LatentGeo: Learnable Auxiliary Constructions in Latent Space for Multimodal Geometric Reasoning

Anonymous ACL submission

Abstract

Despite recent progress in multimodal mathematical reasoning, geometry problems that require auxiliary constructions remain difficult for MLLMs because key visual structures are absent from the input diagram and must be introduced during reasoning. Existing approaches that externalize such constructions through textual descriptions, sketches, or tools often lose spatial precision, introduce execution overhead, or weaken end-to-end optimization. Latent visual reasoning provides a natural alternative by aligning hidden states with visual information, but its use for synthesizing new geometric structures remains underexplored. We propose **LatentGeo**, a framework that represents auxiliary constructions as continuous latent visual states within a standard autoregressive sequence. LatentGeo generates a construction plan, latent construction tokens, and the final reasoning in a unified generation process. The latent representations are trained with auxiliary diagram supervision through a three-stage curriculum and stabilized by LaGDPO during policy optimization. To enable systematic evaluation, we introduce **GeoAux**, a benchmark organized by construction operation and knowledge domain. Experiments on geometric visual reasoning tasks show that LatentGeo achieves superior performance on problems requiring auxiliary constructions while maintaining competitive transfer performance on public multimodal math benchmarks. These results demonstrate that latent representations can support the construction of new geometric elements and transformations over an existing diagram, such as auxiliary lines and spatial projections.

1 Introduction

While multimodal large language models (MLLMs) have shown strong progress on visual mathematical reasoning (Lu et al., 2024; Zhang et al., 2024), geometry remains a particularly challenging setting. Many problems require *auxiliary constructions*, such as auxiliary lines, projections, or unfolded views, that are absent from the original diagram but must be introduced before relevant theorems and spatial relations become accessible. This makes geometric reasoning a form of *constructive* visual reasoning, rather than merely perception of existing diagram content.

Several paradigms provide partial solutions, as illustrated in Fig. 1. Text-only reasoning can describe constructions but often loses spatial fidelity; explicit sketching or tool-augmented execution externalizes them (Hu et al., 2024; Wang et al., 2025a), but adds rendering or execution overhead. Latent visual reasoning offers a closer alternative by showing that hidden states can carry intermediate visual intent (Yang et al., 2025b; Zhang et al., 2025a; Wang et al., 2025d). However, existing work mostly interprets, revisits, or refines visual content already present in the input, whereas geometry requires synthesizing new elements and transformations over the given diagram.

To address this gap, we propose **LatentGeo**, a framework that internalizes auxiliary geometric constructions as continuous latent construction representations. LatentGeo inserts latent construction tokens into the autoregressive reasoning sequence: after a textual construction plan, these tokens form an implicit visual workspace before the model produces the solution and answer. During training, auxiliary-construction diagrams supervise the corresponding latent construction representations, and a three-stage curriculum internalizes this grounding for inference without auxiliary diagrams. Since subsequent policy optimization can destabilize sparse latent construction tokens, we further introduce LaGDPO to preserve latent construction token usage while improving end-task correctness.

We further introduce **GeoAux**, a construction-centric benchmark for geometry problems requiring missing intermediate visual structures. Exist-

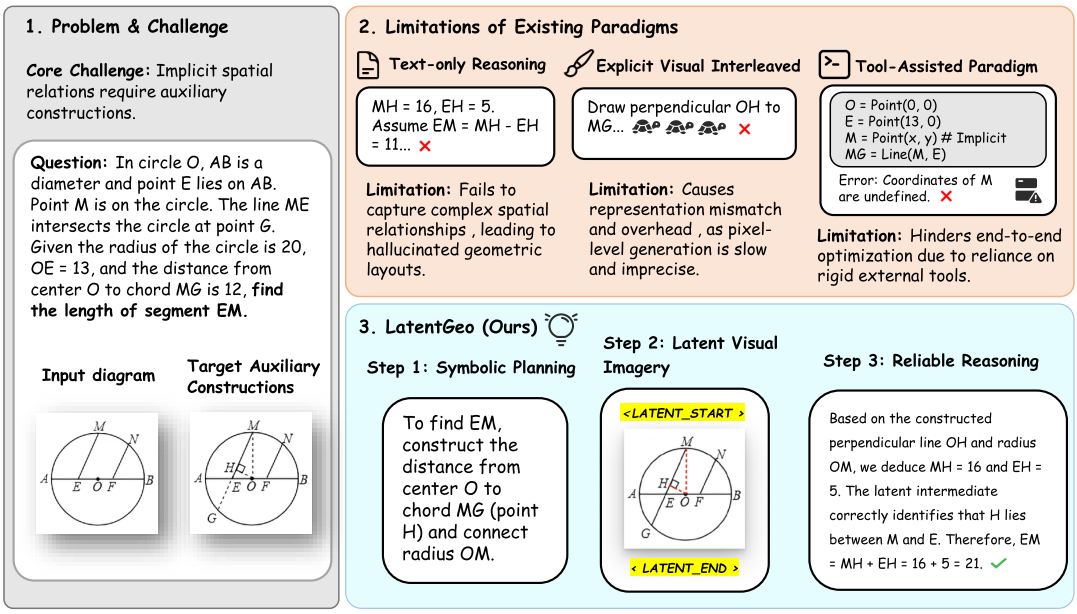


Figure 1: Motivating comparison of auxiliary-construction paradigms for geometric reasoning.

ing multimodal math benchmarks are broad and mainly report aggregate accuracy; their geometry subsets are not organized around Visual Operations. GeoAux addresses this gap by categorizing problems by both Knowledge Domain and Visual Operation, enabling analysis of construction-dependent reasoning beyond final-answer accuracy. Experiments show strong gains on GeoAux, especially on construction-heavy operations such as angular construction, spatial projection, and connectivity. We also evaluate on two public multimodal math benchmarks, MathVista and MathVerse, to assess whether specialization on GeoAux preserves broader visual-math capability.

Our main contributions can be summarized as follows:

- **Latent Constructive Paradigm:** We represent auxiliary geometric constructions as latent construction representations instead of rendered images or external tool calls.
- **Curriculum Internalization Strategy:** We design a three-stage curriculum that internalizes auxiliary-diagram supervision for inference-time latent construction.
- **Latent-Aware Policy Optimization:** We introduce LaGDPO to mitigate latent construction token collapse and stabilize construction-aware reasoning.
- **Construction-Centric Benchmark:** We construct GeoAux for visual-operation-level auxiliary construction evaluation.

2 Related Work

2.1 Multimodal Mathematical and Geometric Reasoning

Multimodal mathematical reasoning studies problems that combine textual questions with visual contexts such as diagrams, plots, and tables. Representative benchmarks, including MathVista (Lu et al., 2024), MathVerse (Zhang et al., 2024), MATHVISION (Wang et al., 2024), and MV-MATH (Wang et al., 2025b), cover diverse visual-mathematical tasks and input formats. SOLID-GEO (Wang et al., 2025c) further targets solid geometry and 3D spatial reasoning, revealing persistent challenges in diagram-grounded deduction and auxiliary constructions. These benchmarks are valuable, but their geometry portions are not organized around constructive operations such as missing connections, alignments, projections, or unfoldings.

Recent methods improve visual-mathematical reasoning through explicit grounding or externalized intermediate representations. MathCoder-VL (Wang et al., 2025a) aligns diagrams with executable code, E-GPS (Wu et al., 2024) formalizes textual and visual inputs for theorem reasoning, Geoint-R1 (Wei et al., 2025) and GeometryZero (Wang et al., 2026) introduce auxiliary constructions via verification or learned decision strategies, and Visual Sketchpad (Hu et al., 2024) externalizes geometric artifacts through sketching. MAVIS (Zhang et al., 2025c), selective visual revis-

itation (Chung et al., 2025), and SVE-Math (Zhang et al., 2025d) instead emphasize diagram perception and visual evidence reuse. Together, these efforts highlight the importance of grounding and auxiliary operations, but they either externalize the construction as code, sketches, or verification steps, or improve perception of the given diagram. This leaves open how to obtain a faithful intermediate construction without leaving the model's generative process.

2.2 Visual Latent Reasoning

Latent reasoning performs inference in continuous hidden spaces instead of relying only on explicit natural-language chains, reducing dependence on surface-form reasoning traces (Sun et al., 2025). In multimodal settings, latent states can also carry intermediate visual intent. Latent-CoT formulates multimodal reasoning as latent-variable inference (Sun et al., 2025); Mirage (Yang et al., 2025b) and Latent Visual Reasoning (Li et al., 2025) interleave textual decoding with latent visual embeddings; Chain-of-Visual-Thought compresses perceptual cues into latent tokens (Qin et al., 2025); Latent Sketchpad refines latent sketches iteratively (Zhang et al., 2025a); and Monet combines multimodal chain-of-thought distillation with reinforcement learning over latent visual tokens (Wang et al., 2025d).

These works show that latent representations can serve as visual intermediates, but they primarily support *perceptual* or refinement behavior: interpreting, revisiting, or refining visual content already present in the input. Geometric auxiliary construction is instead *constructive*: the model must introduce new elements and transformations, such as auxiliary lines, perpendiculars, projections, rotations, or unfolded views, before the relevant spatial relations become available. LatentGeo fills this gap by specializing latent visual reasoning into latent construction representations, supervised by auxiliary diagrams during training and internalized for inference without rendered intermediates.

3 Method

In this section, we present the full design of **LatentGeo** for visually grounded geometric reasoning with auxiliary constructions. Unlike perceptual tasks that interpret existing visual content, our focus is on *constructive* geometric operations that introduce new elements into the reasoning process. We first describe latent auxiliary construction, which inserts latent construction tokens into the autoregressive reasoning sequence and grounds their hidden states through auxiliary visual supervision, as illustrated in Fig. 2. We then introduce a three-stage curriculum learning strategy that progressively internalizes auxiliary construction ability when ground-truth auxiliary diagrams are unavailable at inference time. Next, we detail **LaGDPO**, a latent-aware group-decoupled policy optimization procedure that improves end-task utility with KL-regularized updates while preserving the learned latent construction representations and inference paradigms. Finally, we describe the data construction protocol that separates auxiliary-construction supervision from held-out construction-centric evaluation.

3.1 Latent Auxiliary Construction

Building on recent visual latent reasoning ideas, LatentGeo starts from Qwen2.5-VL (Bai et al., 2025b) and adapts the latent visual-token supervision idea explored in Mirage (Yang et al., 2025b): intermediate visual intent can be represented by hidden states at special token positions rather than by decoded pixels. While Mirage focuses on general latent visual imagination, LatentGeo specializes this idea for *constructive* geometric reasoning, where the model must introduce auxiliary elements that are absent from the original diagram. We therefore augment the tokenizer with a compact set of latent construction tokens and use their hidden states as an internal workspace for operations such as adding auxiliary lines, projections, rotations, and structural partitions.

Given an image-question pair $x = (\mathcal{I}, \mathcal{Q})$, LatentGeo generates a structured response

$$y = [\mathcal{T}_{plan}, s_{lat}, p_{lat}^{\times K}, e_{lat}, \mathcal{R}, \mathcal{A}], \qquad (1)$$

where $\mathcal{T}_{plan}$ describes the intended auxiliary construction, $\mathcal{R}$ is the subsequent reasoning trace, and $\mathcal{A}$ is the final answer. Here s_{lat}, p_{lat}, and e_{lat} denote `<|latent_start|>`, `<|latent_pad|>`, and `<|latent_end|>`, respectively. We denote the hidden states at the K latent pad positions as $\mathcal{Z} = \{\mathbf{z}_k\}_{k=1}^{K}$ and treat them as latent construction representations. Thus, the sequence follows a construct-then-reason workflow: the plan specifies the intended operation, the latent construction tokens provide a continuous construction-aware intermediate, and the final reasoning consumes this intermediate to produce the answer. The full process

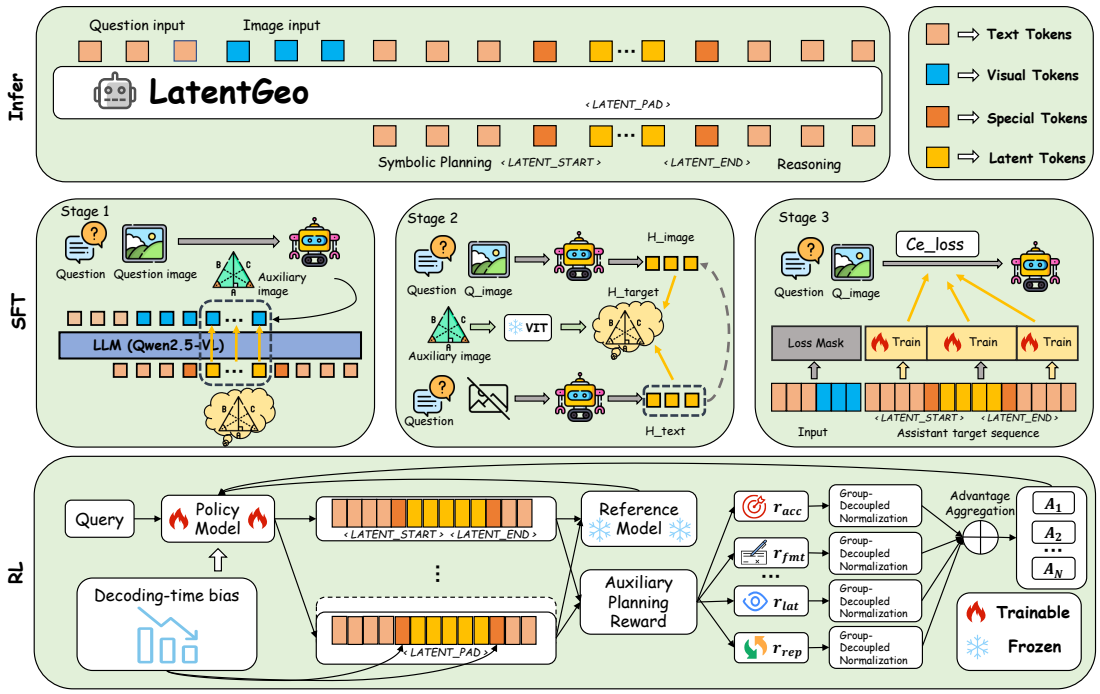


Figure 2: Overview of the **LatentGeo** framework.

remains within standard autoregressive generation, without rendering auxiliary images or invoking external geometric executors.

To ground the latent construction representations in geometric structure, we supervise the hidden states at the latent token positions. Let $\mathbf{H}_{gen} \in \mathbb{R}^{K\times D}$ denote the model hidden states corresponding to the generated latent construction tokens. During training, each problem is paired with a ground-truth auxiliary diagram $\mathcal{I}_{aux}$, which provides the visual target for the intended construction. We encode $\mathcal{I}_{aux}$ using the frozen visual encoder f_{enc}, apply patch-level pooling to obtain K visual tokens, and map them into the LLM hidden space with a lightweight projector Φ:

$$\mathbf{H}_{target} = \Phi\big(\mathrm{Pool}(f_{enc}(\mathcal{I}_{aux}))\big) \in \mathbb{R}^{K\times D}. \quad (2)$$

This target representation gives the latent construction representations a compact but spatially grounded supervision signal without requiring the model to decode or render an auxiliary image.

We align the generated latent states with the auxiliary-diagram target using a hybrid objective:

$$\mathcal{L}_{align} = \lambda_{cos}\Big(1 - \cos(\mathbf{H}_{gen}, \mathbf{H}_{target})\Big) + \lambda_{mse}\|\mathbf{H}_{gen} - \mathbf{H}_{target}\|_2^2. \quad (3)$$

The cosine term encourages semantic alignment in the hidden space, while the MSE term stabilizes representation scale and token-level consistency. Importantly, $\mathcal{I}_{aux}$ and $\mathbf{H}_{target}$ are used only during training. At inference time, LatentGeo must generate the latent construction tokens solely from the original diagram, the question, and its own construction plan. The next subsection describes how our curriculum progressively transfers this supervised latent grounding into autonomous auxiliary construction.

3.2 Multi-stage Curriculum Learning Strategy

Learning this structured construct-then-reason sequence end-to-end is challenging due to the modality gap between textual construction plans and visual auxiliary structures. We therefore use a three-stage curriculum that gradually reduces reliance on ground-truth auxiliary diagrams: Stage 1 aligns latent construction representations to auxiliary visual targets, Stage 2 transfers this grounding from auxiliary diagrams to construction plans, and Stage 3 removes auxiliary supervision to match inference-time generation.

Stage 1: Auxiliary-Visual Latent Alignment. In the initial stage, we ground latent construction tokens in auxiliary construction semantics. The LLM backbone, projector Φ, and latent construction token embeddings are jointly trained, while only the visual encoder (ViT) is kept frozen throughout all stages to preserve pretrained perceptual features. Given the question image $\mathcal{I}$ and construction plan

$\mathcal{T}_{plan}$, the model performs a forward pass and produces hidden states $\mathbf{H}_{gen}$ at the latent token positions. Simultaneously, $\mathcal{I}_{aux}$ is encoded by the frozen ViT and compressed via Φ into the target representation $\mathbf{H}_{target}$ (Eq. 2). The objective is to minimize $\mathcal{L}_{align}$, driving the LLM's hidden space at latent positions to encode the visual semantics of auxiliary constructions. Crucially, by optimizing the full LLM rather than a shallow adapter, this stage shapes the model's internal representation space to natively support constructive geometric reasoning.

Stage 2: Plan-Guided Latent Internalization. Stage 1 establishes visual grounding under full auxiliary supervision, but $\mathcal{I}_{aux}$ is unavailable at inference. Stage 2 bridges this gap by training the model so that $\mathcal{T}_{plan}$ becomes a reliable semantic controller for latent construction. We run two parallel forward passes per batch, both supervised against $\mathbf{H}_{target}$ computed from $\mathcal{I}_{aux}$ via the frozen ViT and Φ:

- **Visual-Conditioned Stream:** Receives $\mathcal{I}$ and $\mathcal{T}_{plan}$, producing $\mathbf{H}_{img}$ supervised by $\mathcal{L}_{sim,img} = \mathcal{L}_{align}(\mathbf{H}_{img}, \mathbf{H}_{target})$. This stream serves as a well-anchored reference signal.
- **Plan-Conditioned Stream:** Receives only $\mathcal{T}_{plan}$, removing direct visual input and forcing the construction plan to serve as the primary driver of latent construction. It produces $\mathbf{H}_{txt}$ supervised by $\mathcal{L}_{sim,txt} = \mathcal{L}_{align}(\mathbf{H}_{txt}, \mathbf{H}_{target})$.

An asymmetric consistency loss further aligns $\mathbf{H}_{txt}$ toward $\mathbf{H}_{img}$, with stop-gradient applied to the latter's latent states to prevent collapse:

$$\mathcal{L}_{cons} = 1 - \cos(\mathbf{H}_{txt},\ \mathrm{sg}(\mathbf{H}_{img}))\,. \tag{4}$$

A small CE term stabilizes language capability during representation alignment. The full Stage 2 loss is:

$$\mathcal{L}_2 = 0.1\,\mathcal{L}_{CE} + \mathcal{L}_{sim,img} + \mathcal{L}_{sim,txt} + \mathcal{L}_{cons}. \tag{5}$$

By end of Stage 2, the model reduces its dependence on $\mathcal{I}_{aux}$ and improves the plan-conditioned controllability of the latent space.

Stage 3: Inference-Matched Constructive Reasoning. In the final stage, all auxiliary visual supervision is removed. The model is trained end-to-end with cross-entropy over the complete assistant output, covering $\mathcal{T}_{plan}$, $\mathcal{Z}$, and $\mathcal{A}$ in a single unified sequence. Latent construction tokens still participate in token-level supervision alongside the reasoning text; what is absent is the auxiliary visual target $\mathbf{H}_{target}$ and the alignment loss $\mathcal{L}_{align}$. Training and inference now follow the same generation protocol, completing the transition from auxiliary-guided latent grounding to autonomous constructive reasoning.

3.3 LaGDPO: Latent-aware Group-Decoupled Policy Optimization

While the three-stage supervised curriculum provides a strong cold start for latent auxiliary construction, it still optimizes token-level likelihood rather than the final objective of geometric problem solving. We therefore further optimize LatentGeo with a GRPO-style reinforcement learning stage, where multiple responses are sampled for each prompt and updated according to group-relative rewards. However, directly applying standard GRPO is unstable in our setting. First, the reward contains heterogeneous objectives, including answer correctness, output format, latent construction token validity, length control, and repetition suppression; normalizing only the aggregated reward can let one component dominate the update. Second, high-temperature rollouts may omit the sparse latent boundary tokens, causing latent construction token usage to receive little learning signal and gradually degrade into pure textual reasoning. To address these issues, we propose **LaGDPO**, which retains the critic-free group optimization framework of GRPO while introducing component-wise reward normalization and latent-aware decoding stabilization.

KL-Regularized Policy Update. Given an input pair $x = (\mathcal{I}, \mathcal{Q})$, the policy π_θ generates a complete response y containing the construction plan, latent construction tokens, reasoning trace, and final answer. We initialize the reference policy π_{ref} from the Stage-3 checkpoint and optimize the actor with a KL-regularized policy-gradient objective:

$$\mathcal{L}_{\text{LaGDPO}}(\theta) = \mathcal{L}_{\text{PG}}(\theta) + \beta\,\mathcal{L}_{\text{KL}}(\theta), \tag{6}$$

where $\mathcal{L}_{\text{PG}}$ follows the standard clipped policy-gradient form and $\mathcal{L}_{\text{KL}}$ constrains the policy from drifting away from π_{ref}. The main distinction of LaGDPO lies in how the rollout rewards are normalized and how latent construction token generation is stabilized during sampling.

Construction-Aware Reward. For each completed response, we compute an outcome reward composed of five terms:

$$R(x, y) = r_{\text{acc}}(y, \mathcal{A}^*) + r_{\text{fmt}}(y) + r_{\text{lat}}(y) + r_{\text{len}}(y) + r_{\text{rep}}(y), \tag{7}$$

where $\mathcal{A}^*$ is the ground-truth answer. The accuracy reward r_{acc} checks the predicted final answer against $\mathcal{A}^*$, and r_{fmt} encourages a parseable final-answer format. The latent reward encourages the model to invoke exactly one well-formed latent construction token span. Let $n_{\text{lat}}(y)$ denote the number of latent token spans delimited by the boundary tokens; we define

$$r_{\text{lat}}(y) = 0.5\,\mathbb{I}[n_{\text{lat}}(y) = 1] - 0.2\,\mathbb{I}[n_{\text{lat}}(y) \neq 1]. \tag{8}$$

The length and repetition terms penalize overly verbose or degenerate responses. Their coefficients and thresholds are listed in Appendix A.3.

Group-Decoupled Advantage Estimation. For each prompt x_i, we sample N responses $\{y_i^{(j)}\}_{j=1}^N$ and compute every reward component in Eq. 7. Standard GRPO normalizes the aggregated scalar reward within the group. In contrast, LaGDPO first normalizes each reward component separately:

$$\tilde{r}_{i,k}^{(j)} = \frac{r_{i,k}^{(j)} - \mu_{i,k}}{\sigma_{i,k} + \epsilon}, \quad \mu_{i,k} = \frac{1}{N}\sum_{j=1}^{N} r_{i,k}^{(j)}, \tag{9}$$

where $k \in \{\text{acc}, \text{fmt}, \text{lat}, \text{len}, \text{rep}\}$. We then aggregate the normalized components as $s_i^{(j)} = \sum_k \tilde{r}_{i,k}^{(j)}$ and whiten $s_i^{(j)}$ over the mini-batch to obtain the final advantage. This group-decoupled estimator prevents high-magnitude auxiliary terms from overwhelming answer correctness or latent token-span validity, while preserving the critic-free group-relative optimization of GRPO.

Latent-Aware Decoding Stabilization. During early RL rollouts, the latent boundary tokens are sparse and can be skipped under high-temperature sampling. Once omitted, the model receives little signal for latent construction, which accelerates collapse toward text-only reasoning. We therefore apply an adaptive logit bias to `<|latent_start|>` and `<|latent_end|>` only during rollout sampling. At training step g, we maintain an exponential moving average of the batch mean latent reward:

$$\begin{aligned} m_g &= \rho m_{g-1} + (1-\rho)\bar{r}_{\text{lat},g}, \\ b_g &= \max\big(b_{\min},\, b_0 \exp(-\lambda m_g)\big). \end{aligned} \tag{10}$$

The bias b_g is added to the logits of the two latent boundary tokens. As the policy learns to produce a valid single latent construction token span, $\bar{r}_{\text{lat},g}$ increases and the bias decays automatically, so the intervention stabilizes early exploration without imposing a fixed decoding pattern at convergence.

3.4 Data Construction and Protocol

Training Data Construction. LatentGeo uses auxiliary-construction supervision only during supervised training. We format each SFT instance as $(\mathcal{I}, \mathcal{Q}, \mathcal{T}_{plan}, \mathcal{I}_{aux}, \mathcal{R}, \mathcal{A})$, where $\mathcal{I}_{aux}$ serves only as a training target for latent alignment and is never provided at inference time. Thus, auxiliary diagrams supervise the latent construction representations rather than becoming an additional inference input. In the RL stage, we use only problem prompts and final answers as outcome supervision. Further dataset composition and formatting details are given in Appendix A.5.

GEOAUX Benchmark Construction. GEOAUX is a held-out diagnostic benchmark for construction-dependent geometry reasoning. Frontier LLMs are used only to propose candidate construction requirements, after which human annotators verify geometric validity, answer uniqueness, diagram clarity, and construction necessity. This separates LLM-assisted candidate generation from the final evaluation labels. Retained instances are labeled by Knowledge Domain and Visual Operation, and near duplicates against the SFT/RL sources are removed. The final benchmark contains 2,228 instances; detailed statistics, filtering criteria, and representative examples are provided in Appendix A.2.

4 Experiments

4.1 Experimental Settings

Baselines For GEOAUX, we compare against representative multimodal math-reasoning models and geometry-oriented methods, including closed-source models GPT-4o-mini (OpenAI, 2024), GPT-4o (OpenAI, 2024), and Qwen3-VL (Bai et al., 2025a), as well as open-source baselines LLaVA-1.6 (Liu et al., 2024), MMR1 (Leng et al., 2025), OpenMMReasoner (Zhang et al., 2025b), R1-Onevision (Yang et al., 2025a), Qwen2.5-VL (Bai et al., 2025b), InternVL2 (Chen et al., 2024), and InternVL3.5 (Wang et al., 2025e), under a unified evaluation protocol. For MATHVISTA and MATHVERSE, the corresponding public baselines are re-

Table 1: **Main Results on GeoAux.** Accuracy (%) on 2,228 instances. We compare LatentGeo with closed-source and open-source MLLMs. *Knowledge Domain*: Analytic (Anal.), Euclidean (Eucl.), Functional (Func.), and Spatial (Spat.). *Visual Operation*: Analytic Overlay (Analy.), Angular Construction (Angle), Circular Augmentation (Circ.), Connectivity (Conn.), Transformation (Trans.), Alignment (Align.), Projection (Proj.), and Partitioning (Part.).

Model	Size	Overall	Knowledge Categories				Visual Operations							
		ALL	Anal.	Eucl.	Func.	Spat.	Analy.	Angle	Circ.	Conn.	Trans.	Align.	Proj.	Part.
Closed-source MLLMs														
GPT-4o-mini	–	12.2	14.3	13.9	12.4	6.2	5.3	8.3	7.3	13.1	11.1	13.3	5.5	10.8
GPT-4o	–	30.3	34.3	31.4	33.2	23.4	30.9	25.6	31.8	32.3	26.1	29.6	27.4	25.5
Qwen3-VL-Flash	–	33.4	43.8	34.1	34.3	25.2	27.7	29.3	30.9	32.8	30.5	36.5	21.2	33.1
Open-source MLLMs														
LLaVA-1.6 (Vicuna-7B)	7B	7.1	5.3	6.9	9.0	8.0	7.5	5.3	5.5	6.4	7.5	6.0	13.7	6.8
MMR1-7B-RL	7B	29.7	27.6	33.9	31.5	18.3	28.7	28.6	23.6	29.1	23.9	28.4	23.3	19.3
OpenMMReasoner-RL	7B	10.3	12.5	10.8	16.3	5.2	9.6	8.3	10.0	9.9	11.1	8.9	5.5	5.8
R1-Onevision-7B	7B	13.2	13.6	14.2	17.4	8.4	8.5	15.0	10.0	13.0	15.0	12.2	8.2	11.9
Qwen2.5-VL-7B	7B	14.5	19.3	13.9	22.5	10.1	12.8	6.8	9.1	14.0	18.6	15.1	11.6	12.7
Qwen2.5-VL-32B	32B	33.1	35.5	34.9	37.1	25.2	26.6	30.1	29.1	34.4	30.5	31.7	29.5	27.7
Qwen3-VL-8B	8B	32.7	38.5	33.3	41.0	24.5	27.7	25.6	30.9	32.2	32.3	32.7	26.7	26.7
InternVL2-8B	8B	15.9	18.1	15.2	17.4	16.1	10.6	15.8	14.6	18.1	16.8	13.7	17.1	11.0
InternVL3.5-8B	8B	25.1	24.9	26.9	27.0	19.4	19.2	18.8	18.2	26.1	22.1	23.6	24.0	19.9
Ours														
LatentGeo$_{SFT}$	7B	32.1	26.0	36.1	25.8	26.5	22.3	34.6	27.3	32.2	22.1	29.3	37.0	23.5
LatentGeo$_{LaGDPO}$	7B	**34.6**	30.6	**37.3**	31.5	**30.5**	27.7	**42.1**	24.6	**34.9**	27.0	31.9	**37.0**	**31.5**
Gain vs Baseline	–	+20.1	+11.3	+23.4	+9.0	+20.4	+14.9	+35.3	+15.5	+20.9	+8.4	+16.8	+25.4	+18.8

ported in Appendix Table 3. Detailed baseline configurations and implementation details are provided in Appendix A.7.

Training Setups SFT uses the three-stage curriculum with 5, 2, and 5 epochs, respectively. We set the number of latent construction tokens to $K = 10$ following preliminary validation and prior latent visual reasoning practice (Wang et al., 2025d). The RL phase runs for one epoch on 4×NVIDIA A800 GPUs; detailed hyperparameters are in Appendix A.3.

Evaluation Benchmarks We evaluate LatentGeo on GeoAux and two public multimodal math benchmarks, MathVista (Lu et al., 2024) and MathVerse (Zhang et al., 2024). GeoAux targets construction-centered reasoning, while the public benchmarks test broader transfer; detailed statistics are in Appendix A.2.

4.2 Main Results

Results on GeoAux. Table 1 shows that LatentGeo achieves the best overall accuracy on GeoAux (34.6%), outperforming GPT-4o, Qwen2.5-VL-32B, and its Qwen2.5-VL-7B backbone by 20.1 points. The gains are largest on construction-heavy operations, including Angular Construction (+35.3), Spatial Projection (+25.4), and Connectivity (+20.9), where solving requires introducing missing angles, projections, or links before deduction can proceed. **LatentGeo**$_{SFT}$ already reaches 32.1%, close to much larger or newer baselines, and LaGDPO further improves categories such as Angular Construction and Partitioning. These results support the effectiveness of latent auxiliary construction and the progressive curriculum: by learning continuous latent construction representations instead of relying on pixel-level rendering or external tools, the model internalizes auxiliary constructions for complex spatial relationships.

Public Benchmark Results. On MathVista and MathVerse (Appendix Table 3), LatentGeo improves over Qwen2.5-VL-7B on geometry-related MathVista categories, including GPS (+3.9), GEO (+1.7), and LOG (+2.8), while maintaining comparable MathVerse performance (41.4% vs. 41.0%). For MathVerse, we compare against our same-protocol Qwen2.5-VL-7B reproduction rather than mixing public leaderboard numbers with different reporting settings. We therefore interpret these public-benchmark results as evidence of preserved transfer behavior, not as a broad claim of general-purpose leaderboard superiority. This indicates that construction-oriented training

Table 2: Ablation on GEOAUX by overall and Visual Operation accuracy.

Model Variant	Overall	GEOAUX Visual Operations (Acc %)							
	ALL	Analy.	Angle	Circ.	Conn.	Trans.	Align.	Proj.	Part.
w/o Latent Block (TextOnly)	26.7	28.7	27.1	30.0	25.3	22.1	24.3	30.8	24.3
w/o Curric. Stage 2	13.1	12.8	19.6	6.4	12.8	7.5	12.2	13.7	14.1
w/o Curric. Stage 3	23.1	24.5	28.6	20.9	22.4	22.6	21.5	21.2	20.5
LatentGeo$_{SFT}$	32.1	22.3	34.6	27.3	32.2	22.1	29.3	37.0	23.5
w/ GRPO (Shao et al., 2024)	26.7	20.2	34.6	27.3	27.0	23.9	24.2	24.7	20.9
w/ GDPO (Liu et al., 2026)	26.1	18.1	30.1	23.6	25.5	25.7	24.9	30.1	22.3
LatentGeo$_{LaGDPO}$	**34.6**	**27.7**	**42.1**	**24.6**	**34.9**	**27.0**	**31.9**	**37.0**	**31.5**

does not simply overfit to GEOAUX; instead, it improves geometry-centered reasoning while preserving broader visual-math transfer.

4.3 Ablation Study

Table 2 validates the contribution of the latent construction tokens, curriculum, and RL objective. Replacing the latent construction tokens with text-only supervision drops SFT accuracy from 32.1% to 26.7%, showing that latent construction representations are more effective for auxiliary geometry under the same pipeline. The gains are most visible on operations requiring precise intermediate structures, such as angular construction, connectivity, and projection. Removing curriculum Stage 2 or Stage 3 further reduces accuracy to 13.1% and 23.1%, confirming that both plan-guided latent internalization and inference-matched reasoning are needed to convert auxiliary-diagram supervision into inference-time latent construction.

For RL, standard GRPO and GDPO fall below the SFT-only model (26.7% and 26.1%), whereas **LatentGeo**$_{LaGDPO}$ improves accuracy to 34.6%. This supports our observation that vanilla policy optimization can skip sparse latent boundary tokens during high-temperature rollouts, weakening latent construction token usage. LaGDPO avoids this failure mode by preserving latent construction token generation during rollout while optimizing end-task correctness.

4.4 Mechanism Study

We examine whether latent construction tokens encode auxiliary-construction information rather than acting as simple continuation placeholders. We apply perturbation-based visual attribution (Chen et al., 2025), which segments the input image into superpixels and identifies regions most influential to the prediction through insertion and deletion scoring.

As shown in Fig. 3, Qwen2.5-VL-7B produces

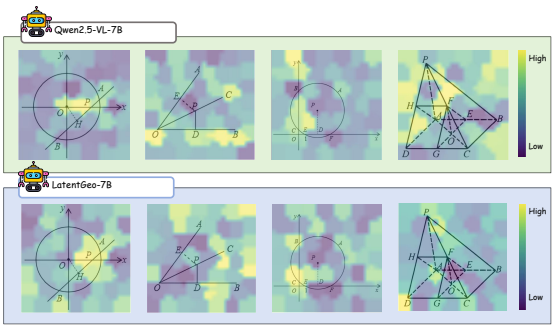


Figure 3: Perturbation-based visual attribution on four geometry problems for Qwen2.5-VL-7B (top, green) and LatentGeo-7B (bottom, blue).

diffuse attributions that often include background regions, indicating weak focus on construction-relevant visual cues. In contrast, LatentGeo concentrates on geometric anchors needed for auxiliary constructions, such as endpoints, intersections, and regions determining auxiliary lines. This suggests that its latent construction tokens carry meaningful geometric semantics beyond simple continuation placeholders.

5 Conclusion

We presented LatentGeo, a framework that models auxiliary geometric constructions with latent construction tokens for multimodal geometric reasoning. This latent formulation reduces representation mismatch and reliance on external tools while supporting end-to-end optimization. Together with a multi-stage training recipe and the GEOAUX benchmark, LatentGeo achieves strong gains on construction-intensive geometry problems and remains competitive on MATHVISTA and MATHVERSE. Ablations and mechanism analyses further show that the curriculum and LaGDPO are important for stable latent internalization, and that latent construction tokens help focus the model on construction-critical geometric primitives.

Limitations

LatentGeo focuses on multimodal geometric reasoning problems in which auxiliary constructions can be represented through latent construction representations. Its current evaluation is centered on geometry and visual-mathematical benchmarks, so the extent to which the learned construction mechanism generalizes beyond mathematical reasoning remains to be tested. The framework also relies on curated auxiliary-construction supervision during training, and scaling this supervision to more diverse domains may require additional annotation or synthetic-data validation. Finally, although LaGDPO improves answer accuracy and stabilizes latent construction token usage, the model can still produce plausible but incorrect constructions or reasoning chains. Such outputs should therefore be treated as assistive rather than authoritative in educational or high-stakes settings.

References

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, and 45 others. 2025a. Qwen3-vl technical report. *Preprint*, arXiv:2511.21631.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025b. Qwen2.5-vl technical report. *Preprint*, arXiv:2502.13923.

Jie Cao and Jing Xiao. 2022. An augmented benchmark dataset for geometric question answering through dual parallel text encoding. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1511–1520. International Committee on Computational Linguistics.

Jiaqi Chen, Tong Li, Jinghui Qin, Pan Lu, Liang Lin, Chongyu Chen, and Xiaodan Liang. 2022. Unigeo: Unifying geometry logical reasoning via reformulating mathematical expression. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3313–3323. Association for Computational Linguistics.

Ruoyu Chen, Xiaoqing Guo, Kangwei Liu, Siyuan Liang, Shiming Liu, Qunli Zhang, Laiyuan Wang, Hua Zhang, and Xiaochun Cao. 2025. Where MLLMs attend and what they rely on: Explaining autoregressive token generation. *arXiv preprint arXiv:2509.22496*.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198.

Jiwan Chung, Junhyeok Kim, Siyeol Kim, Jaeyoung Lee, Min Soo Kim, and Youngjae Yu. 2025. v1: Learning to point visual tokens for multimodal grounded reasoning. *Preprint*, arXiv:2505.18842.

Yushi Hu, Weijia Shi, Xingyu Fu, Dan Roth, Mari Ostendorf, Luke Zettlemoyer, Noah A. Smith, and Ranjay Krishna. 2024. Visual sketchpad: Sketching as a visual chain of thought for multimodal language models. In *Adv. Neural Inform. Process. Syst. (NeurIPS)*.

Mehran Kazemi, Hamidreza Alvari, Ankit Anand, Jialin Wu, Xi Chen, and Radu Soricut. 2023. Geomverse: A systematic evaluation of large models for geometric reasoning. *Preprint*, arXiv:2312.12241.

Sicong Leng, Jing Wang, Jiaxi Li, Hao Zhang, Zhiqiang Hu, Boqiang Zhang, Yuming Jiang, Hang Zhang, Xin Li, Lidong Bing, Deli Zhao, Wei Lu, Yu Rong, Aixin Sun, and Shijian Lu. 2025. Mmr1: Enhancing multimodal reasoning with variance-aware sampling and open resources. *Preprint*, arXiv:2509.21268.

Bangzheng Li, Ximeng Sun, Jiang Liu, Ze Wang, Jialian Wu, Xiaodong Yu, Hao Chen, Emad Barsoum, Muhao Chen, and Zicheng Liu. 2025. Latent visual reasoning. *Preprint*, arXiv:2509.24251.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024. Llava-next: Improved reasoning, ocr, and world knowledge.

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. 2026. Gdpo: Group reward-decoupled normalization policy optimization for multi-reward rl optimization. *Preprint*, arXiv:2601.05242.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2024. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *Int. Conf. Learn. Represent. (ICLR)*.

Pan Lu, Ran Gong, Shibiao Jiang, Liang Qiu, Siyuan Huang, Xiaodan Liang, and Song-Chun Zhu. 2021. Inter-gps: Interpretable geometry problem solving with formal language and symbolic reasoning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6774–6786. Association for Computational Linguistics.

OpenAI. 2024. Gpt-4o mini: advancing cost-efficient intelligence. OpenAI product announcement.

OpenAI. 2024. GPT-4o system card.

Yiming Qin, Bomin Wei, Jiaxin Ge, Konstantinos Kallidromitis, Stephanie Fu, Trevor Darrell, and Xudong Wang. 2025. Chain-of-visual-thought: Teaching vlms to see and think better with continuous visual tokens. *Preprint*, arXiv:2511.19418.

Minjoon Seo, Hannaneh Hajishirzi, Ali Farhadi, Oren Etzioni, and Clint Malcolm. 2015. Solving geometry problems: Combining text and diagram interpretation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1466–1476. Association for Computational Linguistics.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *Preprint*, arXiv:2402.03300.

Weikang Shi, Aldrich Yu, Rongyao Fang, Houxing Ren, Ke Wang, Aojun Zhou, Changyao Tian, Xinyu Fu, Yuxuan Hu, Zimu Lu, Linjiang Huang, Si Liu, Rui Liu, and Hongsheng Li. 2025. Mathcanvas: Intrinsic visual chain-of-thought for multimodal mathematical reasoning. *Preprint*, arXiv:2510.14958.

Guohao Sun, Hang Hua, Jian Wang, Jiebo Luo, Sohail Dianat, Majid Rabbani, Raghuveer Rao, and Zhiqiang Tao. 2025. Latent chain-of-thought for visual reasoning. *Preprint*, arXiv:2510.23925.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hongsheng Li. 2024. Measuring multimodal mathematical reasoning with MATH-Vision. In *Adv. Neural Inform. Process. Syst. (NeurIPS)*. Datasets and Benchmarks Track.

Ke Wang, Junting Pan, Linda Wei, Aojun Zhou, Weikang Shi, Zimu Lu, Han Xiao, Yunqiao Yang, Houxing Ren, Mingjie Zhan, and Hongsheng Li. 2025a. Mathcoder-vl: Bridging vision and code for enhanced multimodal mathematical reasoning. In *Findings of the Association for Computational Linguistics (ACL)*.

Peijie Wang, Zhong-Zhi Li, Fei Yin, Xin Yang, Dekang Ran, and Cheng-Lin Liu. 2025b. MV-MATH: Evaluating multimodal math reasoning in multi-visual contexts. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*.

Peijie Wang, Chao Yang, Zhong-Zhi Li, Fei Yin, Dekang Ran, Mi Tian, Zhilong Ji, Jinfeng Bai, and Cheng-Lin Liu. 2025c. SOLIDGEO: Measuring multimodal spatial math reasoning in solid geometry. *Preprint*, arXiv:2505.21177.

Qixun Wang, Yang Shi, Yifei Wang, Yuanxing Zhang, Pengfei Wan, Kun Gai, Xianghua Ying, and Yisen Wang. 2025d. Monet: Reasoning in latent visual space beyond images and language. *Preprint*, arXiv:2511.21395.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, Zhaokai Wang, Zhe Chen, Hongjie Zhang, Ganlin Yang, Haomin Wang, Qi Wei, Jinhui Yin, Wenhao Li, Erfei Cui, and 56 others. 2025e. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *Preprint*, arXiv:2508.18265.

Yikun Wang, Yibin Wang, Dianyi Wang, Zimian Peng, Qipeng Guo, Dacheng Tao, and Jiaqi Wang. 2026. Geometryzero: Advancing geometry solving via group contrastive policy optimization. *Preprint*, arXiv:2506.07160.

Jingxuan Wei, Caijun Jia, Qi Chen, Honghao He, Linzhuang Sun, Conghui He, Lijun Wu, Bihui Yu, and Cheng Tan. 2025. Geoint-r1: Formalizing multimodal geometric reasoning with dynamic auxiliary constructions. *arXiv preprint arXiv:2508.03173*.

Wenjun Wu, Lingling Zhang, Jun Liu, Xi Tang, Yaxian Wang, Shaowei Wang, and Qianying Wang. 2024. E-GPS: Explainable geometry problem solving via top-down solver and bottom-up generator. In *IEEE Conf. Comput. Vis. Pattern Recog. (CVPR)*.

Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng Yin, Fengyun Rao, Minfeng Zhu, Bo Zhang, and Wei Chen. 2025a. R1-onevision: Advancing generalized multimodal reasoning through cross-modal formalization. *Preprint*, arXiv:2503.10615.

Zeyuan Yang, Xueyang Yu, Delin Chen, Maohao Shen, and Chuang Gan. 2025b. Machine mental imagery: Empower multimodal reasoning with latent visual tokens. *Preprint*, arXiv:2506.17218.

Huanyu Zhang, Wenshan Wu, Chengzu Li, Ning Shang, Yan Xia, Yangyu Huang, Yifan Zhang, Li Dong, Zhang Zhang, Liang Wang, Tieniu Tan, and Furu Wei. 2025a. Latent sketchpad: Sketching visual thoughts to elicit multimodal reasoning in mllms. *Preprint*, arXiv:2510.24514.

Kaichen Zhang, Keming Wu, Zuhao Yang, Bo Li, Kairui Hu, Bin Wang, Ziwei Liu, Xingxuan Li, and Lidong Bing. 2025b. Openmmreasoner: Pushing the frontiers for multimodal reasoning with an open and general recipe. *Preprint*, arXiv:2511.16334.

Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou, Pan Lu, Kai-Wei Chang, Peng Gao, and Hongsheng Li. 2024. Mathverse: Does your multi-modal llm truly see the diagrams in visual math problems? In *Eur. Conf. Comput. Vis. (ECCV)*.

Renrui Zhang, Xinyu Wei, Dongzhi Jiang, Ziyu Guo, Shicheng Li, Yichi Zhang, Chengzhuo Tong, Jiaming Liu, Aojun Zhou, Bin Wei, Shanghang Zhang, Peng Gao, Chunyuan Li, and Hongsheng Li. 2025c. MAVIS: Mathematical visual instruction tuning with an automatic data engine. In *Int. Conf. Learn. Represent. (ICLR)*.

Shan Zhang, Aotian Chen, Yanpeng Sun, Jindong Gu, Yi-Yu Zheng, Piotr Koniusz, Kai Zou, Anton van den Hengel, and Yuan Xue. 2025d. Open eyes, then reason: Fine-grained visual mathematical understanding in multimodal large language models. *arXiv preprint arXiv:2501.06430*.

A Appendix

A.1 Additional Benchmark Results

Table 3: **Evaluation on two public multimodal math benchmarks beyond GEOAUX.** We report geometry-related MATHVISTA categories (GPS/GEO/LOG/TQA) and visual-dependence MATHVERSE splits (ALL_4/TD/VI/VD/VO/V-Avg). MATHVISTA values are reported testmini results except LatentGeo, which is evaluated by us; MATHVERSE follows our four-category protocol, and Qwen2.5-VL-7B is our same-protocol reproduction.

(a) MATHVISTA

Model	GPS	GEO	LOG	TQA
References / closed-source MLLMs				
Human	–	–	40.5	63.2
GPT-4V	50.5	51.0	21.6	65.2
Qwen-VL-Plus	38.5	39.3	19.8	55.1
Gemini-1.0-Pro	40.4	41.0	23.4	61.4
Open-source MLLMs				
LLaVA-1.5-13B	22.7	22.8	15.3	40.5
SPHINX-V2-13B	16.4	17.6	20.7	41.8
OmniLMM-12B	17.8	20.9	21.6	44.9
Math-LLaVA-13B	57.7	56.5	24.3	51.3
Qwen2-VL-7B	41.8	43.1	24.3	60.1
Qwen2.5-VL-7B	66.8	66.9	26.9	66.7
LLaVA-1.5-DP	56.2	46.0	20.7	45.6
Qwen2-VL-DP	56.4	47.2	26.1	64.6
Qwen2.5-VL-DP	72.6	69.3	27.0	68.5
LatentGeo	70.7	68.6	**29.7**	66.5

(b) MATHVERSE

Model	ALL_4	TD	VI	VD	VO	V-Avg
References / closed-source MLLMs						
Random	12.4	12.4	12.4	12.4	12.4	12.4
Human	66.9	71.2	61.4	68.3	66.7	65.5
GPT-4o	50.5	59.8	48.0	46.5	47.6	47.4
Gemini-1.5-Pro	35.5	39.8	32.0	36.8	33.3	34.0
Open-source MLLMs						
Qwen-VL-Plus	21.4	26.0	18.5	19.1	21.8	19.8
InternVL2-8B	32.5	39.0	32.2	30.9	27.7	30.3
InternVL2.5-8B	37.8	43.0	43.0	42.2	22.8	36.0
LLaVA-NeXT-34B	33.9	49.0	35.2	28.9	22.4	28.8
Math-LLaVA-13B	22.4	27.3	24.5	21.7	16.1	20.8
Math-PUMA-Qwen2-7B	33.3	42.1	33.4	31.6	26.0	30.3
MAVIS-7B	34.7	43.2	34.1	29.7	31.8	31.9
InfiMM-Math	33.3	46.7	38.1	32.4	15.8	28.8
Qwen2.5-VL-7B	41.0	47.7	39.6	38.7	38.1	38.8
LatentGeo	41.4	**51.4**	39.9	37.7	37.3	38.3

A.2 Detailed Statistics and Split Information

We report detailed statistics for the evaluation benchmarks used in this work, including MATHVISTA, MATHVERSE, and GEOAUX.

MATHVISTA. We evaluate on the MATHVISTA *testmini* split and report the geometry-related categories used in our main table: **GPS**, **GEO**, **LOG**, and **TQA**.

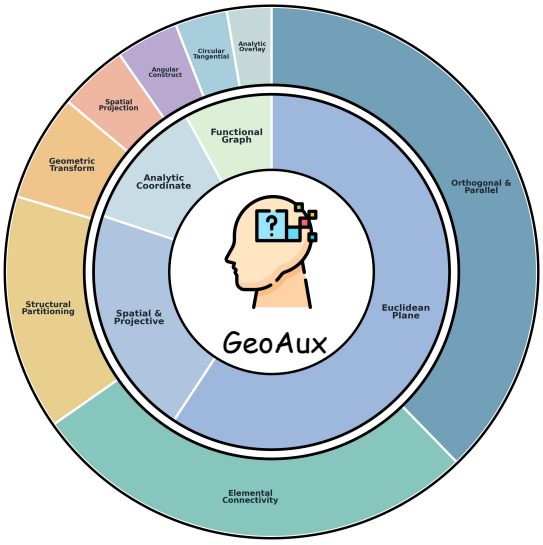
MATHVERSE. We evaluate on the MATHVERSE *testmini* split, which comprises four categories: **TD**, **VI**, **VD**, and **VO**. Each category contains 788 test instances, resulting in a total of 3,152 test samples.

GEOAUX. This section expands the construction protocol summarized in Sec. 3.4. GEOAUX is designed for visually dependent geometry problems whose solutions require introducing missing intermediate structures rather than directly reading values from the original diagram.

The benchmark is built through a proposal–validation pipeline. We first collect candidate geometry problems with diagrams and use frontier LLMs to propose auxiliary-construction requirements and solution rationales, including missing connections, perpendicular or parallel alignments, spatial projections, structural partitions, and analytic overlays. These proposals are treated as candidates rather than accepted labels. Human annotators then verify each item for four criteria: valid geometric relations, unique and checkable answer, clear diagram, and necessity of the proposed construction. Samples failing any criterion are revised or discarded.

To reduce leakage from the training data, we keep GEOAUX disjoint from all SFT and RL sources. Candidate problems are normalized and screened against the training pool using text-similarity filtering, and diagrams are additionally checked for visual near duplicates. Each retained instance is labeled along two axes: *Knowledge Domain*, which indicates the mathematical field, and *Visual Operation*, which indicates the construction skill required. Visual Operation labels are non-exclusive because a single problem may require multiple auxiliary operations.

The detailed distribution of the GEOAUX bench-



Statistic	Number
Total instances	**2,228**
Question type	
– Free-form	1,313
– Multiple-choice	915
Knowledge domain	
– Analytic coordinate geometry	265
– Euclidean plane geometry	1,320
– Functional graph geometry	178
– Spatial & projective geometry	465
Visual operation	
– Analytic overlay	94
– Angular construction	133
– Circular & tangential augmentation	110
– Elemental connectivity	957
– Geometric transformation	226
– Orthogonal & parallel alignment	1,311
– Spatial projection & unfolding	146
– Structural partitioning	498

Figure 4: **Distribution and taxonomy of the GEOAUX benchmark.** Left: taxonomy counts. Right: pie chart. Visual-operation labels are non-exclusive, so their counts may exceed the total number of instances.

mark is summarized in Figure 4. The dataset comprises a total of 2,228 unique instances, with a heavy emphasis on Euclidean Plane Geometry and Alignment-based operations to reflect real-school geometry challenges.

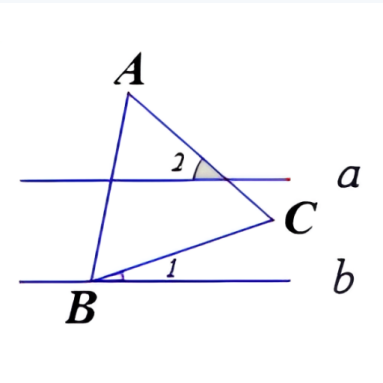
Representative Benchmark Example. We present four benchmark exemplars from GEOAUX across the main knowledge categories.

Benchmark Example 1: Euclidean Plane Geometry

Knowledge. Euclidean Plane Geometry.
Visual Operation. Elemental Connectivity.

Question. Hint: Please solve the problem and provide the final boxed answer as the uppercase option letter only, e.g., $\boxed{A}$.
In the figure, lines $a \parallel b$. In the equilateral triangle $\triangle ABC$, vertex B lies on line b, and $\angle 1 = 20^\circ$. Find the measure of $\angle 2$.
Options: A. 60° B. 45° C. 40° D. 30°

Question Image.



Construction Requirement. Extend AC to meet line b and convert the target angle into a solvable relation under the parallel-line constraint.

Correct Answer. $\boxed{C}$ (40°).

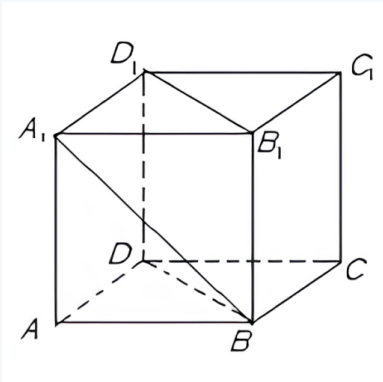
Key Evaluation Focus. This sample tests whether the model can introduce the missing auxiliary connection needed for angle reasoning, rather than relying on direct visual reading alone.

Benchmark Example 2: Spatial & Projective Geometry

Knowledge. Spatial & Projective Geometry.
Visual Operation. Orthogonal & Parallel Alignment; Elemental Connectivity; Structural Partitioning.

Question. Hint: Please solve the problem and provide the final boxed answer as an exact angle measure, keeping degree notation if needed, e.g., $\boxed{50^\circ}$.
In the cube $ABCD{-}A_1B_1C_1D_1$ with edge length a, find the angle between the line A_1B and the plane D_1B_1BD.

Question Image.



Construction Requirement. Introduce the top-face center via diagonal intersection and identify the foot of the perpendicular from A_1 to plane D_1B_1BD.

Correct Answer. $\boxed{30^\circ}$.

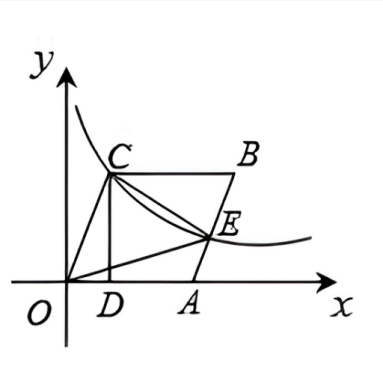
Key Evaluation Focus. This case probes whether the model can decompose a 3D angle-to-plane problem into an auxiliary planar configuration through orthogonality and structural partitioning.

Benchmark Example 3: Analytic Coordinate Geometry

Knowledge. Analytic Coordinate Geometry.
Visual Operation. Elemental Connectivity.

Question. Hint: Please solve the problem and provide the final boxed answer as the uppercase option letter only, e.g., $\boxed{A}$.
In the figure, quadrilateral $OABC$ is a rhombus. Line CD is perpendicular to the x-axis with foot D. The graph of $y = \frac{4}{x}$ passes through C and intersects AB at E. Given $OD = 2$, what is the area of $\triangle OCE$?
Options: A. 2 B. 4 C. $2\sqrt{2}$ D. $4\sqrt{2}$

Question Image.



Construction Requirement. Connect AC and combine the conic constraint with the rhombus side-equality relation to recover the geometry needed for the area computation.

Correct Answer. $\boxed{C}$ $(2\sqrt{2})$.

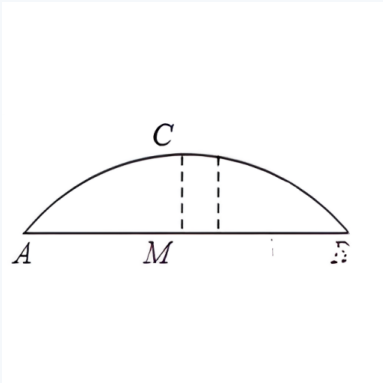
Key Evaluation Focus. This sample tests whether the model can bridge symbolic curve information and geometric structure by adding a non-explicit segment that stabilizes the area reasoning.

Benchmark Example 4: Functional Graph Geometry

Knowledge. Functional Graph Geometry.
Visual Operation. Analytic Overlay.

Question. Hint: Please solve the problem and provide the final boxed answer as the uppercase option letter only, e.g., $\boxed{A}$.
As shown, a bridge arch is modeled by a parabola. The maximum height is 16 meters, and the span is 40 meters. On segment AB, at a point 5 meters from the center M, what is the height of the bridge (in meters)?
Options: A. 14 B. 15 C. 13 D. 12

Question Image.



Construction Requirement. Overlay an analytic coordinate system by taking the bridge span as the x-axis and the midpoint as the origin, then recover the parabola from its vertex and intercepts.

Correct Answer. $\boxed{B}$ (15).

Key Evaluation Focus. This sample examines whether the model can translate a real-scene functional graph into an analytic form before performing numeric evaluation.

A.3 Hyperparameters

SFT Hyperparameters. We fine-tune the model using a multi-stage supervised fine-tuning (SFT) strategy. The learning rate follows a cosine decay schedule, and mixed precision with `bfloat16` is employed to optimize training efficiency. Table 4 summarizes the SFT settings. We use the same number of latent construction tokens, $K = 10$, across SFT and RL stages to keep the latent construction capacity fixed when evaluating the effects of the curriculum and LaGDPO.

Table 4: SFT hyperparameters.

Hyperparameter	Value
Optimizer	AdamW
Learning rate	1×10^{-5}
Weight decay	0.01
Max sequence length	1,024
Number of latent construction tokens K	10
SFT Stage 1 epochs	5
SFT Stage 2 epochs	2
SFT Stage 3 epochs	5
SFT Stage 2 alignment weight	2.0
SFT Stage 3 alignment weight	0 (removed)
Training precision	`bfloat16`

RL Hyperparameters. We fine-tune the model with LaGDPO, a KL-regularized PPO-style update with group-decoupled advantages. Table 5 lists the main RL hyperparameters.

Table 5: RL hyperparameters.

Hyperparameter	Value
KL coefficient β	0.03
PPO epochs	1
Clip ratio (lower) ϵ_l	0.2
Clip ratio (upper) ϵ_h	0.3
Dual clip c	3.0
Optimizer	AdamW
Learning rate	1×10^{-7}
Warmup ratio	0.15
Max gradient norm	2.0
Update global batch size	16
Samples per prompt N	8
Max prompt length	4096
Max response length	2048
Sampling temperature	0.9
Sampling top-p	0.99
Length reward max	0.2
Length target threshold ℓ_t	$0.9L_{\max}$
Repetition penalty weight	1.2
Repetition max penalty	2
3-gram repetition threshold τ_3	0.18
4-gram repetition threshold τ_4	0.12
Consecutive-token run threshold m_0	6
Number of latent construction tokens K	10
Latent logit bias	10
Latent logit bias decay	1.0
Latent reward EMA	0.9

A.4 Prompting Scheme

We adopt one unified prompt for SFT/RL/inference and a two-stage evaluation protocol (answer extraction, then answer judging). We report the three prompt templates below.

Prompt: Geometry Reasoning Instruction (SFT & RL)

[Image]
You are a helpful assistant.
First, analyze the problem with the image and describe all necessary geometric constructions or spatial operations using concise and standard language.
Then, provide a reasoning to solve the problem.
Finally, give the final answer. The final answer must be boxed, e.g., \boxed{answer}.

Question:
{question}

Figure 5: Unified instruction prompt used for SFT, RL, and inference.

Prompt: Answer Extraction for Evaluation

{DEMO_PROMPT}
(Contains extraction guidelines and in-context examples.)

{hint}
Hint: This is a Multiple Choice Question. Extract only the option letter (A, B, C, D); if the model outputs option content, map it back to the corresponding letter when possible.
Hint: This is a Free-form Question. Extract the final value or expression.

Question: {query}
Model response: {response}
Extracted answer:

Figure 6: Answer-extraction template used before final matching.

Prompt: Equivalence-Based Answer Judging

You are a math teacher grading a student's answer.
Question: {question}
Correct Answer: {ground_truth}
Student Answer: {prediction}

Does the student answer match the correct answer?
For numerical answers, they should be equivalent (e.g., 1.5 equals 3/2).
For multiple choice, the option letter should match.
For expressions, they should be algebraically equivalent.

Reply only with "True" or "False".

Figure 7: Judge prompt used for final correctness decision.

A.5 Training Data Sources

SFT Training Data. We construct a high-quality Supervised Fine-Tuning (SFT) dataset comprising approximately 30,000 geometry problems. These samples are aggregated from multiple public geometric reasoning benchmarks, including MATHCANVAS, GEOMETRY3K, GEOQA+, GEOS, and UNIGEO. To ensure the data quality and alignment with our training objective, we employ a rigorous data processing pipeline involving both manual curation and LLM-assisted reformatting. Specifically, raw data instances are first manually filtered to exclude low-quality images or ambiguous problem statements. Subsequently, we utilize advanced LLMs (e.g., GPT-4o) to canonicalize the solution traces into the unified format required for SFT (as detailed in Appendix A.4), ensuring a consistent structure of visual description, reasoning steps, and the final boxed answer.

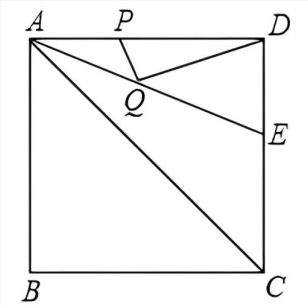
Data Composition. The SFT dataset integrates diverse geometric challenges from the following sources:

- **MATHCANVAS** (Shi et al., 2025): A multimodal dataset emphasizing intrinsic visual reasoning. We select instances that require detailed visual understanding to complement text-based logic.
- **GEOMETRY3K** (Lu et al., 2021): A comprehensive benchmark of high-school geometry problems. We utilize its rich annotations to construct step-by-step reasoning chains.
- **GEOQA+** (Cao and Xiao, 2022): A large-scale geometric question-answering dataset. We filter for problems that involve complex calculation and theorem application.
- **GEOS** (Seo et al., 2015): One of the pioneering datasets for geometry problem solving, providing fundamental plane geometry instances.
- **UNIGEO** (Chen et al., 2022): A unified geometric reasoning dataset covering both calculation and proof problems across various difficulty levels.

Representative SFT Training Example

Question. In square $ABCD$ with side length 2, the angle bisector of $\angle DAC$ intersects DC at E. Points P and Q move on AD and AE, respectively. Find the minimum value of $DQ + PQ$.

Question Image.

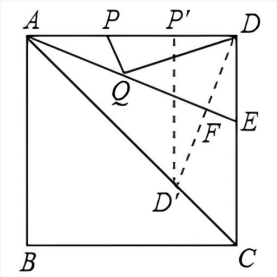


Assistant Output. Reflect D across line AE to point D'. Since Q lies on AE, convert $DQ + PQ$ into the broken path $D'Q + QP$, then reduce the minimization to the distance from D' to line AD. Let P' be the foot of the perpendicular from D' to AD.

<|latent_start|><|latent_pad|>. . . <|latent_pad|><|latent_end|>

Because Q lies on the mirror line AE, reflection gives $DQ = D'Q$. Hence $DQ + PQ = D'Q + QP$. For any point $P \in AD$, the broken path from D' to P through Q is at least the straight-line distance $D'P$, with equality when D', Q, P are collinear. Therefore, the problem reduces to minimizing $D'P$ for P on line segment AD. This minimum is attained when $P = P'$, the foot of the perpendicular from D' to AD, so $\min(DQ + PQ) = D'P'$. Since AE bisects $\angle DAC$, reflecting ray AD across AE maps it to ray AC. Therefore, D' lies on AC and $AD' = AD = 2$. In a square, diagonal AC makes a $45°$ angle with side AD, so $\angle D'AP' = 45°$. Thus $\triangle AD'P'$ is a $45°$-$45°$-$90°$ right triangle with hypotenuse $AD' = 2$, giving $D'P' = \frac{2}{\sqrt{2}} = \sqrt{2}$. Therefore, the minimum value is $\boxed{\sqrt{2}}$.

Auxiliary Image.



RL Training Data. Our RL training data are sampled from three public geometry problem-solving datasets: Geometry3K, GeomVerse, and MathCanvas (Lu et al., 2021; Kazemi et al., 2023; Shi et al., 2025). In total, we collect 3,339 problems, including 1,362 from Geometry3K, 1,309 from GeomVerse, and 668 from MathCanvas. For RL, we use each problem (with its associated diagram/image when provided) as the prompt, and treat the dataset-provided final answer as outcome supervision for reward computation. Importantly, we do not consume intermediate solution traces (e.g., chain-of-thought or step annotations) as supervision in the RL stage.

Data Composition. The RL dataset draws from the following sources:

- **GEOMETRY3K** (Lu et al., 2021): A benchmark of 3,002 high-school geometry multiple-choice problems. We randomly sample 1,362 instances and use the ground-truth answer option as the outcome-based reward signal.
- **GEOMVERSE** (Kazemi et al., 2023): A procedurally generated geometry dataset designed to probe reasoning capabilities under controllable factors such as reasoning depth. We sample 1,309 problems and use only their final answers for outcome supervision.
- **MATHCANVAS** (Shi et al., 2025): A multimodal mathematical reasoning resource emphasizing intrinsic visual reasoning and diagram-centric problem solving. We sample 668 geometry instances and use only the dataset-provided final answers as RL reward supervision.

A.6 Latent-Aware Decoding Bias Analysis

As discussed in Section 3.3, high-temperature rollout sampling during early RL training can cause

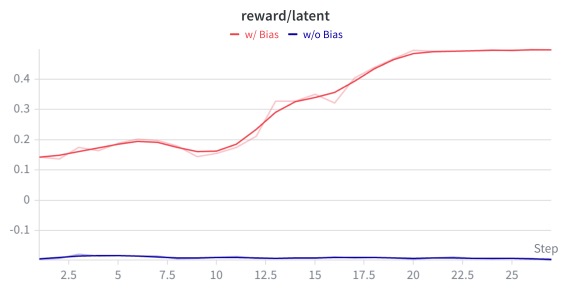


Figure 8: **Latent reward trajectories during RL training.** Red denotes training with the adaptive decoding bias; blue denotes training without it.

the model to skip the sparse latent boundary tokens entirely. Once the model fails to produce valid latent construction token spans in the majority of rollouts, the resulting latent reward remains at the penalty value, and the policy receives no positive gradient signal to recover latent construction token usage. This leads to an irreversible collapse where the model degrades into pure textual reasoning.

To empirically validate this failure mode and the effectiveness of our proposed latent-aware decoding bias (Eq. 10), we compare RL training with and without the adaptive logit bias applied to the latent boundary tokens. As shown in Fig. 8, applying the bias (red) makes the latent reward increase steadily from the early training steps and saturate near the maximum, indicating that the policy reliably learns to produce well-formed latent construction token spans. Without the bias (blue), the latent reward remains near the penalty floor throughout training, confirming that early failure to emit latent boundary tokens under high-temperature sampling leads to collapse of latent construction token usage.

A.7 Baselines

Table 3 summarizes evaluation on two public multimodal math benchmarks, MATHVISTA (Lu et al., 2024) and MATHVERSE (Zhang et al., 2024). The MATHVISTA panel focuses on geometry-related categories and includes representative reference systems, open-source MLLMs, Qwen2.5-VL-7B variants, and LatentGeo. The MATHVERSE panel focuses on visual-dependence splits and includes Random and Human references, representative closed-source MLLMs, open-source general MLLMs, math-specialized MLLMs, Qwen2.5-VL-7B, and LatentGeo. Since these public benchmarks use different reporting protocols, we emphasize matched-category comparisons against the Qwen2.5-VL-7B backbone rather than broad leaderboard claims.

We evaluate **LatentGeo** on the proposed GEOAUX benchmark using two groups of baselines, as summarized in Table 1. For *Closed-Source MLLMs*, we include GPT-4o (OpenAI, 2024), Qwen3-VL-Flash (Bai et al., 2025a), and GPT-4o-mini (OpenAI, 2024). For *Open-Source General MLLMs*, we benchmark LLaVA-1.6 (Vicuna-7B) (Liu et al., 2024), MMR1-7B-RL (Leng et al., 2025), OpenMMReasoner-RL (Zhang et al., 2025b), R1-Onevision-7B (Yang et al., 2025a), Qwen2.5-VL-7B and Qwen2.5-VL-32B (Bai et al., 2025b), Qwen3-VL-8B (Bai et al., 2025a), InternVL2-8B (Chen et al., 2024), and InternVL3.5-8B (Wang et al., 2025e). All models are evaluated under the same GEOAUX protocol and report accuracy (%) over 2,228 instances.

A.8 Case Study

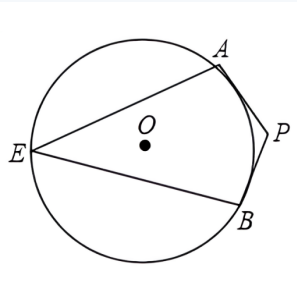
We present several qualitative inference cases from GEOAUX to illustrate how **LatentGeo** introduces latent auxiliary constructions and completes the downstream geometric reasoning chain.

Case Study: Euclidean Angle Reasoning

Knowledge. Euclidean Plane Geometry.
Visual Operation. Elemental Connectivity.

Question. Hint: Please solve the problem and provide the final boxed answer as an exact angle measure, keeping degree notation if needed, e.g., $\boxed{50^\circ}$.
In the figure, PA and PB are tangents to circle O at points A and B, respectively. Point E lies on circle O. If $\angle AEB = 40^\circ$, find $\angle APB$.

Question Image.



Model Response. Join OA and OB. Since PA and PB are tangents to circle O, we have $OA \perp PA$ and $OB \perp PB$, so $\angle OAP = \angle OBP = 90^\circ$.

<|latent_start|><|latent_pad|>. . . <|latent_pad|><|latent_end|>

The inscribed angle $\angle AEB$ subtends arc AB, so the central angle $\angle AOB$ subtending the same arc is twice the inscribed angle: $\angle AOB = 2\angle AEB = 2 \times 40^\circ = 80^\circ$. In quadrilateral $AOPB$, the angles at A and B are right angles, and the angle at O is 80°, so the remaining

angle at P is $\angle APB = 360^\circ - 90^\circ - 90^\circ - 80^\circ = 100^\circ$. Therefore, the answer is $\boxed{100}$.

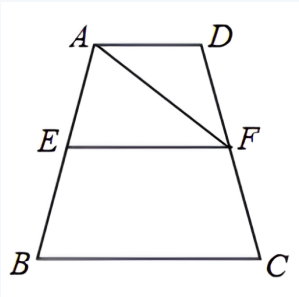
Correct Answer. $\boxed{100^\circ}$.

Case Study: Euclidean Area Reasoning

Knowledge. Euclidean Plane Geometry.
Visual Operation. Orthogonal & Parallel Alignment; Elemental Connectivity.

Question. Hint: Please solve the problem and provide the final boxed answer as an exact value only, e.g., $\boxed{24}$.
In trapezoid $ABCD$, EF is the midline. Segment AF is drawn. If the area of $\triangle AEF$ is $6\,\text{cm}^2$, find the area of trapezoid $ABCD$ (in cm^2).

Question Image.



Model Response. Draw $AG \perp BC$ with G on BC. Since EF is the midline of trapezoid $ABCD$, the distance between the parallel sides AB and CD is AG, and the length of the midline is $EF = \frac{1}{2}(AB+CD)$. Therefore, the area of the trapezoid is $S_{ABCD} = \frac{1}{2}(AB + CD) \cdot AG = EF \cdot AG$.

<|latent_start|><|latent_pad|>. . . <|latent_pad|><|latent_end|>

Because $EF \parallel AB$, the triangles $\triangle AEF$ and $\triangle ABG$ are similar with similarity ratio $\frac{EF}{AB} = \frac{1}{2}$. Hence the area ratio is $\left(\frac{EF}{AB}\right)^2 = \frac{1}{4}$. Given $S_{\triangle AEF} = 6$, we have $S_{\triangle ABG} = 4 \times 6 = 24$. Since $S_{ABCD} = S_{\triangle ABG}$ (the area of the trapezoid equals the area of the larger similar triangle with the same height AG), the area of trapezoid $ABCD$ is 24. Therefore, the answer is $\boxed{24}$.

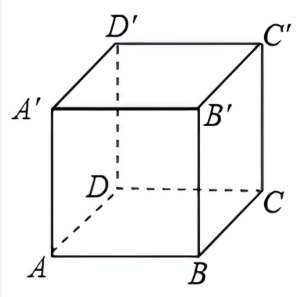
Correct Answer. $\boxed{24}$.

Case Study: Spatial Surface Path

Knowledge. Spatial & Projective Geometry.
Visual Operation. Spatial Projection & Unfolding; Elemental Connectivity.

Question. Hint: Please solve the problem and provide the final boxed answer as a simplified expression only, e.g., $\boxed{40\sqrt{5}}$.
A cube gift box has side length 20 cm. A gecko is at vertex A on the bottom face, and a mosquito is at vertex C' on the top face, as shown. The gecko must reach the mosquito within 0.5 minutes, moving along the surfaces of the box. What is the minimum speed the gecko must crawl per minute (in cm/min)? Express your answer in simplest radical form.

Question Image.



Model Response. Consider the net of the cube that unfolds the two faces containing the gecko and the mosquito so that the shortest path on the surfaces becomes a straight line in the net. In the net, the gecko starts at A on the bottom face and the mosquito is at C' on the top face. The straight-line distance between A and C' in the net is the hypotenuse of a right triangle with legs 20 and 40 (since the bottom and top faces are separated by 20 cm and the horizontal distance between the corresponding positions on the two faces is 40 cm).

<|latent_start|><|latent_pad|>. . . <|latent_pad|><|latent_end|>

Thus the shortest path length is $\sqrt{20^2 + 40^2} = \sqrt{400 + 1600} = \sqrt{2000} = 20\sqrt{5}$ cm. The gecko has 0.5 minutes to cover this distance, so the required minimum speed is $\dfrac{20\sqrt{5}}{0.5} = 40\sqrt{5}$ cm/min. Therefore, the answer is $\boxed{40\sqrt{5}}$.

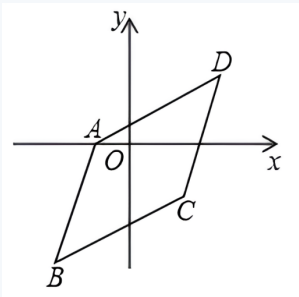
Correct Answer. $\boxed{40\sqrt{5}}$.

Case Study: Analytic Coordinate Reasoning

Knowledge. Analytic Coordinate Geometry.
Visual Operation. Analytic Overlay.

Question. Hint: Please solve the problem and provide the final boxed answer as the uppercase option letter only, e.g., $\boxed{A}$.
In parallelogram $ABCD$, three vertices are $A(-1, 0)$, $B(-2, -3)$, and $C(2, -1)$. What are the coordinates of D?
Options: A. $(3, 1)$ B. $(3, 2)$ C. $(3, 3)$ D. $(3, 4)$

Question Image.



Model Response. Since $ABCD$ is a parallelogram, the diagonals AC and BD bisect each other. Therefore, the midpoint of AC equals the midpoint of BD.

<|latent_start|><|latent_pad|>. . . <|latent_pad|><|latent_end|>

Compute the midpoint of AC: the midpoint formula gives $\left(\frac{-1+2}{2}, \frac{0+(-1)}{2}\right) = \left(\frac{1}{2}, -\frac{1}{2}\right)$. Let $D = (x, y)$. Then the midpoint of BD is $\left(\frac{-2+x}{2}, \frac{-3+y}{2}\right)$. Setting the two midpoints equal gives the system $\dfrac{-2+x}{2} = \dfrac{1}{2}$ and $\dfrac{-3+y}{2} = -\dfrac{1}{2}$. Solve the first: $-2 + x = 1 \Rightarrow x = 3$. Solve the second: $-3 + y = -1 \Rightarrow y = 2$. Hence $D = (3, 2)$. Therefore, the correct option is $\boxed{B}$.

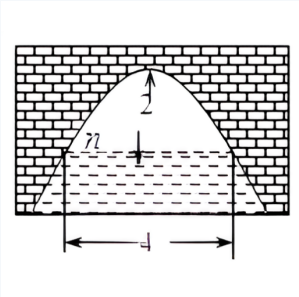
Correct Answer. $\boxed{B}$.

Case Study: Functional Graph Reasoning

Knowledge. Functional Graph Geometry.
Visual Operation. Analytic Overlay.

Question. Hint: Please solve the problem and provide the final boxed answer as the uppercase option letter only, e.g., $\boxed{A}$.
The arch of a bridge is parabolic. When the arch apex is 2 m above the water surface, the water surface width is 4 m. If the water surface drops by 1 m, by how much does the water surface width increase?
Options: A. $1\,m$ B. $2\,m$ C. $(2\sqrt{6} - 4)\,m$ D. $(\sqrt{6} - 2)\,m$

Question Image.



Model Response. Set up a coordinate system with the vertex of the parabola at the origin and the axis of symmetry along the y-axis. Then the parabola is $y = ax^2$ with $a < 0$. The water surface at the initial level is the line $y = -2$, and the water surface at the final level is the line $y = -3$.

<|latent_start|><|latent_pad|>. . . <|latent_pad|><|latent_end|>

At the initial level $y = -2$, the water surface is the line $y = -2$. The intersection with the parabola is at $y = -2$, so $-2 = ax^2$, giving $x = \pm\sqrt{-\frac{2}{a}}$. The width is $2\sqrt{-\frac{2}{a}}$. At the final level $y = -3$, the intersection is at $y = -3$, so $-3 = ax^2$, giving $x = \pm\sqrt{-\frac{3}{a}}$. The new width is $2\sqrt{-\frac{3}{a}}$. The increase in width is $2\sqrt{-\dfrac{3}{a}} - 2\sqrt{-\dfrac{2}{a}} = 2\sqrt{-\dfrac{1}{a}}\left(\sqrt{3} - \sqrt{2}\right) = 2\sqrt{6} - 4$. Therefore, the correct option is $\boxed{C}$.

Correct Answer. $\boxed{C}$.