Multimodal Graph-of-Thoughts: Hypothesis-Verification Graphs for Multimodal Reasoning in Vision-Language Models

Irina Belyaeva
University of Maryland, Baltimore County, USA.
irinbel1@umbc.edu

Abstract

Graph-of-Thought (GoT) frameworks extend chain-of-thought reasoning by exploring structured hypothesis graphs with refinement and aggregation. However, existing approaches rank candidate hypotheses primarily by textual plausibility, making them brittle in multimodal settings where linguistically coherent reasoning can contradict perceptual evidence, especially under visual ambiguity, contradictory cues, or compositional inference. We introduce Multimodal Graph-of-Thoughts (MM-GoT), a training-free, verification-constrained inference framework that integrates cross-modal grounding directly into hypothesis graph search. Rather than scoring branches by generative likelihood alone, MM-GoT augments each reasoning node with three modality-specific verification signals—semantic consistency, spatial validity, and attentional grounding—enabling evidence-based pruning, revision, and synthesis of candidate reasoning paths during inference. This allows the model to suppress perceptually unsupported branches, preserve competing interpretations under ambiguity, and reconcile mutually compatible trajectories before committing to a final answer. We evaluate MM-GoT on four benchmarks spanning complementary aspects of grounded multimodal reasoning across four open-weight MLLM backbones. Under matched inference budgets, MM-GoT consistently outperforms Chain-of-Thought (CoT), multimodal CoT (MM-CoT), Tree-of-Thoughts (ToT), and Graph-of-Thoughts (GoT) baselines, improving mean accuracy by 3.1 percentage points over GoT on average across four benchmarks and four backbones, while reducing tokens per query by 22–24%. Gains are largest under high visual ambiguity, where MM-GoT improves by up to 6.9 percentage points, consistent with verification suppressing linguistically plausible but perceptually invalid branches. Ablations confirm that both cross-modal verification and graph-structured synthesis contribute independently to improved grounded reasoning.

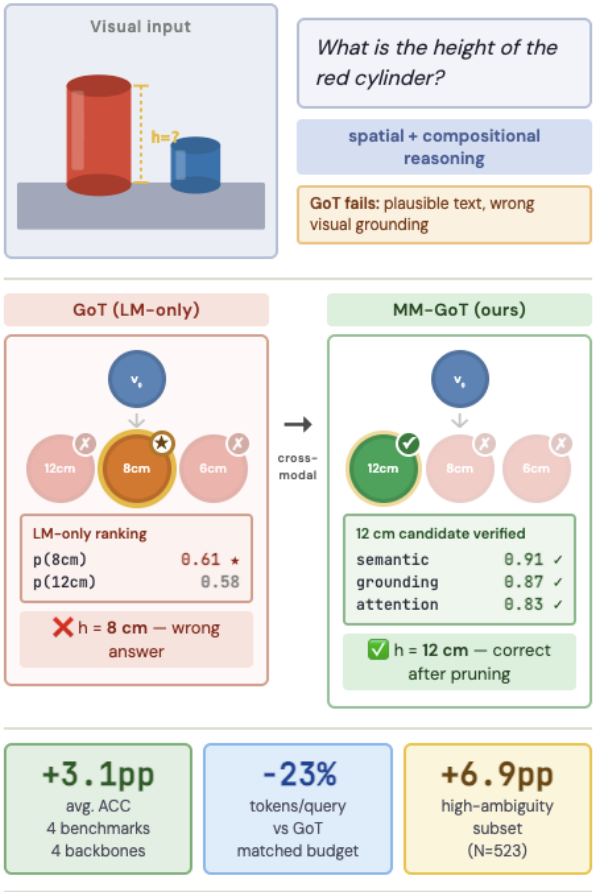


Figure 1. Illustrative example showing how MM-GoT corrects an LM-preferred but visually inconsistent branch through cross-modal verification.

1. Introduction

Multimodal foundation models have achieved impressive progress on vision–language tasks [4, 6, 9, 11, 19], yet they remain brittle on problems requiring abductive inference, defeasible belief revision, and grounded multi-step reasoning. This limitation is especially visible in settings involving visual ambiguity and conflicting evidence, where

models produce explanations that are linguistically coherent but inconsistent with what is actually perceived. Such failures reflect a deeper cognitive gap: current vision–language models can generate plausible rationales, but lack a mechanism for maintaining, revising, and verifying intermediate hypotheses against perceptual evidence as reasoning unfolds. Recent approaches including Chain-of-Thought (CoT) [16], multimodal CoT (MM-CoT) [12, 22], Self-Consistency [15], Tree-of-Thoughts (ToT) [13, 18, 20], and Graph-of-Thoughts (GoT) [2, 14] improve multi-step inference by exploring alternative reasoning paths. However, these methods operate primarily in language space, ranking hypotheses by generative plausibility rather than perceptual validity. Visually inconsistent interpretations therefore remain competitive throughout the reasoning process, allowing early perceptual errors to propagate into the final answer.

We introduce **Multimodal Graph-of-Thoughts (MM-GoT)**, a cognitively motivated framework for deliberative multimodal inference that incorporates cross-modal verification directly into graph-structured hypothesis search. MM-GoT augments each reasoning node with modality-specific verification energies for semantic alignment, spatial compatibility, and perceptual consistency — instantiating defeasible belief revision as a computational mechanism. This enables the model to suppress perceptually unsupported branches, preserve competing hypotheses under ambiguity, and integrate mutually compatible trajectories before committing to an answer, analogous to evidence accumulation in dual-process theories of cognition.

We evaluate MM-GoT on four benchmarks spanning complementary aspects of grounded multimodal reasoning: BlackSwan [5] for abductive and defeasible reasoning under contradictory evidence, MME [8] for broad multimodal perception and cognition, MMMU-Pro [21] for expert-level multimodal inference, and MMStar [3] for structured visual reasoning grounded in perceptual evidence. Under matched inference budgets, MM-GoT consistently outperforms CoT, ToT, and GoT baselines, with the largest gains on tasks where correct reasoning requires revising plausible hypotheses in light of visual evidence.

Our contributions are summarized as follows:

(1) **Multimodal verification-constrained graph search.** We formulate multimodal reasoning as inference-time search over a hypothesis graph augmented with cross-modal verification, moving beyond purely language-based structured reasoning.

(2) **Perceptual grounding as structural constraint.** We introduce modality-specific verification energies for semantic alignment, spatial compatibility, and perceptual consistency that guide branch expansion, pruning, and merging without additional model training.

(3) **Multi-path evidence synthesis under ambiguity.** We demonstrate that parallel reasoning trajectories can be reconciled through perceptual compatibility, improving robustness on multimodal tasks that require revising plausible hypotheses in light of visual evidence. Fig. 2 illustrates the main results of the paper.

2. Methodology

2.1. MM-GoT method

MM-GoT performs inference-time graph search by iterating candidate generation, modality-specific verification using semantic and perceptual consistency signals, and ranking, pruning, and synthesis—suppressing textually plausible but perceptually invalid branches while merging mutually supported trajectories.

2.2. MM-GoT Reasoning Problem Formulation

Let $\mathcal{X} = \{x^{(1)}, \ldots, x^{(M)}\}$ denote the available input modalities, where one modality is text (question and prompt) and the others may include images, diagrams, or spatial signals. The goal is to infer an output y that is jointly consistent with all modalities in $\mathcal{X}$. MM-GoT performs inference via structured exploration over a directed hypothesis graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, constructed dynamically at inference time to maintain multiple competing reasoning trajectories instead of committing to a single chain.

Multimodal reasoning states. Each node $v \in \mathcal{V}$ represents a discrete reasoning state

$$v = \big(t_v,\, \mathcal{Z}_v\big), \tag{1}$$

where t_v is a textual hypothesis (a candidate rationale or step) and $\mathcal{Z}_v = \{z_v^{(1)}, \ldots, z_v^{(M)}\}$ are modality-specific grounding variables that provide evidence for verifying t_v (entity localizations, spatial relations, and attention-derived evidence). These grounding variables enable explicit cross-modal verification at each state during graph search.

2.3. Verification-Constrained Multimodal Reasoning Objective

To ensure reasoning remains grounded in perceptual evidence, we move beyond scoring nodes by textual likelihood alone. We define a multimodal scoring function $E(v \mid \mathcal{X})$ that incorporates modality-specific verification energies, casting inference as a search for the lowest-energy state. For a given node v, the total energy is defined as

$$\begin{aligned} E(v \mid \mathcal{X}) = &-\lambda \log P_\theta(t_v \mid t_{\text{prev}}, \mathcal{X}) \\ &+ (1-\lambda) \sum_{i=1}^{M} \Psi_i\Big(z_v^{(i)}, x^{(i)}\Big), \end{aligned} \tag{2}$$

where $\lambda \in [0, 1]$ balances generative plausibility against cross-modal verification, P_θ is the frozen MLLM backbone,

and Ψ_i measures alignment between grounding $z_v^{(i)}$ and modality $x^{(i)}$. Inference proceeds by minimizing $E(v \mid \mathcal{X})$ over the hypothesis graph. During search, visually inconsistent branches characterized by high verification energy are pruned early, whereas nodes exhibiting strong cross-modal consistency are prioritized for expansion or synthesis.

2.4. Node Scoring with Cross-Modal Verification

For implementation, we instantiate the verification-constrained objective in (2) as a node-ranking score used during graph search. Higher scores indicate more promising candidates for expansion, pruning, and synthesis. Concretely, MM-GoT replaces language-only hypothesis ranking with a verification-aware node score:

$$s(v_i) = \underbrace{\alpha_{\mathrm{LM}} \log p_\theta(h_i)}_{\text{LM prior}} + \underbrace{\alpha_1 \, \phi_{\mathrm{VQA}}(v_i)}_{\text{semantic consistency}} + \underbrace{\alpha_2 \, \phi_{\mathrm{GND}}(v_i)}_{\text{spatial validity}} + \underbrace{\alpha_3 \, \phi_{\mathrm{ATT}}(v_i)}_{\text{attentional grounding}}, \tag{3}$$

where h_i is the textual hypothesis at node v_i, p_θ is the frozen MLLM backbone, and ϕ_{VQA}, ϕ_{GND}, and ϕ_{ATT} denote the semantic consistency, spatial grounding, and attentional grounding verifiers, respectively. The nonnegative coefficients $\{\alpha_{\mathrm{LM}}, \alpha_1, \alpha_2, \alpha_3\}$ are selected by grid search on the validation split. We use $s(v_i)$ to rank candidate nodes for expansion, pruning, and synthesis during graph search. Equivalently, this ranking corresponds to minimizing an energy $E(v_i) = -s(v_i)$ up to affine transformation. We instantiate the verification term using three bounded functions, ϕ_{VQA}, ϕ_{GND}, and ϕ_{ATT}; their exact definitions are provided in Supplementary Materials Sec. A.1–A.3.

2.5. Graph Topology and Reasoning Operations

Inference proceeds via structured exploration over $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Edges encode functional transformations:

- **Refinement ($v_i \to v_j$):** Transforms a coarse hypothesis into a more granular representation while preserving semantic consistency.
- **Decomposition ($v_i \to \{v_{j_1}, v_{j_2}, \dots\}$):** Splits a complex multimodal reasoning problem into structured sub-tasks for parallel exploration.
- **Revision ($v_i \rightsquigarrow v_k$):** Introduces corrective hypotheses when cross-modal verification reveals inconsistency. If the verification score improves for v_k, the search pivots toward this corrected state.

2.6. Multi-Path Evidence Synthesis (Merging)

A unique property of the MM-GoT framework is its ability to perform **node merging** when distinct reasoning trajectories converge. This operation occurs when two reasoning paths, $\mathcal{P}_1$ and $\mathcal{P}_2$, arrive at semantically equivalent hypotheses t that are grounded in aligned perceptual evidence $\mathcal{Z}$. If the distance between their groundings $\mathrm{dist}(\mathcal{Z}_a, \mathcal{Z}_b)$ is below a threshold ϵ, the nodes are synthesized:

$$v_{\text{merged}} = \mathcal{M}(v_a, v_b). \tag{4}$$

The merged node utilizes the multimodal scoring function (2) to calculate a reinforced energy value:

$$E(v_{\text{merged}} \mid \mathcal{X}) = \mathrm{Agg}\big(E(v_a \mid \mathcal{X}), E(v_b \mid \mathcal{X})\big), \tag{5}$$

where $\mathrm{Agg}(E_a, E_b) = \frac{1}{2}(E_a + E_b) - \gamma\big(1 - \mathrm{dist}(Z_a, Z_b)/\epsilon\big)$ lowers the merged node's energy proportionally to the perceptual agreement between the two paths, with $\gamma \geq 0$ tuned on the validation split alongside λ. This synthesis reduces the effective branching factor of the search and improves the robustness of the final inference.

2.7. Inference-Time Search

The inference procedure in MM-GoT is a dynamic, energy-guided exploration of the hypothesis graph $\mathcal{G}$. Unlike standard autoregressive decoding, our search algorithm iteratively expands the frontier of reasoning states while enforcing cross-modal grounding.

2.7.1. Best-First Hypothesis Expansion

We maintain a priority queue of active reasoning nodes, ordered by their total energy $E(v \mid \mathcal{X})$ as defined in (2). At each step, the algorithm selects the lowest-energy node v_{best} for expansion. For the selected node, the underlying model generates a set of candidate child hypotheses $\{t_{v'_1}, t_{v'_2}, \dots, t_{v'_k}\}$. Crucially, for each candidate, the model also predicts or retrieves the associated **perceptual anchors** $\mathcal{Z}_{v'}$ from the input modalities $\mathcal{X}$.

2.7.2. Energy-Based Pruning and Merging

Once a child node v' is generated, its verification energy $\Psi(v')$ is computed against the input modalities $\mathcal{X}$. This provides a real-time check against the raw data before the branch is allowed to propagate further:

1. **Pruning:** If the total energy $E(v' \mid \mathcal{X})$ exceeds a dynamic threshold τ_{prune}, the branch is discarded. This early exit prevents the accumulation of perceptual errors and linguistic hallucinations.
2. **Merging:** If the node survives pruning, the algorithm evaluates existing frontier nodes for semantic and perceptual equivalence. If a match is found, the synthesis operation $\mathcal{M}(v, v')$ is performed as described in Sec. 2.6.

The search terminates when a terminal node reaches the maximum reasoning depth or a high-confidence solution y is extracted from the lowest-energy path in $\mathcal{G}$.

Algorithm 1 MM-GoT Inference-Time Search

Input: Multimodal input $\mathcal{X}$, energy threshold τ, max depth D
Initialize: Priority Queue $Q \leftarrow \{v_{start}\}$, Graph $\mathcal{G} \leftarrow \{v_{start}\}$
while Q is not empty **and** depth $(\mathcal{G}) < D$ **do**
  $v \leftarrow$ pop_min_energy(Q) {Select best frontier node}
  $\{t_{v'}, \mathcal{Z}_{v'}\} \leftarrow$ LLM_Expand$(v, \mathcal{X})$ {Generate child hypotheses}
  for each candidate child node v' **do**
    Compute $E(v' \mid \mathcal{X})$ via Eq. (2) {Cross-modal verification}
    if $E(v' \mid \mathcal{X}) > \tau$ **then**
      Prune v' {Discard grounded hallucination}
    else
      if $\exists v_{match} \in \mathcal{G}$ s.t. is_mergeable(v', v_{match}) **then**
        $v_{new} \leftarrow \mathcal{M}(v', v_{match})$ {Multi-path synthesis}
        Update $\mathcal{G}$ with v_{new} and push to Q
      else
        Add v' to $\mathcal{G}$ and push to Q
      end if
    end if
  end for
end while
Output: Path $\mathcal{P} \subset \mathcal{G}$ with minimal cumulative energy

3. Evaluation

3.1. Models and Benchmarks

We evaluate MM-GoT on four open-weight multimodal large language models representing diverse architectural families: Qwen3-VL-Thinking [1], DeepSeek-VL2 [17], GLM-4.6V [10], and InternVL3 [23]. MM-GoT changes only the inference procedure and leaves model weights unchanged. We choose these backbones for their strong multimodal and mathematical reasoning performance, support for extended test-time reasoning, and public availability, which enables reproducible evaluation. Their diversity in scale and architecture also allows us to assess whether MM-GoT generalizes across heterogeneous open multimodal backbones.

We evaluate MM-GoT on four benchmarks spanning complementary multimodal reasoning abilities: BlackSwan [5] for defeasible and abductive reasoning under contradictory evidence, MME [8] for broad multimodal perception and cognition, MMMU-Pro [21] for expert-level multimodal reasoning, and MMStar [3] for structured multimodal reasoning with substantial reliance on visual evidence.

3.2. Baselines, Metrics, and Protocol

We compare against CoT [16] and MM-CoT [12, 22], ToT [13, 20], and GoT [2, 14]. GoT is the critical baseline because it matches graph-structured hypothesis exploration while isolating the effect of cross-modal verification.

The primary metric is accuracy (ACC, %) under each benchmark's official evaluation protocol. We also report $\Delta\text{ACC} = \text{ACC}(\text{MM-GoT}) - \text{ACC}(\text{GoT})$ in percentage points (p.p.), and tokens per query (Tok/Query) as an inference-cost proxy, where token usage is defined as the total number of input and output tokens consumed over the full decoding trajectory.

Unless otherwise noted, all methods use identical task prompts and matched inference budgets, including the same maximum generation budget, maximum search depth, and maximum number of node expansions. Token usage is measured over the full decoding process, with early-terminated runs contributing only the tokens generated before the stopping criterion is met. For each method, we report the mean and standard deviation over 64 seeded runs. Statistical significance is assessed using paired bootstrap tests over evaluation examples, with Bonferroni correction for multiple comparisons [7].

4. Results

Table 1 reports accuracy ACC (%) across benchmarks, backbones, and inference-time baselines under matched inference budgets. MM-GoT achieves the highest average accuracy on all four backbones, improving over GoT by +2.7 to +3.7 p.p., which indicates that its gains are consistent across heterogeneous open-weight MLLMs rather than dependent on a particular backbone. The largest improvements appear on BlackSwan, MMMU-Pro, and MMStar, where success depends more directly on grounded multi-step reasoning and visually supported hypothesis selection. Gains on MME are comparatively smaller, consistent with its broader perception-oriented scope. Compared to linear reasoning baselines (CoT and MM-CoT), MM-GoT reduces the propagation of perceptual errors by pruning hypotheses that are linguistically plausible but visually unsupported. Compared to structured search baselines (ToT and GoT), MM-GoT further improves branch ranking, pruning, and synthesis through cross-modal verification, yielding more reliable traversal of the reasoning graph.

Averaged over four benchmarks and four backbones, MM-GoT improves over GoT by +3.1 p.p. in ACC, with the largest average gains on Qwen3-VL-Thinking (+3.7 p.p.) and InternVL3 (+3.3 p.p.). Fig. 2 shows that gains are largest on BlackSwan, MMMU-Pro, and MMStar, while remaining consistently positive on MME. This pattern supports the claim that cross-modal verification is most useful when reasoning requires ambiguity resolution and vi-

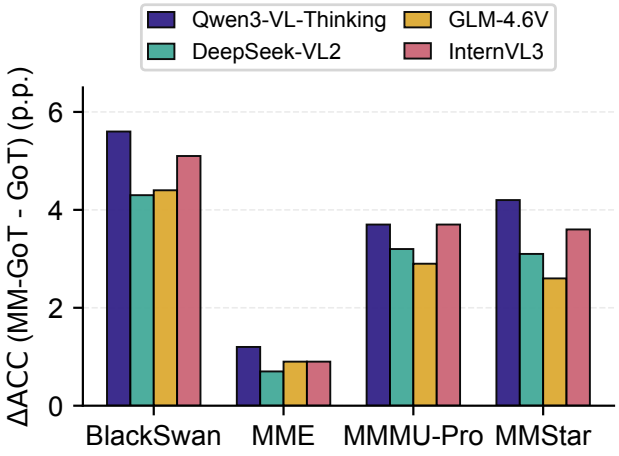


Figure 2. Per-benchmark accuracy gains of MM-GoT over GoT across four open-weight MLLM backbones (Qwen3-VL-Thinking, DeepSeek-VL2, GLM-4.6V, and InternVL3). Bars report $\Delta\text{ACC} = \text{ACC}(\text{MM-GoT}) - \text{ACC}(\text{GoT})$ on four multimodal reasoning benchmarks: BlackSwan, MME, MMMU-Pro, and MMStar.

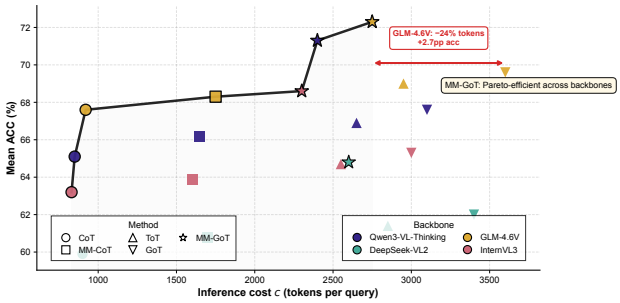


Figure 3. Accuracy–efficiency tradeoff across backbones and inference strategies. Global Pareto plot of mean accuracy (ACC, %) versus inference cost c, measured as total generated tokens per query, across all evaluated backbones and reasoning methods. Each point denotes one backbone–method configuration, and the black curve marks the Pareto frontier of non-dominated tradeoffs. Relative to GoT on the same backbone, MM-GoT (stars) shifts the tradeoff frontier upward and leftward, reducing token cost by 22–24% while improving mean accuracy by 2.7–3.7 points.

sually grounded hypothesis selection. Ablation results are reported in Sec. 5.

4.1. Inference-Time Efficiency and Search Dynamics

Fig. 3 plots mean accuracy versus generated tokens per query for all backbone–method configurations. MM-GoT lies on the accuracy–efficiency Pareto frontier, improving mean accuracy by 2.7–3.7 p.p. over GoT on the same backbone while reducing token cost by 22–24%. This indicates that its gains come from more effective search rather than larger inference budgets.

4.2. Robustness Analysis

We evaluate whether MM-GoT's gains persist under difficult conditions using benchmark-defined hard splits and two stratified cross-benchmark subsets: *high ambiguity* ($N = 523$, visually confusable entities indicated by high detector overlap and low attention-consistency scores) and *high compositionality* ($N = 601$, multi-hop structure indicated by the number of referenced entities and relations in the scene graph). All subset criteria are fixed prior to evaluation.

Table 2 reports accuracy with QWEN3-VL-THINKING as the backbone MLLM. Relative to the overall average gain of $+3.7$ (p.p.) over GoT on the main benchmarks (Table 1), MM-GoT achieves larger improvements on all challenging subsets, ranging from $+5.7$ to $+6.9$ (p.p.). The largest gain occurs on the high-ambiguity subset ($+6.9$ p.p.), consistent with cross-modal verification helping suppress perceptually inconsistent but linguistically plausible branches early in the search. The high-compositionality subset also shows a substantial improvement ($+6.6$ p.p.), suggesting that graph-structured synthesis coupled with verification benefits multi-step reasoning beyond linear or tree-based search. On the MMMU-Pro hard split, MM-GoT improves over GoT by $+5.7$ p.p., indicating that the advantage persists on benchmark-defined difficult examples as well.

5. Ablation Studies

We report mean ACC together with efficiency statistics, including tokens per query and, where applicable, incremental FLOPs/query.

5.1. Ablating Cross-Modal Verification

MM-GoT uses the verification-aware node score in (3). To isolate the contribution of each verifier, we keep the search topology and synthesis operator fixed and ablate individual verification components by zeroing the corresponding terms.

Controlled setup. All ablations use QWEN3-VL-THINKING under the same split and budget as Table 1. Search topology and synthesis operator are held fixed; only the verification coefficients in (3) are zeroed to isolate individual verifier contributions.

Verification variants. We evaluate: **(i) GoT (no verification)** with $\alpha_1 = \alpha_2 = \alpha_3 = 0$; **(ii) single-verifier** variants enabling exactly one of ϕ_{ATT}, ϕ_{GND}, or ϕ_{VQA} by setting the corresponding coefficient to a nonzero value and zeroing the remaining two; **(iii) pairwise-verifier** variants enabling two verification signals at once by zeroing only the remaining coefficient; and **(iv) full MM-GoT**, which enables all three verification signals. For readability, Table 3 refers to these components as ATT, GND, and VQA.

Table 3 shows that augmenting GoT with any cross-

Table 1. Accuracy (%) across benchmarks for each evaluated MLLM. Benchmark columns report mean accuracy over 64 runs. The average accuracy column is reported as mean ± std. All methods are evaluated under matched inference budgets and consistent prompting. Best results per backbone and benchmark are bolded. Gain over GoT is reported as ΔACC only for MM-GoT and denotes the average accuracy improvement relative to GoT for the same backbone, in percentage points (p.p.), computed over BlackSwan, MME, MMMU-Pro, and MMStar.

Backbone	Method	BlackSwan	MME	MMMU-Pro	MMStar	Avg. ACC (%)	ΔACC
Qwen3-VL-Thinking	CoT	58.4	76.8	61.1	63.9	65.1 ± 0.4	–
	MM-CoT	60.1	77.4	62.3	65.0	66.2 ± 0.4	–
	ToT	61.8	77.2	63.0	65.8	66.9 ± 0.4	–
	GoT	62.7	77.6	63.8	66.4	67.6 ± 0.4	–
	MM-GoT	**68.3**	**78.8**	**67.5**	**70.6**	$\mathbf{71.3 \pm 0.3}$	$\mathbf{+3.7}$
DeepSeek-VL2	CoT	52.9	72.5	56.1	58.0	59.9 ± 0.5	–
	MM-CoT	54.2	73.1	56.9	58.8	60.8 ± 0.4	–
	ToT	55.3	73.0	57.8	59.4	61.4 ± 0.4	–
	GoT	56.1	73.4	58.3	60.1	62.0 ± 0.4	–
	MM-GoT	**60.4**	**74.1**	**61.5**	**63.2**	$\mathbf{64.8 \pm 0.4}$	$\mathbf{+2.8}$
GLM-4.6V	CoT	61.2	78.1	64.7	66.2	67.6 ± 0.3	–
	MM-CoT	62.0	78.6	65.5	67.1	68.3 ± 0.3	–
	ToT	63.4	78.5	66.2	67.8	69.0 ± 0.3	–
	GoT	64.1	78.9	66.9	68.4	69.6 ± 0.3	–
	MM-GoT	**68.5**	**79.8**	**69.8**	**71.0**	$\mathbf{72.3 \pm 0.3}$	$\mathbf{+2.7}$
InternVL3	CoT	56.8	74.6	59.8	61.5	63.2 ± 0.4	–
	MM-CoT	57.9	75.0	60.6	62.3	63.9 ± 0.4	–
	ToT	59.0	75.1	61.5	63.0	64.7 ± 0.4	–
	GoT	59.8	75.4	62.1	63.8	65.3 ± 0.4	–
	MM-GoT	**64.9**	**76.3**	**65.8**	**67.4**	$\mathbf{68.6 \pm 0.3}$	$\mathbf{+3.3}$

Table 2. Robustness on challenging subsets using QWEN3-VL-THINKING as the backbone. Accuracy (%) on the MMMU-Pro hard split and on stratified challenging subsets. *High ambiguity*: high detector overlap among top detections and low attention-consistency scores ($N = 523$). *High compositionality*: high entity/relation count extracted from the question and the scene graph ($N = 601$). Relative to the overall average gain of $+3.7$ (p.p) over GoT in Table 1, MM-GoT achieves up to $+6.9$ (p.p) on challenging subsets.

Subset	ACC (%)			ΔACC
	MM-CoT	GoT	MM-GoT	
Hard split (MMMU-Pro, $N = 412$)	42.9	46.1	51.8	+5.7
High ambiguity ($N = 523$)	44.6	49.2	56.1	+6.9
High compositionality ($N = 601$)	51.3	55.8	62.4	+6.6

Compare to main benchmarks (Table 1): MM-CoT 66.2%, GoT 67.6%, MM-GoT 71.3% (+3.7pp).

modal verifier improves accuracy on both BlackSwan and MMMU-Pro relative to the no-verification baseline, confirming that verification is a major driver of MM-GoT's gains under a matched inference budget. Among single-verifier variants, ϕ_{VQA} provides the strongest standalone improvement on MMMU-Pro, whereas ϕ_{ATT} and ϕ_{GND} yield stronger individual gains on BlackSwan, consistent with its heavier dependence on visual grounding. Pairwise combinations consistently outperform their constituent single-verifier variants, and the full model performs best overall, indicating that semantic, spatial, and attentional verification provide complementary rather than redundant evidence during search.

Compute-optimal reasoning. To characterize the trade-off between accuracy and verification compute, we analyze the accuracy–cost Pareto frontier in Fig. 4, where accuracy is the mean ACC over BlackSwan and MMMU-Pro and cost is the normalized per-query verification overhead $\tilde{c} = \Delta\text{FLOPs}(c) \,/\, \Delta\text{FLOPs}^{*}$, with $\Delta\text{FLOPs}(c) = \text{FLOPs}(c) - \text{FLOPs}(\text{GoT})$ and $\Delta\text{FLOPs}^{*} = 2.7\text{G}$ denoting MM-GoT's total overhead. Since verifier forward passes are independent, overheads are additive and $\tilde{c} \in [0, 1]$. All configurations share the same decoding budget and search schedule, so shifts along the frontier reflect verifier compute rather than increased exploration. Single-verifier configurations improve accuracy but require at most $\tilde{c} \leq 0.56$ of the full verification budget, and ϕ_{GND} and ϕ_{VQA} are Pareto-dominated by pairwise configurations above that thresh-

Table 3. Cross-modal verification ablations on BlackSwan and MMMU-Pro using QWEN3-VL-THINKING under the same evaluation split and matched inference budget as Table 1. Mean ACC (%) $\pm$ std over 64 runs. ΔFLOPs (G) is the per-query overhead in giga floating point operations ($\times 10^9$) relative to the GoT (no verification) baseline. $^*p < 0.05$ (paired bootstrap, Bonferroni-corrected).

Verification setting	BlackSwan	MMMU-Pro	ΔFLOPs (G)
GoT (no verification)	62.7 ± 0.3	63.8 ± 0.3	–
GoT + ATT	$64.9 \pm 0.3^*$	65.1 ± 0.3	+0.3
GoT + GND	$64.7 \pm 0.3^*$	64.9 ± 0.4	+0.9
GoT + VQA	64.1 ± 0.4	$65.8 \pm 0.3^*$	+1.5
GoT + GND + ATT	66.3 ± 0.3	66.4 ± 0.3	+1.2
GoT + VQA + ATT	66.9 ± 0.3	67.4 ± 0.3	+1.8
GoT + VQA + GND	66.7 ± 0.3	67.8 ± 0.3	+2.4
MM-GoT (ATT + GND + VQA)	**68.3** ± 0.3	**67.5** ± 0.3	**+2.7**

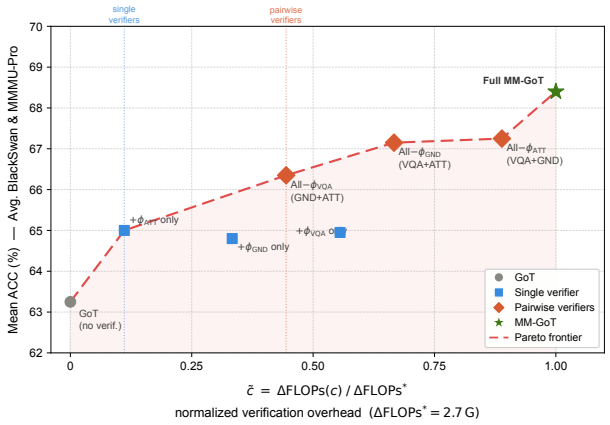


Figure 4. Accuracy–compute Pareto frontier for cross-modal verification ablations on BlackSwan and MMMU-Pro. Each point is one verifier configuration; the x-axis shows verification overhead normalized to MM-GoT's total budget. Single-verifier ablations (■) are Pareto-dominated above half the budget by pairwise configurations (⧫). Full MM-GoT (★, $\tilde{c} = 1.0$) remains Pareto-optimal and recovers efficiency after the low-return All$-\phi_{\text{ATT}}$ step, confirming that all three verifier signals contribute jointly and that no partial configuration achieves the same accuracy.

old, indicating that no single cue achieves a favourable accuracy–compute trade-off at moderate cost. All three pairwise combinations—$\phi_{\text{GND}} + \phi_{\text{ATT}}$, $\phi_{\text{VQA}} + \phi_{\text{ATT}}$, and $\phi_{\text{VQA}} + \phi_{\text{GND}}$—lie on the Pareto frontier at $\tilde{c} \in [0.44, 0.89]$, consistent with complementary error coverage across semantic checking, spatial grounding, and attentional consistency. Full MM-GoT ($\tilde{c} = 1.0$) remains Pareto-optimal and recovers efficiency after the low-return All$-\phi_{\text{ATT}}$ step, yielding $+0.65$ p.p. for the final 11% of verification budget (2.17 p.p./G), confirming that all three verifier signals contribute jointly and that no partial configuration achieves the same accuracy.

5.2. Ablating Graph Structure and Synthesis

We isolate the contribution of graph structure and synthesis from verification by holding the scoring function (3) fixed and varying topology, synthesis operator, and branching factor k, using QWEN3-VL-THINKING under the same conditions as Table 1.

Variants. We compare: **(i) GoT (no verification)**, which uses graph-structured reasoning without cross-modal verification; **(ii) tree-only + verification**, which disables node merging and multi-parent links while keeping verification fixed; **(iii) graph without synthesis + verification**, which allows graph connectivity but replaces synthesis with a single-parent update; **(iv) synthesis operator ablations**, which compare max pooling, weighted sum, and gated fusion for multi-parent aggregation; and **(v) branching factor sensitivity**, which sweeps $k \in \{1, 2, 4, 8\}$ under the full MM-GoT configuration.

Topology versus verification. Tree-only search with verification improves mean accuracy from 63.25% to 64.50% over GoT, and allowing graph connectivity without synthesis improves further to 65.85%, confirming that both verification and graph-level aggregation contribute independently.

The impact of synthesis. MM-GoT with $k = 2$ and synthesis reaches 66.55% at 1.2G FLOPs, outperforming graph search without synthesis (65.85% at the same cost), showing that multi-path aggregation extracts more signal per FLOP than non-interacting parallel branches.

Operator selection. Among the three synthesis operators, weighted sum (wsum) provides the strongest accuracy–efficiency trade-off, while gated fusion attains the highest raw accuracy at a modest additional compute cost. Specifically, wsum improves over max pooling from 66.45% to 67.20% mean accuracy while requiring 1.8G FLOPs versus 1.5G for max pooling, and remains more efficient than gated fusion, which reaches 67.60% at 2.1G FLOPs. We therefore adopt wsum as the default synthesis operator in MM-GoT because it offers the more favorable operating point under matched inference budgets.

Branching factor sensitivity. Increasing the branching factor improves accuracy up to a point, but with diminishing returns in compute efficiency. Mean accuracy rises from 64.40% at $k = 1$ to 66.55% at $k = 2$, and further to 67.90% at $k = 4$, showing that moderate graph width is important for maintaining and reconciling competing hypotheses. However, increasing to $k = 8$ yields only a marginal improvement to 68.15% while nearly doubling the compute overhead from 2.7G to 5.1G FLOPs and increasing the average number of expanded nodes from 14.6 to 27.3. We therefore adopt $k = 4$ as the default operating point, as it offers the best balance between accuracy and inference cost under matched budgets.

Table 4. Graph structure and synthesis ablations on BlackSwan and MMMU-Pro using Qwen3-VL-Thinking under the same evaluation split and matched inference budget as Table 1. Mean ACC (%) $\pm$ std over 64 runs. ΔFLOPs (G) is the per-query overhead relative to the GoT baseline. Nodes reports the average number of expanded nodes until termination. $^{*}p < 0.05$, $^{**}p < 0.01$ (paired bootstrap, Bonferroni-corrected).

Configuration	BlackSwan	MMMU-Pro	ΔFLOPs (G)	Nodes
Topology variants				
GoT (no verification)	62.7±0.3	63.8±0.3	–	12.3
Tree-only + verification	64.1±0.3*	64.9±0.3*	+0.8	10.1
Graph w/o synthesis + verification	65.6±0.3**	66.1±0.3**	+1.2	13.5
Synthesis operator variants				
Graph + max operator	66.2±0.3	66.7±0.3	+1.5	13.8
Graph + weighted sum	67.1±0.3**	67.3±0.3**	+1.8	14.1
Graph + gated fusion	67.6±0.3**	67.6±0.3**	+2.1	14.3
Branching factor k				
MM-GoT ($k = 1$)	64.0±0.3	64.8±0.3	+0.5	8.2
MM-GoT ($k = 2$)	66.4±0.3	66.7±0.3	+1.2	11.4
MM-GoT ($k = 4$)	68.3±0.3**	67.5±0.3**	+2.7	14.6
MM-GoT ($k = 8$)	68.5±0.3	67.8±0.3	+5.1	27.3
Full MM-GoT ($k = 4$)	**68.3**±0.3	**67.5**±0.3	**+2.7**	**14.6**

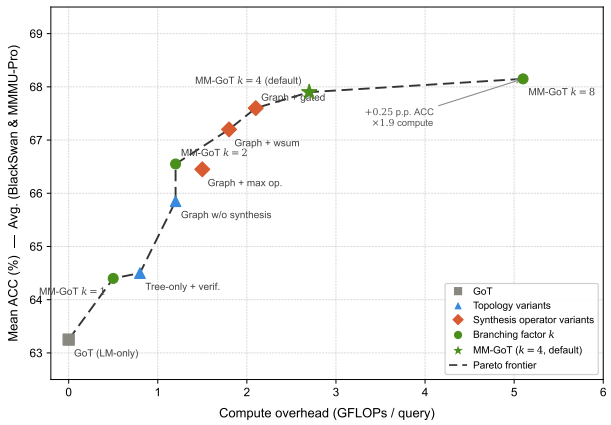


Figure 5. Accuracy–cost Pareto frontier for graph structure and synthesis ablations on BlackSwan and MMMU-Pro. Each point represents a topology or synthesis variant, and the dashed curve connects Pareto-optimal configurations. Among synthesis operators, weighted sum (wsum) offers the strongest accuracy–efficiency trade-off, while gated fusion attains slightly higher raw accuracy at higher compute cost. Full MM-GoT ($\star$, $k = 4$, 2.7G FLOPs/query) is the preferred operating point: increasing to $k = 8$ yields only a +0.25% accuracy gain at 1.9$\times$ the compute overhead.

6. Discussion

Why does cross-modal verification help? Grounding hypotheses in perceptual evidence during graph search improves both accuracy and efficiency beyond textual reasoning alone. As shown in Sec. 5, the gains are not explained by added compute: verification-guided pruning outperforms likelihood-only pruning at matched or lower cost by removing branches that are linguistically plausible but perceptually inconsistent. This mismatch between textual coherence and visual validity is the central failure mode MM-GoT addresses.

When does MM-GoT help most? The benefits are largest when visual ambiguity is high and language priors are weak. On the high-ambiguity subset, MM-GoT improves over GoT by $+6.9$ p.p., versus $+3.7$ p.p. on the full benchmark average for the same backbone, with the same trend across all four backbones. This suggests that verification is most valuable when multiple textually plausible hypotheses remain competitive and correct prediction depends on rejecting those that fail perceptual checks.

Graph structure versus verification. Verification alone is not sufficient. Variant (iv) in Sec. 5—linear reasoning with full verification but without graph structure—still underperforms MM-GoT, indicating that graph-structured inference contributes independently. Verification removes perceptually invalid branches, while synthesis ((4)–(5)) aggregates compatible partial evidence across trajectories. These two mechanisms are complementary.

Limitations. ϕ_{GND} degrades on abstract or diagrammatic inputs where object-centric localization is ill-defined. ϕ_{ATT} requires access to cross-attention maps and is therefore unavailable for some black-box models; in such cases, MM-GoT can still operate with ϕ_{VQA} and ϕ_{GND}. Our robustness subsets rely on proxy ambiguity indicators rather than human-verified labels, so these results are suggestive rather than definitive. Finally, the pruning threshold τ and branching factor k are tuned on in-distribution validation splits, and their transfer to substantially different task types remains to be tested.

7. Conclusion

We introduced MM-GoT, a verification-constrained inference framework that integrates cross-modal grounding into hypothesis graph search. By augmenting reasoning nodes with semantic, spatial, and attention-based verification signals, MM-GoT prunes perceptually inconsistent branches early and synthesizes compatible reasoning trajectories before final prediction.

Across four benchmarks and four MLLM backbones, MM-GoT consistently outperforms CoT, tree-search baselines, and language-only GoT under matched inference budgets. The gains are largest on visually ambiguous and compositionally difficult examples, where language priors alone are less reliable. Ablations further show that cross-modal verification and graph-structured synthesis provide complementary benefits. Future directions include adaptive pruning thresholds, extending ϕ_{GND} to relational and layout-based grounding, and temporal consistency for video reasoning.

References

[1] 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, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report, 2025. 4

[2] Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, et al. Graph of thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI conference on artificial intelligence*, pages 17682–17690, 2024. 2, 4

[3] Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *Advances in Neural Information Processing Systems*, 37:27056–27087, 2024. 2, 4

[4] Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025. 1

[5] Aditya Chinchure, Sahithya Ravi, Raymond Ng, Vered Shwartz, Boyang Li, and Leonid Sigal. Black swan: Abductive and defeasible video reasoning in unpredictable events. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24201–24210, 2025. 2, 4

[6] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407, 2024. 1

[7] Olive Jean Dunn. Multiple comparisons among means. *Journal of the American statistical association*, 56(293):52–64, 1961. 4

[8] Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*, 2023. 2, 4

[9] Google. Gemini 3 deep think is now available in the gemini app. The Keyword (Google Blog), 2025. Accessed: Feb. 1, 2026. 1

[10] Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, et al. Glm-4.5 v and glm-4.1 v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. *arXiv preprint arXiv:2507.01006*, 2025. 4

[11] Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, et al. Glm-4.5v and glm-4.1v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. *arXiv preprint arXiv:2507.01006*, 2025. 1

[12] Huadai Liu, Kaicheng Luo, Jialei Wang, Wen Wang, Qian Chen, Zhou Zhao, and Wei Xue. Thinksound: Chain-of-thought reasoning in multimodal llms for audio generation and editing. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. 2, 4

[13] Jieyi Long. Large language model guided tree-of-thought. *arXiv preprint arXiv:2305.08291*, 2023. 2, 4

[14] Linhao Luo, Yuan-Fang Li, Gholamreza Haffari, and Shirui Pan. Reasoning on graphs: Faithful and interpretable large language model reasoning. *arXiv preprint arXiv:2310.01061*, 2023. 2, 4

[15] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations (ICLR)*, 2023. 2

[16] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022. 2, 4

[17] Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, et al. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. *arXiv preprint arXiv:2412.10302*, 2024. 4

[18] Yuxi Xie, Kenji Kawaguchi, Yiran Zhao, James Xu Zhao, Min-Yen Kan, Junxian He, and Michael Xie. Self-evaluation guided beam search for reasoning. *Advances in Neural Information Processing Systems*, 36:41618–41650, 2023. 2

[19] Jianwei Yang, Reuben Tan, Qianhui Wu, Ruijie Zheng, Baolin Peng, Yongyuan Liang, Yu Gu, Mu Cai, Seonghyeon Ye, Joel Jang, Yuquan Deng, Lars Liden, and Jianfeng Gao. Magma: A foundation model for multimodal ai agents. *arXiv preprint arXiv:2502.13130*, 2025. 1

[20] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822, 2023. 2, 4

[21] Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun, Botao Yu, Ge Zhang, Huan Sun, et al. Mmmu-pro: A more robust multi-discipline multimodal understanding benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15134–15186, 2025. 2, 4

[22] Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. Multimodal chain-of-thought reasoning in language models. *arXiv preprint arXiv:2302.00923*, 2023. 2, 4

[23] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025. 4

Supplementary Materials for the following paper: Multimodal Graph-of-Thoughts: Hypothesis-Verification Graphs for Multimodal Reasoning in Vision-Language Models

Contents

Preface

In this supplementary material, we provide additional detail to enhance the main paper's content.

A. Cross-Modal Verification Functions

This section provides the exact definitions of the three cross-modal verification functions used by MM-GoT in the main paper. These functions are the concrete verifier components of the verification-aware node score in Eq. (3) of the main paper. Each verifier takes as input a candidate textual hypothesis h_i at node v_i together with the input image $\mathcal{I}$, and returns a scalar score in $[0, 1]$, where larger values indicate stronger cross-modal support.

A.1. VQA-based Semantic Consistency

The semantic consistency verifier, denoted ϕ_{VQA}, measures whether the reasoning step expressed in h_i is semantically supported by the visual evidence in $\mathcal{I}$. Given node v_i, we query a frozen VQA model with the prompt *"Is the following reasoning step correct?"*, using h_i as the candidate reasoning step and conditioning on the image. We define

$$\phi_{\mathrm{VQA}}(v_i) = P_{\mathrm{VQA}}(\text{“Yes”} \mid h_i, \mathcal{I}) \in [0, 1], \tag{1}$$

where P_{VQA} denotes the probability assigned by the frozen VQA model to an affirmative response. Higher values of $\phi_{\mathrm{VQA}}(v_i)$ indicate that the proposed reasoning step is more semantically consistent with the image.

During search, the semantic verifier is activated only when

$$\phi_{\mathrm{VQA}}(v_i) > \tau_{\mathrm{VQA}}, \tag{2}$$

where $\tau_{\mathrm{VQA}} = 0.6$ is selected on the validation split.

A.2. Grounding-based Spatial Validity

The spatial validity verifier, denoted ϕ_{GND}, measures whether entities referenced in the hypothesis h_i can be localized in the image with high confidence and limited ambiguity. Let $\mathcal{E}(h_i)$ denote the set of visual entity mentions extracted from h_i. For each entity $e \in \mathcal{E}(h_i)$, we obtain a set of candidate detections from a frozen grounding model:

$$\mathcal{B}(e) = \{(b_k, s_k)\}_{k=1}^{K}, \tag{3}$$

where b_k is a predicted bounding box and $s_k \in [0, 1]$ is its confidence score. Let $s^{(1)}(e)$ and $s^{(2)}(e)$ denote the top-1 and top-2 detection scores for entity e, respectively. We define

$$\phi_{\mathrm{GND}}(v_i) = \frac{1}{|\mathcal{E}(h_i)|} \sum_{e \in \mathcal{E}(h_i)} s^{(1)}(e)\, \sigma\left(\frac{s^{(1)}(e) - s^{(2)}(e)}{\gamma} \right), \tag{4}$$

where $\sigma(\cdot)$ is the logistic function and γ controls the softness of the uniqueness margin. This formulation favors both confident localization and unambiguous grounding. When $\mathcal{E}(h_i) = \varnothing$, the grounding verifier is not applied.

During search, the spatial verifier is activated only when

$$\phi_{\mathrm{GND}}(v_i) > \tau_{\mathrm{GND}}, \tag{5}$$

where $\tau_{\mathrm{GND}} = 0.4$ is selected on the validation split.

A.3. Attention-based Grounding Consistency

The attentional grounding verifier, denoted ϕ_{ATT}, measures whether image regions attended to by the MLLM align with the regions expected from the hypothesis. Let $A \in \mathbb{R}^{N \times M}$ denote the cross-attention map from the final layer of the frozen MLLM when processing h_i with image $\mathcal{I}$, where N is the number of text tokens and M is the number of image patches. For entity-referring tokens in h_i, we compute an attention distribution a_{entity} over image patches. We compare it against an expected spatial prior a_{expected}, derived from prior grounded nodes or from available supervision when present, using cosine similarity:

$$\phi_{\text{ATT}}(v_i) = \cos(a_{\text{entity}}, a_{\text{expected}}) \in [0, 1]. \tag{6}$$

Higher values indicate that the model's internal attention is concentrated on image regions compatible with the hypothesized reasoning step.

During search, the attentional verifier is activated only when

$$\phi_{\text{ATT}}(v_i) > \tau_{\text{ATT}}, \tag{7}$$

where $\tau_{\text{ATT}} = 0.5$ is selected on the validation split.

Together, ϕ_{VQA}, ϕ_{GND}, and ϕ_{ATT} provide complementary semantic, spatial, and attention-level evidence for ranking candidate nodes during graph search. Their outputs are combined with the LM prior in the node score defined in Eq. (3) of the main paper. In the ablation study in Sec. 5.1, we enable or disable individual verifiers by zeroing the corresponding terms while keeping graph construction, search topology, and synthesis fixed.

B. Ablation Study

B.1. Cross-Benchmark Performance Profile

Evaluating a reasoning framework on a limited set of benchmarks risks tying conclusions to specific task formats or visual domains. To assess whether MM-GoT generalizes beyond any single benchmark, we evaluate all methods on four datasets spanning abductive and defeasible multimodal reasoning (BlackSwan), broad multimodal perception and cognition (MME), expert-level multimodal understanding (MMMU-Pro), and general multimodal reasoning (MMStar). Table S.1 and Figure S.2 summarize the resulting cross-benchmark performance profile. To reduce backbone-specific effects, we report results averaged across all evaluated MLLM backbones (Qwen3-VL-Thinking, DeepSeek-VL2, GLM-4.6V, and InternVL3), and provide per-backbone breakdowns in Supplementary Table S.2.

Several patterns emerge. First, MM-GoT achieves its largest gains on BlackSwan and MMStar, where reasoning requires resolving ambiguity and maintaining multiple competing hypotheses under perceptual evidence. Second, gains on MMMU-Pro remain consistent across backbones, indicating that MM-GoT improves expert-level multimodal understanding beyond any single model family. Third, improvements on MME are smaller in absolute terms but remain positive, which is expected on a broad perception-and-cognition benchmark with less headroom for structured search. Finally, the relative ordering of methods (CoT, MM-CoT, ToT, GoT, and MM-GoT) is preserved across all four benchmarks, suggesting that the progression from linear prompting to verification-constrained graph search reflects a stable hierarchy of inference-time reasoning capability rather than benchmark-specific tuning. Collectively, these results support MM-GoT as a general-purpose multimodal reasoning framework.

B.2. Ablating Pruning and Ranking Policies

To test the hypothesis that verification improves search efficiency by suppressing inconsistent branches early, we ablate pruning and ranking policies while holding the search expansion budget (maximum node expansions) fixed. We compare:

Supplementary Table S.1. Cross-benchmark performance profile averaged across all evaluated MLLM backbones (Qwen3-VL-Thinking, DeepSeek-VL2, GLM-4.6V, and InternVL3). ΔACC denotes the absolute gain in percentage points (p.p.) of MM-GoT over GoT. The corresponding plot is shown in Fig. S.2.

Benchmark	**ACC (%)**					**ΔACC**
	CoT	**MM-CoT**	**ToT**	**GoT**	**MM-GoT**	
BlackSwan	57.3	58.6	59.9	60.7	**65.5**	+4.8
MME	75.5	76.0	76.0	76.3	**77.3**	+0.9
MMMU-Pro	60.4	61.3	62.1	62.8	**66.2**	+3.4
MMStar	62.4	63.3	64.0	64.7	**68.1**	+3.4
Avg	64.0	64.8	65.5	66.1	**69.3**	**+3.1**

Avg is the unweighted mean across the four benchmarks.
ΔACC is computed relative to GoT: $\Delta\text{ACC} = \text{ACC}_{\text{MM-GoT}} - \text{ACC}_{\text{GoT}}$ (p.p.).

Supplementary Table S.2. Per-backbone breakdown of cross-benchmark performance, ordered by decreasing Full MM-GoT average accuracy. We report GoT, Full MM-GoT, and ΔACC (p.p.).

Benchmark	**ACC (%)**											
	GLM-4.6V			**Qwen3-VL-Thinking**			**InternVL3**			**DeepSeek-VL2**		
	GoT	**MM-GoT**	**ΔACC**	**GoT**	**MM-GoT**	**ΔACC**	**GoT**	**MM-GoT**	**ΔACC**	**GoT**	**MM-GoT**	**ΔACC**
BlackSwan	64.1	68.5	+4.4	62.7	68.3	+5.6	59.8	64.9	+5.1	56.1	60.4	+4.3
MME	78.9	79.8	+0.9	77.6	78.8	+1.2	75.4	76.3	+0.9	73.4	74.1	+0.7
MMMU-Pro	66.9	69.8	+2.9	63.8	67.5	+3.7	62.1	65.8	+3.7	58.3	61.5	+3.2
MMStar	68.4	71.0	+2.6	66.4	70.6	+4.2	63.8	67.4	+3.6	60.1	63.2	+3.1
Avg	69.6	72.3	**+2.7**	67.6	71.3	**+3.7**	65.3	68.6	**+3.3**	62.0	64.8	**+2.8**

Avg is the unweighted mean across the four benchmarks.
ΔACC denotes the absolute gain of MM-GoT over GoT in percentage points (p.p.).
Wins: Full MM-GoT outperforms GoT on **4/4** benchmarks for each backbone.

(i) No pruning. All candidate nodes are retained up to the expansion limit, maximizing branching and allowing inconsistent branches to persist throughout the search.

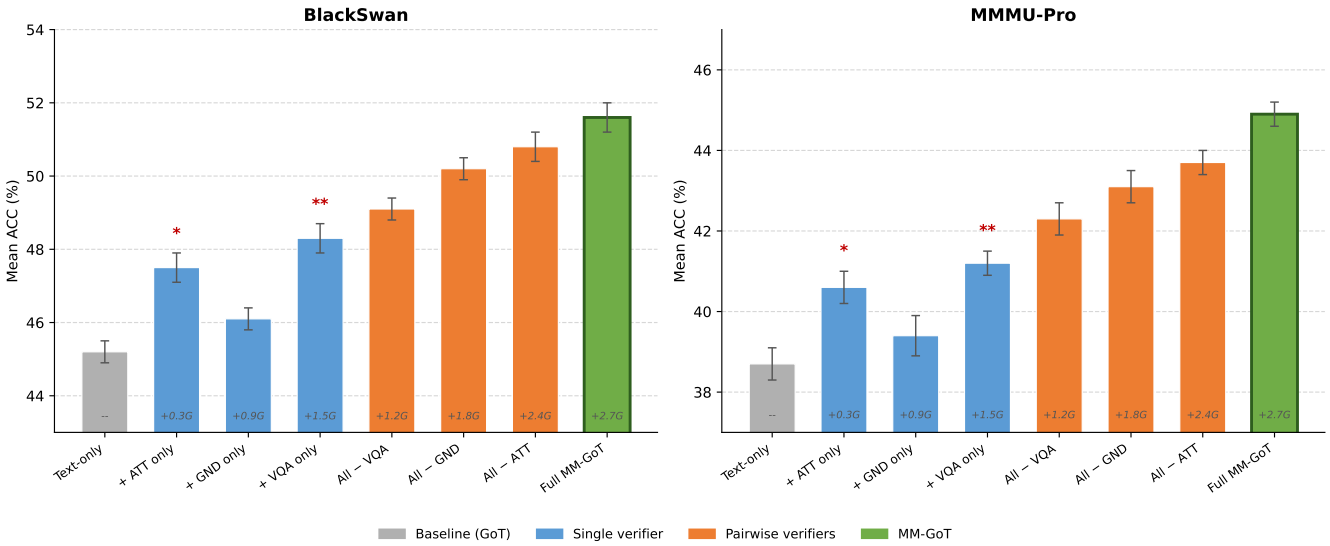
(ii) Likelihood-only pruning. Nodes are pruned using text-likelihood scores rather than verification scores, isolating the effect of pruning from the effect of cross-modal verification signals.

(iii) Verification-based pruning (MM-GoT default). Nodes are pruned using the verification-constrained objective in Eq. (5), suppressing visually inconsistent branches early in the search.

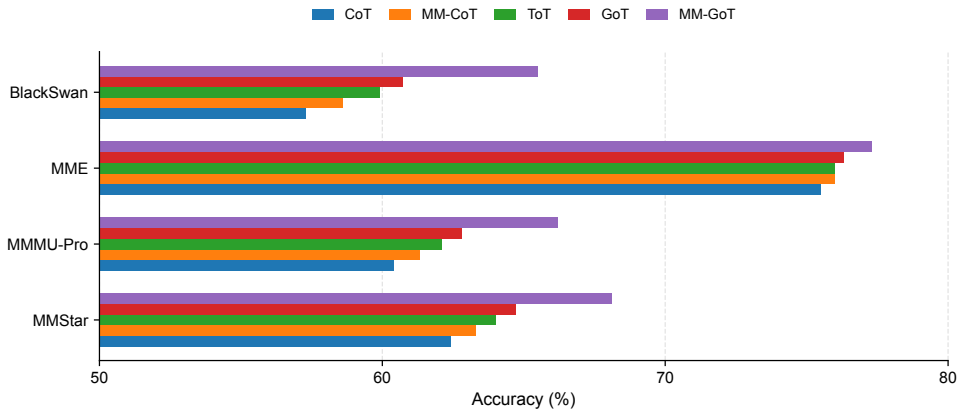
(iv) Threshold and top-k sweeps. We vary the pruning threshold τ and retained set size $k \in \{1, 2, 4, 8\}$ to characterize sensitivity and identify stable operating points across benchmarks.

Table S.3 reports mean accuracy (ACC (%)), tokens per query (Tok/Q), and the fraction of pruned nodes on BlackSwan and MMMU-Pro.[1] Verification-based pruning improves both accuracy and efficiency relative to likelihood-only pruning, indicating that cross-modal verification removes branches that are textually plausible yet visually inconsistent. Removing pruning entirely yields the highest token cost and the lowest accuracy, suggesting that retaining all branches amplifies perceptual errors rather than resolving them. Sweeps over τ show a broad stable region for $\tau \in [0.3, 0.6]$, with performance degrading under overly aggressive pruning ($\tau = 0.2$) and under-pruning ($\tau = 0.8$), motivating $\tau = 0.5$ as a robust default. Top-k sweeps indicate that $k = 4$ provides a favorable accuracy–efficiency trade-off, consistent with the branching-factor analysis in Sec. 5.2 of the main text.

[1] In our implementation, smaller τ increases pruning aggressiveness, while larger τ retains more candidates.



Supplementary Figure S.1. Per-benchmark accuracy breakdown for cross-modal verification ablations. Mean ACC (%) ± std over 64 runs on BlackSwan (left) and MMMU-Pro (right) for each verifier configuration. ΔFLOPs (G) annotations are shown below each bar. $^{*}p < 0.05$, $^{**}p < 0.01$ (paired bootstrap, Bonferroni-corrected). Colors correspond to Figure 3 of the main paper.



Supplementary Figure S.2. Cross-benchmark performance profile. Mean ACC (%) on BlackSwan, MME, MMMU-Pro, and MMStar for CoT, MM-CoT, ToT, GoT, and MM-GoT, averaged across evaluated MLLM backbones. Corresponding values are reported in Table S.1, with per-backbone results in Supplementary Table S.2.

B.3. Budget Sensitivity and Scaling

We study how MM-GoT scales with inference-time compute by varying the per-query ΔFLOP budget while holding the backbone, prompt template, and maximum reasoning depth fixed. All methods share the same candidate-generation procedure and expansion schedule at each budget level, so observed differences reflect compute utilization rather than decoding artifacts.

Sample efficiency. Fig. 3 (main text) shows that MM-GoT's accuracy gains are front-loaded within the compute budget. At a per-query overhead of only $0.3{\times}10^9$ ΔFLOPs (achieved by enabling the attention-consistency verifier alone), mean accuracy across BlackSwan and MMMU-Pro rises from 41.95% (text-only baseline) to 44.05%, a gain of $+2.1$ p.p. for roughly 11% of the full MM-GoT overhead. Adding pairwise verifiers further increases accuracy to 46.65% at $1.8{\times}10^9$ ΔFLOPs, recovering approximately 65% of the full-system gain at approximately 67% of its cost. Full MM-GoT reaches 48.25% ($+6.3$ p.p. over the text-only baseline) at 2.7×10^9 ΔFLOPs.

These gains arise because verification concentrates compute on visually consistent hypotheses and suppresses perceptually inconsistent branches early, preventing them from accumulating downstream token cost; by contrast, likelihood-driven search allocates compute more uniformly across candidates regardless of visual validity.

Diminishing returns at high budgets. The graph-structure and synthesis ablation in Fig. 4 (main text) illustrates the opposite end of the scaling curve. Increasing the branching factor from $k{=}4$ (MM-GoT) to $k{=}8$ yields only $+0.2$pp accuracy at $1.9\times$ the per-query ΔFLOP cost, placing $k{=}8$ off the accuracy–efficiency Pareto frontier. This saturation arises because, at high budgets, the residual expanded branches are already largely visually consistent; additional exploration refines plausible hypotheses rather than recovering from perceptual errors, yielding negligible accuracy improvement per unit of added compute.

Practical operating region. Together, the two Pareto frontiers define a favorable operating region between roughly 1.2–2.7×10^9 ΔFLOPs, where additional compute yields consistent accuracy gains and MM-GoT dominates the ablated variants in the accuracy–efficiency trade-off. Outside this range—either under extremely tight budgets where aggressive pruning reduces candidate diversity, or under very large budgets where verification becomes redundant—marginal returns diminish. Our default setting of $k{=}4$ lies near the knee of this curve and is used throughout the paper. Unlike scalar reward models employed in Best-of-N test-time scaling [1, 2], MM-GoT's verification signals are compositional and modality-grounded, enabling informative pruning at low budgets rather than relying on post-hoc candidate re-ranking.

Supplementary Table S.3. Pruning policy ablations on BlackSwan and MMMU-Pro. Mean ACC (%) ± std over 64 runs. Tok/Q is the average number of tokens per query. Pruned (%) is the fraction of candidate nodes removed before synthesis.

Configuration	**BlackSwan**	**MMMU-Pro**	**Tok/Q**	**Pruned (%)**
Pruning strategy				
No pruning	49.1±0.4	42.6±0.3	2840	0.0
Likelihood-only	50.3±0.3	43.4±0.4	2210	31.4
Verif.-based (MM-GoT, default)	**51.6**±0.4	**44.9**±0.3	1980	38.7
Threshold τ sweep (verif.-based, $k = 4$)				
$\tau = 0.2$ (over-pruning)	47.3±0.5	40.8±0.4	1340	61.2
$\tau = 0.3$	50.9±0.3	44.1±0.3	1820	42.3
$\tau = 0.5$ (default)	**51.6**±0.4	**44.9**±0.3	1980	38.7
$\tau = 0.6$	51.4±0.3	44.6±0.4	2050	35.1
$\tau = 0.8$ (under-pruning)	50.1±0.4	43.2±0.3	2560	18.3
Top-k retained nodes sweep ($\tau = 0.5$)				
$k = 1$	48.4±0.3	41.7±0.4	1420	52.1
$k = 2$	50.2±0.4	43.3±0.3	1710	44.6
$k = 4$ (default)	**51.6**±0.4	**44.9**±0.3	1980	38.7
$k = 8$	51.7±0.3	45.0±0.4	2640	24.3

References

[1] Aisha Khatun and Daniel G. Brown. A study on large language models' limitations in multiple-choice question answering. *arXiv preprint arXiv:2401.07955*, 2024. 6

[2] Charlie Snell, Kanishk Jaeckle, Aviral Kumar, and Sergey Levine. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024. 6