From Scratch to Precise Illustration: Automatic Illustration Generation with Iterative Multi-Agent Refinement

Anonymous ACL submission

Abstract

Automated generation of illustrations for educational exams can significantly reduce the labor-intensive workload of human exam authors. While recent LLM research has automated textual question generation, visual components remain underexplored despite their prevalence in real-world exams. In this work, we extend tool-augmented LLMs to illustration generation via specialized Tool Sandbox, and introduce **Actre**, an iterative multi-agent framework that improves tool utilization through feedback-driven refinement. Furthermore, we propose **IllustrationBench**, a graded benchmark of 320 real college entrance exam questions across four types and three difficulty levels. Experiments show Actre significantly outperforms one-shot baselines, particularly on high-difficulty tasks, demonstrating the effectiveness of iterative tool-augmented agents for educational content generation.

1 Introduction

Visual aids, such as geometric diagrams and function curves, are indispensable in educational exams (Strohmaier et al., 2020). Historically, the creation of these illustrations has relied on manual drafting by human experts, which has become a significant bottleneck; therefore, replacing it with AI is essential for increasing educational productivity.

However, automatically generating constraint-compliant images remains challenging (Bosheah and Bilicki, 2025). The difficulty lies in reconciling varied linguistic descriptions with rigid mathematical constraints. On one hand, generic text-to-image models (Rombach et al., 2022; Li et al., 2019; Ramesh et al., 2021) lack the "spatial consciousness" needed to respect precise coordinates or topological relationships. On the other hand, rule-based systems (Udristoiu et al., 2010), while mathematically sound, are often too rigid to decode the nuanced and diverse ways in which humans describe problems.

Recent advances in Agent-LLMs (Parisi et al., 2022; Chen et al., 2024) offer a promising direction (Qu et al., 2025). However, existing LLM-based educational applications (Motwani et al., 2024) have predominantly focused on the generation of purely textual questions(Wang et al., 2022; Nguyen et al., 2022). This creates a significant "missing link" in the automated pipeline: a text-only question is incomplete without its corresponding visual context.

To bridge this gap, we propose a framework that equips Agent-LLMs with a tailored *Tool Sandbox* and evaluate it on our proposed IllustrationBench, a new graded benchmark built from 320 real college entrance exam questions across four types and three difficulty levels, each paired with official reference images. We introduce an enhanced agent architecture that improves illustration quality through iterative error correction guided by multimodal feedback(Goswami et al., 2025). Experiments show that our approach significantly outperforms one-shot baselines, especially on high-difficulty geometry and function tasks, demonstrating that illustration generation is inherently iterative. Our work makes three contributions:

- We pioneer the application of Agent-LLMs to propositional illustration in education.
- We introduce Actre—an agent framework that improves tool-augmented generation via collaborative drafting and revision.
- We then propose IllustrationBench—a benchmark designed to systematically evaluate the precision and logical alignment of existing illustration generation systems.

2 Related Works

2.1 Tool-Augmented Language Agents

Recent work shows that equipping LLMs with external tools effectively tackles complex tasks, from code generation (Schick et al., 2023) to mathemati-

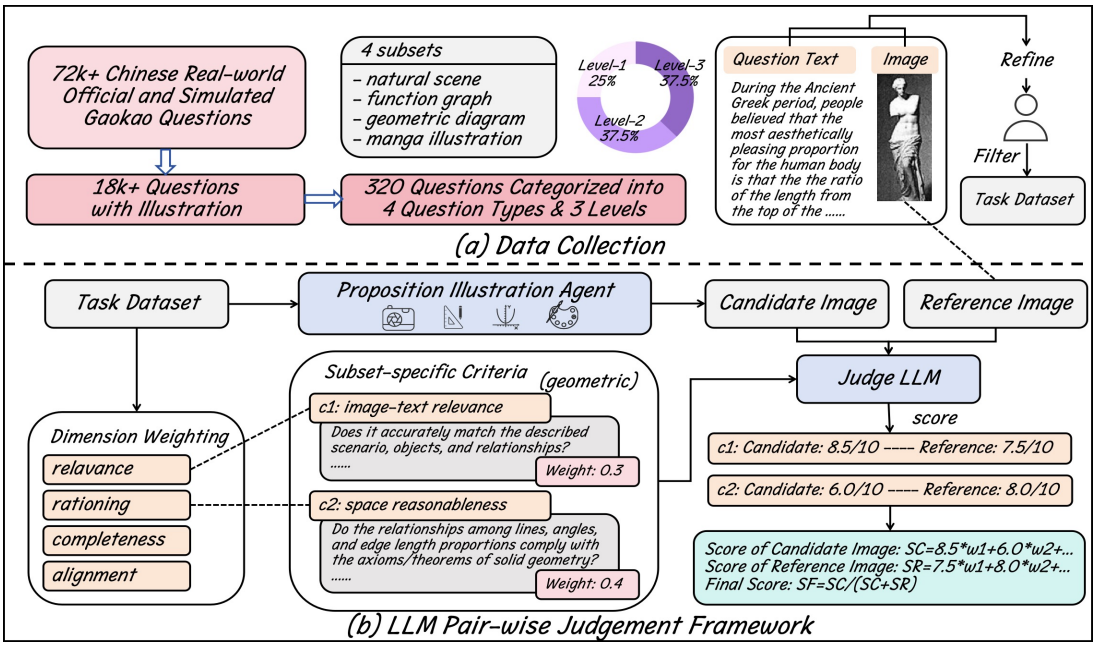


Figure 1: Overview of IllustrationBench: (a) Data curation from Gaokao exams into 4 types × 3 levels; (b) LLM judge evaluates candidate vs. reference images using weighted, subset-specific criteria.

cal reasoning (Qin et al., 2023). Frameworks like ReAct (Yao et al., 2022) and Toolformer (Schick et al., 2023) integrate reasoning with tool use, while newer approaches focus on iterative refinement (Zhang et al., 2024) or domain-specific tool integration (Wu et al., 2023). However, most of them are limited to textual outputs, and tasks requiring visual outputs that are precisely aligned with the input text constraints receive little attention.

2.2 AI for Educational Content Generation

Automated educational content generation has seen progress in problem synthesis (Polozov et al., 2015), diagram parsing (Gao et al., 2023), and multimodal QA (Pramanick et al., 2024). In particular, some methods explore question generation with image augmented (Patil and Patwardhan, 2020). However, even in these cases, images are often treated as supplementary or retrieved from databases, rather than being synthesized under explicit constraints.

3 Methodology

We present IllustrationBench, a benchmark for evaluating LLMs' ability to generate accurate, semantically aligned illustrations from proposition-based exam questions (Figure 1), and Actre, a two-agent framework that iteratively refines outputs through *Act-then-Revise* cycles: an *Actor* drafts illustrations via tool invocation, while a *Revisor* provides critiques until convergence or a maximum iteration limit (Figure 2).

3.1 IllustrationBench

Data Collection. IllustrationBench is curated from over 72k recent official Gaokao and 2025 mock exam questions (Beijing, Tianjin, Fujian). From 18k+ illustrated items, we retain 320 questions that (i) support tool-augmented agent automation and (ii) effectively probe textual-to-visual alignment, evenly distributed across four illustration-dependent categories:

- **Geometric Diagrams**: Require precise planar or spatial figures which are sensitive to coordinate and topological correctness.
- **Function Graphs**: Plot mathematical functions with key features essential for solving.
- **Natural Scene Illustrations**: Depict real-world contexts (e.g., architecture, landscapes); prioritize contextual fidelity over photorealism.
- **Manga-style Illustrations**: Narrative cartoons conveying abstract or metaphorical ideas.

Each category contains 80 questions, stratified by difficulty (L1: 20, L2: 30, L3: 30) based on *graphical reasoning burden*; more detailed criteria are in Appendix B.

Evaluation Framework. Given the open-ended nature of illustration generation, we adopt a *pairwise comparative judgment* protocol (Jeong et al.,

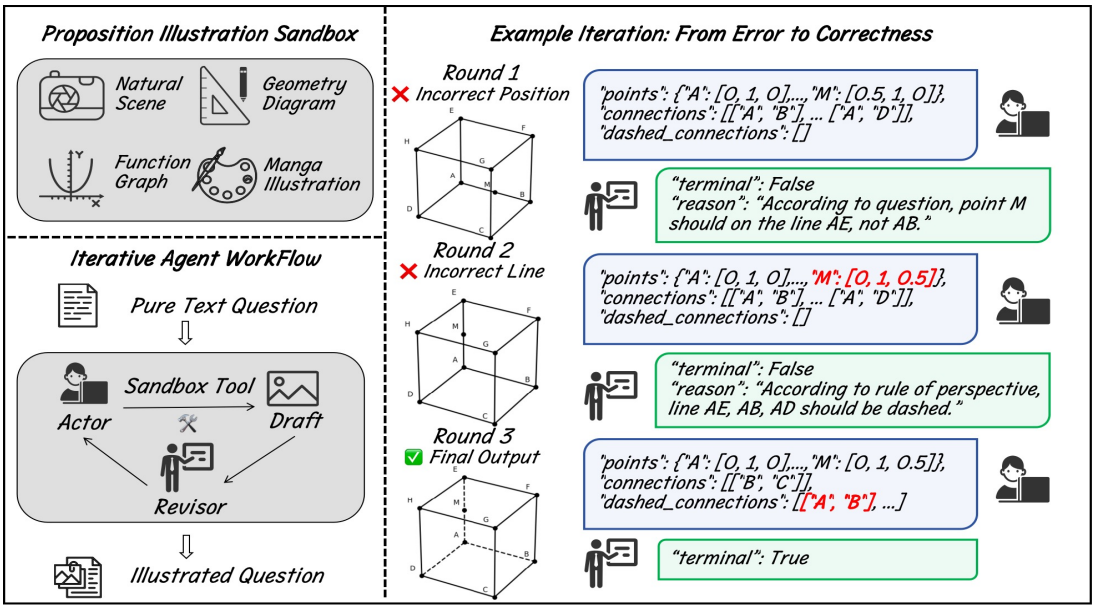


Figure 2: The Actre framework iteratively refines illustrations via Actor-Revisor collaboration using a domain-specific sandbox. Right: An Example showing how geometric errors are corrected across rounds.

2025) using LLM judges. For each question, candidate image $\mathcal{I}_c$ and reference image $\mathcal{I}_r$ are scored against type-specific criteria $\mathcal{C} = \{c_1, \ldots, c_k\}$ with weights $\{w_i\}$ ($\sum w_i = 1$). Each criterion yields $s_i(\mathcal{I}) \in [0, 10]$, and total score is $S(\mathcal{I}) = \sum w_i \cdot s_i(\mathcal{I})$.

To mitigate positional bias in LLM's judgement, we perform *position-swapped evaluations*: both $(\mathcal{I}_c, \mathcal{I}_r)$ and $(\mathcal{I}_r, \mathcal{I}_c)$ are judged, and scores are averaged. The final normalized score is:

$$\frac{\text{Avg}(S(\mathcal{I}_c))}{\text{Avg}(S(\mathcal{I}_c)) + \text{Avg}(S(\mathcal{I}_r))}.$$

This ensures fair, calibrated comparisons across diverse illustration types. Type-specific weights are detailed in Appendix C.

3.2 Actre: Iterative Refinement for Proposition Illustration Generation

Actre addresses the limitations of one-shot tool invocation by emulating human-like iterative refinement through an *Act-then-Revise* loop. The framework consists of two collaborative agents:

- **Actor Agent** (A): Generates draft illustrations by invoking a tool from the Proposition Illustration Sandbox with inferred parameters $(t, \mathbf{p})$, conditioned on the question Q and revision feedback.
- **Revisor Agent** (R): Evaluates the Actor's output against the question's semantic and graphical requirements, returning structured feedback that includes a boolean `terminal` flag (indicating whether the draft is acceptable) and a natural language `reason` explaining the judgment.

The workflow proceeds iteratively:

1. **Initialization**: Actor receives Q and generates initial draft $D_0 = \text{Tool}(t_0, \mathbf{p}_0)$.
2. **Revision Round** k: Revisor evaluates D_k and returns $(\texttt{terminal}_k, \texttt{reason}_k)$.
3. **Termination Check**: If $\texttt{terminal}_k = \texttt{True}$ or $k = K_{\max}$, stop.
4. **Update**: Actor uses $\texttt{reason}_k$ to refine parameters and generate D_{k+1}.

As shown in Figure 2, even a simple cube may require multiple corrections—e.g., fixing misplaced points or missing hidden edges—to satisfy geometric and perspective constraints, with each round reducing errors that would cause one-shot failure.

4 Experiments

4.1 Baselines

We evaluate a diverse set of models, spanning closed-source commercial systems, open-weighted large language models, and dedicated reasoning models, along with our proposed Actre variants. These baselines are categorized into two groups: (1) *One-shot Generators*, which directly invoke tools without iteration; and (2) *Iterative Refiners*, which incorporate our Actre framework—where the Revisor agent in all iterative variants is implemented via the multimodal version of gpt-4o. Details of tool sandbox are provided in Appendix A.

Models	Natural Scene				Function				Geometry				Manga			
	Lvl-1	Lvl-2	Lvl-3	Avg.	Lvl-1	Lvl-2	Lvl-3	Avg.	Lvl-1	Lvl-2	Lvl-3	Avg.	Lvl-1	Lvl-2	Lvl-3	Avg.
Grok-3-mini	32.3	27.5	23.1	27.1	29.2	22.1	10.1	19.4	30.8	18.3	7.2	17.3	34.5	28.1	23.4	27.9
Grok-3	36.4	31.2	28.5	31.5	33.5	26.8	15.4	24.2	35.1	22.7	10.9	21.4	38.2	31.6	26.3	31.3
Qwen3-8B	31.1	26.8	22.4	26.2	28.1	20.6	10.7	18.8	33.6	20.4	7.1	18.7	36.9	30.2	25.1	30
Qwen3-235B-A22B	39.2	34.6	30.1	34.1	37.9	31.2	21.4	29.2	41.2	28.9	14.3	26.5	43.5	36.8	31.2	36.4
DeepSeek-v3	39.5	35.8	32.4	35.5	36.5	27.6	22.1	27.8	40.7	28.6	15.2	26.6	40.1	33.9	28.7	33.5
DeepSeek-r1	43.8	39.2	36.5	39.3	43.2	34.4	28.1	34.2	44.3	30.8	17.1	29	42.8	35.6	30.1	35.3
gpt-4o	39.9	35.1	31.8	35.1	34.6	26.9	21.9	27	41.8	29.1	15.6	27.2	46.2	39.5	33.8	39
gpt-o3-reasoning	46.1	42.5	37.8	41.6	44.1	36.8	30.8	36.4	**44.7**	31.5	17.8	29.7	45	38.1	32.5	37.7
Kimi-K2-Thinking	43.2	38.9	35.6	38.7	41.8	35.7	28.2	34.4	42	29.3	17.4	28	43.7	36.4	30.9	36.2
Actre(Qwen3-8B)	38.5	33.2	29.8	33.3	36.4	29.5	19.8	27.6	39.4	26.8	12.3	24.5	41.8	34.7	29.2	34.4
Actre(gpt-4o)	**47.8**	**44.2**	**38.5**	**43**	**47.6**	**41.8**	**35.4**	**40.9**	44.6	**35.7**	**21.4**	**32.6**	**48.9**	**42.3**	**36.5**	**41.8**

Table 1: Performance comparison across four illustration question types and three complexity levels. The parentheses in the last two rows indicate the base LLM used as the Actor in Actre.

4.2 Main Results

As shown in Table 1, Actre significantly outperforms all one-shot baselines across most question types and difficulty levels, achieving the highest average scores in all categories. In Geometry, it leads at Lvl-2 and Lvl-3 (above 5.4 over GPT-4o baseline), while narrowly trailing gpt-o3-reasoning at Lvl-1. When initialized with Qwen3-8B, Actre still improves by above 4.0 points over its non-iterative version. Crucially, Actre exhibits the slowest performance degradation as complexity increases—demonstrating greater robustness under cognitive load, especially in high-stakes domains like geometric reasoning and symbolic manga interpretation. We also present distinct capability profiles across models in Appendix D.

4.3 Analysis

Models	Function			
	Lvl-1	Lvl-2	Lvl-3	Avg.
gpt-4o	35.4	28.8	13.2	24.6
gpt-4o (reverse position)	33.6	25	10.6	21.8
DeepSeek-v3	38.1	28.4	13.5	25.2
DeepSeek-v3 (reverse position)	34.9	25.4	10.7	22.3

Table 2: Impact of input position on pair-wise evaluation scores in Function tasks.

Positional Bias in Pair-wise Evaluation. To validate our position-swapped protocol, we compare scores when the candidate image is presented first vs. second. As shown in Table 2, models like gpt-4o and DeepSeek-v3 suffer score drops up to 2.8 points when evaluated second — confirming strong positional bias in LLM judges. Without order reversal, evaluations would be systematically skewed. Our dual-presentation design effectively mitigates this artifact, ensuring fair and reproducible scoring.

Iteration Patterns Across Difficulty Levels. Figure 3 shows Actre's iteration count per task type and difficulty. Geometry demands the most iterations due to spatial precision needs. Natural Scene and Manga require fewer, as they prioritize semantic over geometric fidelity. Though gpt-4o converges faster, both models follow similar trends per task, indicating the framework's behavior is task-driven, not model-dependent.

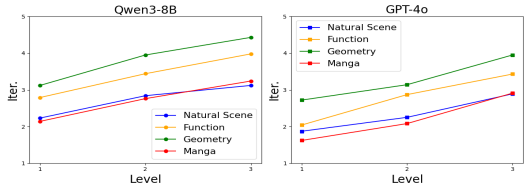


Figure 3: Average number of iterations required by Actre to converge across difficulty levels.

5 Conclusion

We present Actre, a tool-augmented LLM agent that iteratively generates exam-aligned illustrations by emulating human-like drafting and revision. Paired with IllustrationBench, a new benchmark of real exam questions across four types and three difficulty levels, our framework outperforms one-shot baselines—especially on complex geometry and function tasks—demonstrating that iterative refinement is essential for constraint-aware illustration generation in education.

Limitations

Our method is limited to predefined illustration types and tool sandbox, which may hinder accessibility. The evaluation, though bias-mitigated, still depends on LLM judgment and may miss subtle human perceptual nuances. Finally, underspecified or ambiguous questions remain challenging, as the framework assumes sufficient textual grounding for accurate illustration.

References

Zenab Bosheah and Vilmos Bilicki. 2025. Challenges in generating accurate text in images: A benchmark for text-to-image models on specialized content. *Applied Sciences*, 15(5):2274.

Sijia Chen, Yibo Wang, Yi-Feng Wu, Qingguo Chen, Zhao Xu, Weihua Luo, Kaifu Zhang, and Lijun Zhang. 2024. Advancing tool-augmented large language models: Integrating insights from errors in inference trees. *Advances in Neural Information Processing Systems*, 37:106555–106581.

Jiahui Gao, Renjie Pi, Jipeng Zhang, Jiacheng Ye, Wanjun Zhong, Yufei Wang, Lanqing Hong, Jianhua Han, Hang Xu, Zhenguo Li, and 1 others. 2023. G-llava: Solving geometric problem with multi-modal large language model. *arXiv preprint arXiv:2312.11370*.

Kanika Goswami, Puneet Mathur, Ryan Rossi, and Franck Dernoncourt. 2025. Plotgen: Multi-agent llm-based scientific data visualization via multimodal feedback. *arXiv preprint arXiv:2502.00988*.

Hawon Jeong, ChaeHun Park, Jimin Hong, Hojoon Lee, and Jaegul Choo. 2025. The comparative trap: Pairwise comparisons amplifies biased preferences of llm evaluators. In *Proceedings of the 8th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 79–108.

Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip Torr. 2019. Controllable text-to-image generation. *Advances in neural information processing systems*, 32.

Sumeet Ramesh Motwani, Chandler Smith, Rocktim Jyoti Das, Rafael Rafailov, Ivan Laptev, Philip HS Torr, Fabio Pizzati, Ronald Clark, and Christian Schroeder de Witt. 2024. Malt: Improving reasoning with multi-agent llm training. *arXiv preprint arXiv:2412.01928*.

Huy A Nguyen, Shravya Bhat, Steven Moore, Norman Bier, and John Stamper. 2022. Towards generalized methods for automatic question generation in educational domains. In *European conference on technology enhanced learning*, pages 272–284. Springer.

Aaron Parisi, Yao Zhao, and Noah Fiedel. 2022. Talm: Tool augmented language models. *arXiv preprint arXiv:2205.12255*.

Charulata Patil and Manasi Patwardhan. 2020. Visual question generation: The state of the art. *ACM Computing Surveys (CSUR)*, 53(3):1–22.

Oleksandr Polozov, Eleanor O'Rourke, Adam M Smith, Luke Zettlemoyer, Sumit Gulwani, and Zoran Popovic. 2015. Personalized mathematical word problem generation. In *IJCAI*, pages 381–388.

Shraman Pramanick, Rama Chellappa, and Subhashini Venugopalan. 2024. Spiqa: A dataset for multimodal question answering on scientific papers. *Advances in Neural Information Processing Systems*, 37:118807–118833.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, and 1 others. 2023. Toolllm: Facilitating large language models to master 16000+ real-world apis. *arXiv preprint arXiv:2307.16789*.

Changle Qu, Sunhao Dai, Xiaochi Wei, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, Jun Xu, and Ji-Rong Wen. 2025. Tool learning with large language models: A survey. *Frontiers of Computer Science*, 19(8):198343.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. Zero-shot text-to-image generation. In *International conference on machine learning*, pages 8821–8831. Pmlr.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36:68539–68551.

Anselm R Strohmaier, Kelsey J MacKay, Andreas Obersteiner, and Kristina M Reiss. 2020. Eye-tracking methodology in mathematics education research: A systematic literature review. *Educational Studies in Mathematics*, 104(2):147–200.

Stefan Udristoiu, Anca Ion, and Dan Mancas. 2010. Rule based modelling of images semantic concepts. In *International Conference on Agents and Artificial Intelligence*, volume 2, pages 540–543. SCITEPRESS.

Zichao Wang, Jakob Valdez, Debshila Basu Mallick, and Richard G Baraniuk. 2022. Towards human-like educational question generation with large language models. In *International conference on artificial intelligence in education*, pages 153–166. Springer.

Yiran Wu, Feiran Jia, Shaokun Zhang, Hangyu Li, Erkang Zhu, Yue Wang, Yin Tat Lee, Richard Peng, Qingyun Wu, and Chi Wang. 2023. Mathchat: Converse to tackle challenging math problems with llm agents. *arXiv preprint arXiv:2306.01337*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*.

Di Zhang, Xiaoshui Huang, Dongzhan Zhou, Yuqiang Li, and Wanli Ouyang. 2024. Accessing gpt-4 level mathematical olympiad solutions via monte carlo tree self-refine with llama-3 8b. *arXiv preprint arXiv:2406.07394*.

A Implementation Details of the Proposition Illustration Sandbox

To bridge the gap between abstract textual propositions and precise visual outputs, we developed a specialized **Proposition Illustration Sandbox**. This sandbox provides a set of unified interfaces that translate high-level agent instructions into deterministic symbolic renderings or high-fidelity generative content, specifically tailored to the four task categories in IllustrationBench.

A.1 Mathematical Function Rendering Engine

The Function Engine is designed for analytical rigor, ensuring that generated **Function Graphs** faithfully represent the mathematical properties of the question.

- **Symbolic Transformation:** The engine accepts standard LaTeX expressions and utilizes a symbolic parsing layer to convert expressions into evaluable functions via `NumPy`.
- **High-Resolution Sampling:** To ensure smoothness and capture subtle local extrema, the engine performs equidistant sampling of $N = 1000$ points within the agent-specified domain $[x_{min}, x_{max}]$.
- **Contextual Annotation:** Beyond curve plotting via `Matplotlib`, the tool supports specific coordinate highlighting. Given x-coordinates, it automatically calculates $y = f(x)$, renders the points with markers, and adds text labels.

A.2 Constraint-Aware Geometric Construction Tool

Unlike generic image generators, this tool operates on a **Coordinate-Based Synthesis** logic to guarantee topological correctness for **Geometric Diagrams**.

- **Spatial Mapping:** The agent provides a relative coordinate system $\mathcal{P} = \{(x_i, y_i, z_i)\}$ for all vertices. For 3D solids, the engine applies an isometric projection to map 3D spatial points onto a 2D canvas while maintaining parallel relationships.
- **Structural Differentiation:** The tool implements a dual-path rendering pipeline. *Visible Paths* are rendered as solid lines, while *Occluded Paths* (e.g., hidden edges in a frustum) are rendered as dashed lines (`linestyle='-'`), preserving the perspective integrity of the geometric body.

A.3 Task-Specific Generative Pipelines

For non-symbolic illustrations, we utilize Latent Diffusion Models (LDMs) with checkpoints specialized for distinct educational scenarios:

- **Realistic Vision Pipeline (Natural Scene Tasks):** To address **Natural Scene Illustrations** (e.g., real-world physics contexts or architectural scenarios), we employ the *Realistic Vision* checkpoint. It is optimized for photorealistic accuracy and environmental logic, translating descriptive prompts into scenes that maintain the proportions and lighting required for context-grounded exam questions.
- **Anything XL Pipeline (Manga/Narrative Tasks):** For **Manga-style Illustrations** that involve abstract metaphors or narrative cartoons, we utilize the *Anything XL* architecture. This model is fine-tuned on a distinct illustrative aesthetic distribution, effectively capturing the symbolic intent and artistic conventions of educational manga where photorealism is secondary to metaphorical clarity.

B Difficulty Annotation Criteria

To enable granular analysis of model performance across cognitive tiers, each question in IllustrationBench is annotated with a difficulty level (L1–L3) based on the *graphical reasoning burden* imposed on the agent—i.e., how challenging it is to generate a correct illustration from the textual prompt alone. The detailed criteria for each illustration type are as follows:

Geometric Diagrams

- **L1**: Basic planar or solid shapes (e.g., cube, triangular pyramid) with explicitly specified vertex coordinates and no auxiliary constructions. Requires minimal inference and direct tool invocation.
- **L2**: Stereometric figures involving up to three auxiliary points or construction lines (e.g., altitudes, medians). Demands spatial reasoning to resolve implicit geometric constraints.

- **L3**: Complex configurations with more than three auxiliary elements, dynamic points (e.g., locus problems), or underspecified geometry requiring heuristic inference (e.g., "a point P such that $\angle APB = 90°$").

Function Graphs

- **L1**: Fully specified function expressions with explicit coefficients (e.g., $f(x) = 2x^2 - 3x + 1$). No symbolic derivation needed.
- **L2**: Partially specified functions requiring contextual inference to recover the complete form (e.g., "the function passes through $(1, 2)$ and has slope 3 at $x = 0$").
- **L3**: Open-ended or qualitatively described functions (e.g., "sketch a cubic polynomial with one local maximum and one inflection point"), where the agent must choose valid parameters or interpret abstract constraints.

Natural Scene Illustrations

- **L1**: Depiction of isolated, well-known entities without relational constraints (e.g., "the Leaning Tower of Pisa"). Focuses on object recognition and basic scene composition.
- **L2**: Scenes involving size, scale, or spatial relationships (e.g., "a statue whose proportions follow the golden ratio"). Requires proportional or geometric reasoning.
- **L3**: Illustrations governed by compositional or causal constraints (e.g., "a bowl whose pattern matches the geometric properties described in the question"), demanding environmental logic and coherent perspective.

Manga-style Illustrations

- **L1**: Literal rendering of described actions or objects (e.g., "a person holding a stack of books"). Involves minimal symbolic abstraction.
- **L2**: Representation of simple metaphors or emotional states (e.g., "several soldiers standing in a line"). Requires basic iconographic fluency.
- **L3**: Illustrations conveying layered meaning, irony, or narrative progression (e.g., "a Qing-dynasty figure with a long braid, used to satirize corruption"), demanding cultural-literary inference and expressive stylistic control.

This tiered annotation schema supports both holistic performance evaluation and fine-grained ablation studies along distinct cognitive and representational dimensions.

C Evaluation Weights

To ensure a nuanced and domain-appropriate evaluation, we assign distinct weights to three core scoring dimensions for each of the four question types, as shown in Table 3.

Question Type	Scoring Dimensions
Geometric Diagram	`"Text-Image Relevance": 0.3,` `"Information Completeness": 0.3,` `"Spatial Reasonableness": 0.4`
Function Graph	`"Text-Image Relevance": 0.3,` `"Information Completeness": 0.3,` `"Graph Accuracy": 0.4`
Natural Scene	`"Text-Image Relevance": 0.4,` `"Information Completeness": 0.3,` `"Logical Reasonableness": 0.3`
Manga Illustration	`"Text-Image Relevance": 0.4,` `"Information Completeness": 0.3,` `"Depth Alignment": 0.3`

Table 3: Scoring Dimensions by Question Type

D Multidimensional Model Capability Profiling

The radar charts in Figure 4 reveal distinct capability profiles across models. In Figure 4(a), larger parameter models (e.g., Qwen3-235B) dominate across all dimensions, but even smaller models like Qwen3-8B show balanced performance — hinting at architectural efficiency. More revealingly, Figure 4(b) contrasts reasoning-enhanced models (DeepSeek-r1, gpt-o3-reasoning) with their non-reasoning counterparts. The former consistently outperform in Function and Geometry — domains requiring symbolic manipulation and constraint satisfaction — validating that explicit reasoning capabilities directly translate to superior tool-use performance in proposition illustration tasks.

E Data Examples

Fig 3, 4 presents sample entries from the dataset, grouped together by difficulty levels (Level 1–3). Each example includes a Question (partially truncated for layout brevity, with non-essential content omitted) and a Reference Image.

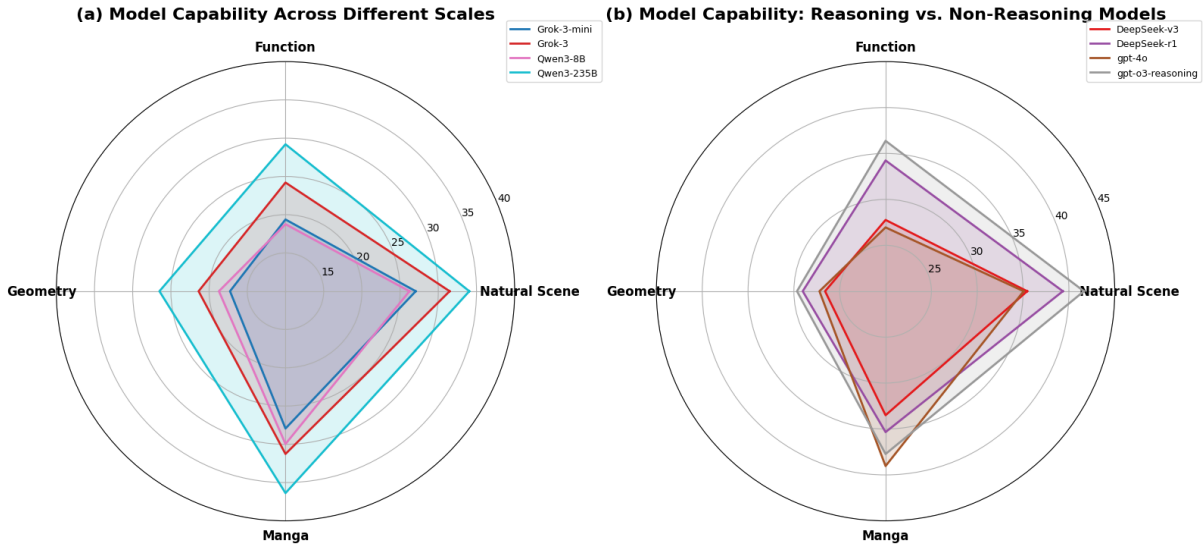


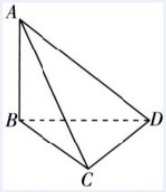
Figure 4: (a) Model capability comparison across four illustration scenarios, showing scale-dependent performance. (b) Reasoning-enhanced models (purple, gray) outperform their non-reasoning counterparts in Function and Geometry, highlighting the value of structured reasoning for tool-augmented visual generation.

Level 1

Text

Question: In tetrahedron ABCD, AB ⊥ CD, AB = CD = 1, BD = √2, BC = AD = √3. How many pairs of perpendicular faces exist in the tetrahedron?

Reference Image

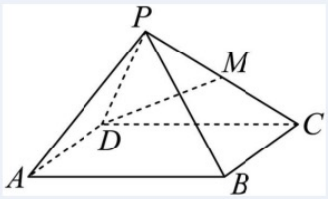


Level 2

Text

Question: In a pyramid P–ABCD, quadrilateral ABCD is a rhombus, ∠BAD = 60°, triangle PAD is an equilateral triangle, and M is the midpoint of edge PC. Let line l be the intersection of planes PAD and PBC.
(1) Prove: l // plane ABCD;
(2) Prove: BC ⊥ DM;

Reference Image



Level 3

Text

Question: As shown in the figure, PA ⊥ plane ABCD, the base ABCD is a rectangle, PA = AB = (1/2)BC = 1, and point E is the midpoint of PB.
(1) Prove: AE ⊥ PC.
(2) If points M and N are on PD and AC respectively, such that PM/DM = AN/CN = 2, and Q is any point on MN, determine whether the volume of triangular pyramid P–ABQ is constant. If yes, prove it and find the constant value; if not, explain why.

Reference Image

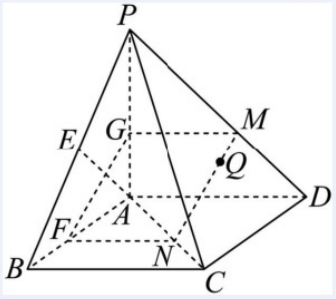


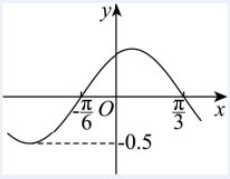
Figure 5: Geometry Examples

Level 1

Text

Given the function $f(x) = A\cos(\omega x + \varphi)$, where $A > 0$, $\omega > 0$, and $|\varphi| < \pi/2$, and part of its graph is shown in the figure. .... Analysis: ...Therefore, the function is $f(x) = 0.5\cos(2x - \pi/6)$,

Reference Image

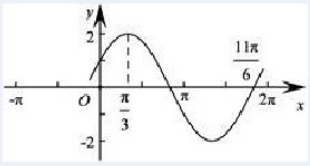


Level 2

Text

Given the function $f(x) = A\sin(\omega x + \varphi)$, where $A > 0$ and $|\varphi| < \pi/2$, its graph is shown in the figure. ... What is the smallest possible value of $|x_0|$?
Analysis: From the problem and the graph, we can see that the amplitude $A = 2$.
The period T is calculated as $(4/3) \times (11\pi/6 - \pi/3) = 2\pi$. Therefore, $\omega = 2\pi / T = 1$. ...

Reference Image



Level 3

Text

As shown in the figure is the graph of $y = f'(x)$. Then, what are the intervals on which the function $y = f(x)$ is decreasing ?
Analysis: We determine this based on the relationship between the derivative and monotonicity. From the graph of the derivative, we see that $f'(x) < 0$ when $-2 < x < 0$ or $x > 2$.
Therefore, the decreasing intervals of $f(x)$ are $(-2, 0)$ and $(2, +\infty)$.

Reference Image

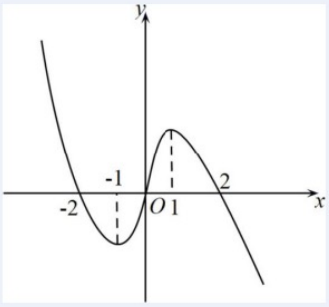


Figure 6: Function Examples