 1×10^{-6}, a PPO mini-batch size of 16, and a per-GPU PPO micro-batch size of 4. During rollout, vLLM is used with tensor parallelism of 2, a per-GPU log-probability micro-batch size of 4, and GPU memory utilization capped at 0.75. The hyper-parameter reward weight β_1, β_2 are set to 0.2, 0.05 respectively. For KGRPO (Zhang et al., 2026a), we tune the hyper-parameter α from 0.7 to 0.9